

Politechnika Wroclawska
Wydział Informatyki i Zarządzania

Podejście hybrydowe
w analizie ekonomicznej przedsiębiorstwa

Jacek Zabawa

Rozprawa doktorska napisana pod kierunkiem
prof. dr hab. inż. Edwarda Radosińskiego

Słowa kluczowe:
systemy wspomaganie decyzji
symulacja komputerowa
systemy hybrydowe
analiza finansowa

Wrocław 2005

Spis treści

1	Analiza decyzyjna	1-1
1.1	Określenie przedmiotu badań. Systematyka podejmowania decyzji oraz znaczenie kapitału intelektualnego	1-1
	Omówiono własności decyzji menedżerskich. Podkreślono złożony charakter procesu podejmowania decyzji i konieczność opracowania ich metodologii. Przedstawiono znaczenie pojęcia kapitału intelektualnego w uwzględnieniu oprócz wielu wskaźników ekonomicznych także rozumienia znaczenia zjawisk. Przedstawiono argumenty za koniecznością rozważania wielu miar w zadaniu wyznaczania wartości przedsiębiorstwa.	
1.2	Analiza decyzyjna jako proces	1-3
	Przedstawiono kategorie działań badawczych prowadzących do podjęcia decyzji. Podkreślono konieczność opracowania alternatyw decyzyjnych i ich oceny. Omówiono związki analizy decyzyjnej ze strukturalizacją problemów oraz weryfikacją symulacyjną.	
1.3	Analiza wielokryterialna, modelowanie wariantów decyzyjnych, ocena ryzyka	1-4
	Przedstawiono rozważania nad podejmowaniem decyzji przy uwzględnieniu wielu kryteriów, zmian już wybranych wariantów oraz nad istotnością posiadania odpowiedniej informacji. Omówiono fazy analizy decyzyjnej i ich zgodność z etapami budowy modelu. Zaprezentowano techniki modelowania niepewności za pomocą tabeli wypłat, drzewa decyzyjnego, teorii wartości informacji, analizy pre posteriori oraz analizy wrażliwości.	
1.4	Metodologia i instrumentarium analizy decyzyjnej	1-9
	Przedstawiono porównanie metod analizy ekonometrycznej i modelowania symulacyjnego. Omówiono znaczenie weryfikacji merytorycznej w badaniu hipotez. Przedstawiono klasyfikację wg regularności podejmowania decyzji.	
1.5	Metody optymalizacyjne w analizie decyzyjnej	1-11
	Zdefiniowano pojęcie optymalizacji. Przedstawiono istotę wielokryterialności, skalaryzowanie, optymalności w sensie Pareto.	
1.6	Problemy na jakie napotyka analiza decyzyjna	1-12
	Omówiono perspektywy zastosowania analizy wrażliwości oraz integracji metod symulacyjnych i innych technik poszukiwania rozwiązań. Przedstawiono przesłanki prowadzące do systemu innowacyjnego oraz wyróżniono kompetencje przedsiębiorstwa. Wskazano na ich wsparcie przez systemy wspomagania decyzji.	
1.6.1	Analiza decyzyjna a pojedynczy podmiot decyzyjny	1-13
	Podkreślono konieczność zastosowania zasady racjonalności w trakcie podejmowania decyzji oraz jej krytykę z pozycji psychologii kognitywnej wynikającą z ograniczeń reprezentacji problemu.	
1.6.2	Uwzględnienie grupowego podmiotu decyzyjnego	1-14
	Przedstawiono uwarunkowania grupowego podejmowania decyzji oraz jego wymiary: społeczny, teoriogrowy i sieci komunikacji społecznych. Omówiono komunikację synchroniczną i asynchroniczną oraz poziomy komunikacji.	
1.6.3	Podejmowanie decyzji – aspekty psychologiczne – strategie	1-16
	Przedstawiono uwarunkowania strategii o charakterze autokratycznym oraz partycypacyjnym.	
1.6.4	Zebranie informacji o procesach gospodarczych	1-17
	Omówiono kluczową rolę rachunkowości jako źródła wiarygodnej informacji dla analizy oraz prezentacji stanu faktycznego.	

1.7	Podejmowanie decyzji w warunkach niepewności	1-19
1.7.1	Zastosowanie podejścia teoriogrowego oraz wiedzy eksperckiej	1-19
	Omówiono kryteria oceny wariantów decyzyjnych (max-min). Podkreślono wagę wiedzy eksperckiej i przesłanki podejścia hybrydowego. Przedstawiono zalecenia dotyczące postępowania w warunkach ryzyka.	
1.7.2	Symulacja i analiza statystyczna	1-21
	Omówiono zastosowanie symulacji i podstawowe pojęcia statystyki.	
1.7.3	Drzewa prawdopodobieństw	1-22
	Przedstawiono technikę reprezentacji niepewności za pomocą drzew prawdopodobieństw i rolę reguły Bayesa.	
1.7.4	Aspekty zarządzania ryzykiem i ocena zagrożenia	1-23
	Omówiono istotę systemów wczesnego ostrzegania i modele dekompozycji i agregacji ryzyka.	
1.7.5	Technika estymacji	1-24
	Zaprezentowano technikę szacowania nieznanymi wartościami zmiennych w celu usunięcia z modelu niepewności.	
1.8	Analiza decyzyjna jako przedmiot badań	1-26
	Przedstawiono podstawowe pojęcia analizy decyzyjnej takie jak wybór, proces poznawczy i celowość. Zdefiniowano pojęcia tabeli wypłat i funkcji użyteczności. Omówiono klasyfikację reguł decyzyjnych.	
1.9	Podejmowanie decyzji jako proces fazowy	1-28
	Przedstawiono podział procesu podejmowania decyzji na poszczególne fazy wyróżnione przez Simona a następnie rozszerzone przez Mintzberga.	
2	Modelowanie i symulacja	2-1
2.1	Symulacja i modelowanie symulacyjne - przedmiot badań	2-1
	Zdefiniowano pojęcie modelowania. Podkreślono rolę modelowania i symulacji w badaniach naukowych. Określono korzyści płynące z zastosowania symulacji. Opisano działania składające się na proces modelowania. Podkreślono podobieństwa modelowania i symulacji. Zaprezentowano definicje symulacji.	
2.2	Cele zastosowania symulacji	2-3
	Przedstawiono argumenty świadczące o konieczności zastosowania symulacji jako metody weryfikacyjnej wariantów decyzyjnych oraz jej wykorzystania w analizie wrażliwości.	
2.3	Symulacja a optymalizacja	2-5
	Wskazano na odmienną naturę podejścia symulacyjnego i optymalizacji. Podkreślono przewagę symulacji pod względem realistyczności i przydatności w podejmowaniu decyzji. Omówiono charakterystykę zmiennych modelu symulacyjnego. Przedstawiono priorytety obowiązujące podczas budowy modelu symulacyjnego.	
2.4	Definicje pojęć modelowania	2-7
	Podano różnorodne definicje pojęcia modelu. Przedstawiono oczekiwania dotyczące postępowania mającego na celu budowę modelu. Podkreślono rolę prawidłowego wypełnienia modelu danymi. Omówiono użyteczne wartości oferowane przez proces modelowania i jego efekty.	
2.5	Modelowanie – klasyfikacja	2-10
	Podano klasyfikację i cele modelowania wg Savolainena, dotyczącą: procesów biznesowych, danych, finansów i symulacji, optymalizacji procesów i modelowania produktu.	

2.6	Znaczenie uwzględnienia losowości w modelach	2-13
	Omówiono charakterystykę modeli deterministycznych, probabilistycznych. Podkreślono rolę symulacyjnej techniki Monte Carlo w badaniu modeli deterministycznych i jej przydatność w decyzyjnej analizie kosztów i zysków.	
2.7	Budowa modelu symulacyjnego	2-16
2.7.1	Model konceptualny: weryfikacja i walidacja	2-16
	Przedstawiono zespoły czynności prowadzące do zbudowania modelu konceptualnego. Omówiono zagadnienia weryfikacji i walidacji modeli na platformie informatycznej. Podano szczegółowe zalecenia w kontekście budowy hybrydowego SWD Ekanwin.	
2.7.2	Etapy tworzenia modelu symulacyjnego	2-20
	Zaprezentowano charakterystykę siedmiu etapów prowadzących do opracowania modelu symulacyjnego gotowego do zatwierdzenia. Wskazano na podobieństwo z wymaganiami powodzenia (sukcesu) implementacji systemów eksperckich.	
2.7.3	Wskazówki metodologiczne i techniki walidacji	2-21
	Przedstawiono czynności wymagane do przeprowadzenia walidacji modelu. Omówiono znaczenie kalibrowania i strojenia modelu. Wskazano na realną konieczność budowy lub obsługi wielu wariantów modelu w zależności od scenariuszy warunków. Wyróżniono nieformalne i formalne metody walidacji. Omówiono walidację konceptualną i operacyjną. Przedstawiono klasyfikację technik walidacji.	
2.7.4	Badania symulacyjne – wskazówki metodologiczne, instrumentarium	2-24
	Omówiono wskazówki dotyczące badań symulacyjnych w kontekście etapów badania naukowego, od zdefiniowania problemu do wyciągnięcia wniosków z wykonanych eksperymentów. Zaprezentowano dwunastoetapowy schemat badania symulacyjnego wg Wintona. Przedstawiono pożądane charakterystyki etapu akwizycji danych i techniki ich wykonywania.	
2.7.5	Wskazówki metodologiczne - eksperymenty i raport z badań symulacyjnych	2-29
	Przedstawiono definicję eksperymentu. Przedstawiono proponowaną przez Lawrence’a i Pasternacka strukturę raportu. Podkreślono znaczenie czynnika losowości w modelowaniu niepewności. Wyróżniono trzy podstawowe podejścia prowadzenia wielowariantowych eksperymentów: takiej samej liczby eksperymentów dla wszystkich strategii, takich samych rozkładów zmiennych losowych i takich samych wartości zmiennych losowych.	
2.8	Symulacja ciągła i dyskretna	2-32
2.8.1	Pojęcia symulacji ciągłej	2-32
	Zaprezentowano podstawowe pojęcia symulacji ciągłej takie jak zasoby, strumienie i atrybuty. Omówiono znaczenie i rodzaje sprzężeń zwrotnych. Omówiono zastosowania modeli systemodynamicznych.	
2.8.2	Pojęcia symulacji dyskretniej	2-32
	Przedstawiono zastosowanie symulacji dyskretniej i główne podejścia w modelowaniu systemów zdarzeń dyskretnych. Omówiono charakterystykę pojęć wykorzystywanych w opisie modelu dyskretnego.	

2.9 Symulacja – podstawowe zalety i trudności **2-34**

Wymieniono przesłanki wykorzystania symulacji i eksperymentu w rozwiązywaniu problemów zarządzania. Przedstawiono wartości uzyskiwane dzięki symulacji, użytkowe zalety symulacji, porównano modele symulacyjne i optymalizacyjne. Omówiono trudnienia i niedogodności symulacji w tym brak możliwości zapewnienia optymalności wyznaczonych rozwiązań. Sformułowano postulaty mające prowadzić do lepszego wsparcia rozwiązywania problemów przez narzędzia symulacyjne takie jak przygotowanie szablonów programowych i konfigurowalnych modułów.

3 Zarządzanie wiedzą, sztuczna inteligencja, systemy wspomaganie decyzji **3-1**

3.1 Metodologia reprezentacji wiedzy **3-1**

Przedstawiono związek reprezentacji wiedzy z modelowaniem. Wyróżniono statyczne i dynamiczne aspekty wiedzy i klasyfikację metod odkrywania wiedzy.

3.2 Znaczenie psychologii poznania dla metodologicznych podstaw reprezentacji wiedzy i technik sztucznej inteligencji **3-2**

Omówiono klasyfikację procesów podejmowania decyzji z punktu widzenia psychologii decyzji. Przedstawiono zasady dekompozycji procesu podejmowania decyzji. Wskazano na znaczenie ontologii w procesie formalizacji wiedzy. Przedstawiono podstawowe struktury reprezentacji wiedzy w systemach eksperckich.

3.3 Wiedza a informacja i zarządzanie wiedzą **3-4**

Przedstawiono rodzaje reprezentacji wiedzy w klasyfikacji Mechtova i Moshkovicha tzn. reprezentację regułową i reprezentację za pomocą przykładów i ich charakterystykę. Zaprezentowano definicję wiedzy, cele oraz etapy jej pozyskania. Przedstawiono jej związki z informacją i danymi oraz rodzaje wiedzy i ich uwarunkowania. Omówiono kodyfikację i personalizację jako sposoby pozyskiwania i współdzielenia wiedzy.

3.4 Rola uczenia maszynowego **3-7**

Zdefiniowano pojęcie uczenia się i jego cechy charakterystyczne. Wskazano na konkurencyjność uczenia maszynowego i pozyskiwania wiedzy za pomocą dialogu z ekspertem. Zaakcentowano przydatność rozumowania przez analogie. Wyszczególniono metody uczenia maszynowego w tym nadzorowane, ze wzmocnieniem oraz bez nadzoru. Podano przykłady metod indukcyjnych i automatycznego analizowania i ich zastosowania w budowie modelu dziedziny wiedzy. Wskazano na istotną rolę heurystyk i podejścia generalizującego.

3.5 Sztuczna inteligencja + psychologia kognitywna – definicja **3-9**

Przedstawiono definicję sztucznej inteligencji.

3.5.1 Systemy wspomaganie decyzji – sieci neuronowe **3-9**

Przedstawiono definicję sztucznych sieci neuronowych. Omówiono mocne i słabe strony SSN. Zaprezentowano obszary zastosowań. Przedstawiono zasadę działania. Podkreślono znaczenie procesu uczenia (trenowania) sieci. Omówiono działanie neuronów i poszczególnych warstw SSN.

3.5.2 Systemy wspomaganie decyzji – metody wnioskowania symbolicznego **3-11**

Przedstawiono cechy wyróżniające systemów eksperckich. Omówiono zasady wnioskowania symbolicznego i reprezentację wiedzy za pomocą ram. Przedstawiono komponenty systemu eksperckiego i wskazano na kluczową rolę procesu pozyskiwania wiedzy.

3.6 Systemy eksperckie - zastosowanie w analizie finansowej **3-14**

Opisano pięć głównych grup zastosowań systemów eksperckich w analizie bankowej (oceny zagrożenia bankructwem). Przedstawiono przesłanki wykorzystania technologii SE oraz przykłady wdrożeń. Zaprezentowano także wdrożenia decyzyjnej analizy wielokryterialnej na platformie systemów wspomaganie decyzji. Omówiono podejście optymalności Pareto oraz hierarchii celów.

Wskazano na konieczność uwzględnienia alternatywnych rozwiązań podczas rozważania problemów decyzyjnych.

3.7 Systemy wspomaganie decyzji 3-17

3.7.1 Systemy wspomaganie decyzji - rys historyczny 3-17

Opisano etapy dotychczasowej ewolucji systemów wspomaganie decyzji oraz zidentyfikowano wspólne cechy dotychczasowych SWD.

3.7.2 Systemy wspomaganie decyzji – pojęcia i klasyfikacja 3-17

Przedstawiono definicję i zakres znaczeniowy pojęcia systemów wspomaganie decyzji. Rozważano umiejscowienie SWD wśród narzędzi informatycznych. Wyszczególniono cele i etapy projektowania SWD. Omówiono etapy wypełniania SWD danymi pochodzącymi z pomiarów oraz będących wynikiem ludzkiego doświadczenia. Porównano pracę SWD i systemów przetwarzania danych. Przedstawiono pięć aspektów systemowych SWD: środowisko, rolę, komponenty, układ i zasoby. Omówiono komponenty SWD.

3.7.3 Systemy wspomaganie decyzji – elementy teorii 3-21

Przedstawiono moduły funkcjonalne SWD oraz elementy skupiające zainteresowania badawcze: środowisko, zadania, strategie implementacji, możliwości, konfiguracje, użytkownika i jego zachowanie oraz wyniki działania.

3.7.4 Wsparcie SWD przez technologie przetwarzania informacji 3-23

Opisano pojęcia i zastosowanie hurtowni danych. Zwrócono uwagę na potrzebę filtrowania danych oraz zintegrowanie i nieulotność informacji pochodzących z hurtowni danych. Przedstawiono postulaty dotyczące źródeł informacji dla tematycznych hurtowni danych. Opisano korzyści związane z uruchomieniem hurtowni danych. Zaprezentowano elementy (etapy) procesu data mining. Przedstawiono założenia technologii OLAP i opisano klasy jej implementacji.

3.7.5 Przykłady doniesień literaturowych o SWD w zakresie zarządzania 3-26

Przytoczono doniesienia literaturowe na temat zastosowań systemów wspomaganie decyzji. Omówiono wiele narzędzi SWD w tym: WinQSB – wykorzystujące symulację w zarządzaniu operacjami, Folio – wspomagające zarządzanie portfelem, @Risk – wykorzystujące symulację Monte Carlo w analizie ryzyka.

3.7.6 Zadania modułu symulacyjnego SWD 3-28

Omówiono umiejscowienie modułu symulacyjnego w architekturze SWD. Wskazano rolę symulacji i prognozowania jako niezbędny warunek adaptacji do zmieniającego się środowiska. Omówiono przesunięcie paradygmatu pętli podejmowania decyzji w kierunku pojęcia perspektyw.

3.7.7 Metodologia badań i klasyfikacja SWD 3-28

Opisano czynniki mogące wpływać na osiągnięcie sukcesu SWD. Scharakteryzowano potrzebę konstruowania inteligentnych systemów informowania kierownictwa polegającą na tym, że najistotniejsze jest dostosowanie obsługi SIK do środowiska pracy menedżera. Omówiono tryby obsługi SIK. Wyszczególniono zadania systemów eksperckich. Zaprezentowano warunki w których istnieje potrzeba zastosowania grupowych systemów wspomaganie decyzji. Omówiono techniki określenia wspólnych preferencji grupy w rozproszonym procesie podejmowania decyzji. Przedstawiono poglądy z pozycji nauk o poznaniu na odmienne sposoby rozwiązywania problemów niezgodności między decydentami: autokratyczny, konsultacyjny, grupowy i delegowalnego. Wyszczególniono cechy grupowych SWD. Przedstawiono moduły zawarte w strukturze inteligentnych SWD.

3.8	Regulowe systemy eksperckie	3-34
3.8.1	Charakterystyka i struktura systemów eksperckich	3-34
	Przedstawiono definicję systemu eksperckiego. Porównano SE ze standardowymi SIK i wskazano przesłanki przewagi SE. Omówiono pojęcie faktu i reguły wnioskowania oraz typową strukturę systemu eksperckiego opartego na regułach. Przedstawiono funkcje mechanizmu wnioskowania i techniki wnioskowania: wnioskowanie wprzód, w tył oraz mieszane. Wyszczególniono i omówiono moduły składające się na system ekspercki: pozyskiwania wiedzy, monitora, interfejsu użytkownika, mechanizmu wnioskującego, modułu objaśniającego oraz bazy wiedzy. Przedstawiono różnorodne klasyfikacje wiedzy przechowywanej w bazie wiedzy. Omówiono charakter pamięci w systemie eksperckim. Przedstawiono cechy użytkowe charakterystyczne SE.	
3.8.2	Klasyfikacja regulowych systemów eksperckich	3-40
	Przedstawiono macierz i interpretację klasyfikacji systemów eksperckich wg Meyera i Curleya.	
3.8.3	Regulowe systemy eksperckie – zalety i wady	3-41
	Przedstawiono przesłanki przydatności i opłacalności SE w rozwiązywaniu problemów. Podkreślono możliwość wykorzystania wiedzy pochodzącej od wielu ekspertów. Wymieniono korzyści oczekiwane po zastosowaniu SE. Przedstawiono także trudności związane z uwzględnieniem semantyki wiedzy zawartej w SE.	
3.8.4	Modelowanie wiedzy w systemach eksperckich	3-43
	Przedstawiono dwuwymiarowy model KCM wg Holsapple'a. Omówiono pojęcie pozyskiwania wiedzy, selekcji, generowania, internalizacji i eksternalizacji wiedzy. Zwrócono uwagę na następstwa poszczególnych grup działań w KCM. Zdefiniowano działania: pomiaru, kontrolowania, koordynowania i dowodzenia wiedzą.	
3.8.5	Pozyskiwanie wiedzy w systemach eksperckich	3-45
	Opisano znaczenie środowiska rozwoju. Przedstawiono trzy sposoby pozyskiwania wiedzy w celu wypełniania bazy wiedzy: techniki analizy protokołów: równoczesną i retrospektywną, wywiad: strukturalizowany i nieustrukturalizowany oraz obserwację. Przedyskutowano ich zalety i wady. Omówiono także technikę wielowymiarowego skalowania.	
3.8.6	Rola inżyniera wiedzy w zadaniu akwizycji wiedzy	3-46
	Podkreślono odpowiedzialny i złożony charakter pracy inżyniera wiedzy. Omówiono składniki środowiska rozwoju bazy wiedzy. Przedstawiono charakterystykę sesji z ekspertem. Poruszono aspekty technik prowadzenia wywiadu i środków niewerbalnych. Wskazano na trudności związane z transformacją pozyskanej wiedzy do postaci reguł wnioskowania. Omówiono proces wnioskowania wykonywany przez ludzkiego eksperta. Wskazano na potrzebę respektowania faz i pojęć modelowania. Podkreślono konieczność doskonalenia wiedzy i rozważania dołączenia dodatkowego modułu. Przedstawiono zagadnienie walidacji i weryfikacji bazy wiedzy. Wskazano na możliwość rewizji i restrukturyzacji istniejącej bazy wiedzy, rolę przykładów testowych, aktualizacji i działań pielęgnacyjnych. Rozważono znaczenie środowiska konsultacji oraz dwa tryby korzystania z SE: konwersacyjny oraz bardziej aktywny tryb dialogu, określany terminem asystenta menedżera i jego charakterystykę. Wskazano niebezpieczeństwo pojawienia się wiedzy pozbawionej jednorodności, gdy ekspert i inżynier wiedzy posługują się niekompatybilnym aparatem pojęciowym.	
3.8.7	Moduł objaśniający w systemach eksperckich	3-51
	Przedstawiono klasyfikację pytań na które można uzyskać odpowiedź dzięki skorzystaniu z modułu objaśniającego przebieg procesu wnioskowania. Wskazano na możliwość analizy wariantów decyzyjnych.	

3.8.8 Agenda a architektura tablicowa **3-52**

Omówiono zadania i typologię agendy w SE. Zdefiniowano pojęcia architektury tablicowej oraz agenta. Rozważono przydatność podejścia kantowskiego, opartego na współlistnieniu różnych metodologii rozwiązań i reprezentacji oraz heglowskiego, postulującego syntezę wielu poglądów w realizacji architektury tablicowej i analizy eksperymentów.

3.9 Wnioskowanie oparte na przykładach. Ramy **3-54**

Przedstawiono założenia metody CBR. Podkreślono możliwości integracji z innymi metodami, w szczególności z techniką reprezentacji wiedzy w postaci ram. Zdefiniowano pojęcie i elementy ramy. Przedstawiono poziomy ram: widzialnej syntaktyki, semantyki, tematyczne, narracyjne. Omówiono podobieństwa między ramami a sieciami semantycznymi. Przedstawiono pojęcia obiektowości i enkapsulacji.

3.10 Tablice decyzyjne **3-57**

Przedstawiono podobieństwa i różnice między reprezentacją wiedzy za pomocą reguł wnioskowania oraz tablic decyzyjnych. Wyszczególniono elementy tworzące tablicę decyzyjną. Podano klasyfikację tablic decyzyjnych. Przedstawiono przepis na budowanie tablicy decyzyjnej. Omówiono problemy związane z występowaniem braku kompletności, spójności i poprawności.

3.11 Techniki odkrywania wiedzy w bazach danych **3-59**

Omówiono rolę baz danych w procesie przetwarzania danych ekonomicznych. Przedstawiono podejście odkrywania wiedzy w bazach danych (KDD). Wskazano na przesłanki zastosowania KDD. Zdefiniowano pojęcie wiedzy oraz etapy procesu KDD. Podkreślono przydatność hurtowni danych i narzędzi data mining. Przedstawiono klasyfikację metod odkrywania wiedzy: tradycyjne, metody statystyczne, drzewa decyzyjne, sieci neuronowe i podejście bayesowskie. Porównano podejście KDD oraz tradycyjnych technik analizy danych za pomocą języków dostępu do baz danych.

3.12 Algorytm genetyczny **3-62**

Przedstawiono definicję i koncepcję algorytmu genetycznego. Rozważono argumenty za wykorzystaniem operatora migracji. Wyszczególniono etapy w schemacie algorytmu. Przedstawiono pojęcia i charakterystykę funkcji przystosowania, operatora selekcji, krzyżowania, mutacji, chromosomu i sposoby jego kodowania. Omówiono pojęcie generycznego programowania genetycznego. Przedstawiono sposoby budowy populacji początkowej.

3.13 Czynniki wpływające na powodzenie SWD **3-67**

Przedstawiono zestawienie czynników, które wg doniesień literaturowych mają wpływ na powodzenie systemów wspomaganie decyzji w szczególności SE. Omówiono: zadowolenie użytkownika, wsparcie przez kierownictwo, cechy użytkownika końcowego, charakterystykę narzędzia budowy SE oraz znaczenie inicjatywy użytkowników końcowych.

3.14 Przyczyny niepowodzeń i przesłanki postulatu hybrydowości **3-69**

Przedstawiono wybrane przyczyny niepowodzeń we wdrożeniach systemów eksperckich. Omówiono uwarunkowania braku korzystania z narzędzia, braku wsparcia projektu, niewłaściwych metod pozyskiwania wiedzy, braku przydatności SE, doniosłości problemu, brak finansowego wsparcia projektu i niedoceniecie czynników społeczno-organizacyjnych na etapie implementacji. Przedstawiono przesłanki zastosowania podejścia hybrydowego, integracyjnego w celu uzyskania efektu synergicznego w SE. Wskazano na konieczność szczególnej uwagi podczas projektowania komponentu pozyskiwania wiedzy. Zaproponowano zintegrowanie metod symulacji i SE. Podano zalecenia dotyczące budowy SE według Turbana.

3.15 Projektowanie SWD **3-73**

Przedstawiono specyfikę projektowania SWD. Omówiono cykl życia SWD i motywacje działań podczas konstruowania SWD. Podkreślono rolę technologii systemów inteligentnych. Przedstawiono poglądy o wyzwaniach stojących przed przyszłościowymi SWD.

3.16	Problematyka projektowania interfejsu człowiek-komputer w systemach wiedzy	3-75
3.16.1	Definicja interfejsu człowiek – komputer w SWD	3-75
	Zdefiniowano pojęcie i cele interfejsu użytkownika. Wyróżniono rodzaje interfejsów.	
3.16.2	Modele interakcji z systemem przetwarzania wiedzy	3-75
	Przedstawiono architekturę współdziałania systemu przetwarzania wiedzy i interfejsu użytkownika. Wyróżniono komponent interpretacji czynności (zapytań i poleceń) oraz komponent wizualizacji. Przedstawiono charakterystykę i strukturę warstwy interfejsu. Przedstawiono wymiar adaptacyjny i ustalenia zgodności reprezentacji wiedzy między interfejsem oraz systemem przetwarzania wiedzy. Omówiono formalne podejścia Rathnama i Seeheim dotyczące przedstawienia struktury modułu interfejsu: komponent sterowania dialogiem, prezentacji i interfejsu aplikacji.	
3.16.3	System hybrydowy – wizualizacja komputerowa	3-77
	Przedstawiono pojęcie wizualizacji komputerowej. Wskazano jej niezbędność w warunkach dużej liczby danych. Wyszczególniono cele wizualizacji. Omówiono etapy wizualizacji danych i przyswajania przez człowieka: przetwarzanie liczb na obrazy oraz percepcję obrazów. Przedstawiono wizualizację jako fazę analizy danych znajdującą zastosowanie w analizach jakościowych. Omówiono funkcje pakietów wizualizacji. Wyszczególniono formy (elementy) graficzne wykorzystywane w wizualizacji. Opisano sposoby radzenia sobie z wielowymiarowością danych takie jak rzutowanie danych, mimiki twarzy czy technika równoległych osi.	
4	Systemy hybrydowe	4-1
4.1	Hybrydowość	4-1
4.1.1	Pojęcie hybrydowego systemu wspomaganie decyzji	4-1
	Omówiono pojęcie hybrydowego systemu bazowanego na wiedzy. Przedstawiono przesłanki projektowania i implementacji struktur hybrydowych. Wskazano na konieczność opracowania założeń SWD umożliwiających rozwiązywanie złożonych problemów decyzyjnych.	
4.1.2	Pojęcie modułowości	4-1
	Wskazano na przydatność architektury modularnej. Przedstawiono zalety delegowania zadań poszczególnym modułom. Omówiono potrzebę komunikacji między modułami.	
4.1.3	Przesłanki hybrydowości	4-2
	Przedstawiono argumenty za zastosowaniem hybrydowości jako podejścia konstruowania systemów wspomaganie decyzji. Wskazano na warunki pojawiania się efektu synergicznego. Podano definicję hybrydy. Podkreślono, że hybrydowość nie stanowi uniwersalnego rozwiązania.	
4.2	Cele opracowania hybrydowych systemów wspomaganie decyzji	4-4
	Przedstawiono klasyfikację sposobów hybrydyzacji SWD. Wskazano na dotychczasowe motywacje konstruowania hybryd SWD i technik sztucznej inteligencji. Podkreślono potrzebę wykorzystania mocnych stron poszczególnych implementowanych technik oraz osiągnięcia efektu synergicznego, w szczególności po połączeniu symulacji i systemu eksperckiego. Wskazano na kluczowe znaczenie uświadomienia słabych stron poszczególnych technik – np. optymalizacji, które powinny zostać usunięte dzięki rozwiązaniu hybrydowemu.	
4.3	Przykłady systemów wspomaganie decyzji rozszerzonych rozbudowanych o hybrydowość	4-6
	Przedstawiono koncepcję hybrydy SE i optymalizacji w zadaniu zarządzania inwestycjami oraz rolę podejścia modelowania strukturalnego. Omówiono zastosowanie hybrydy sieci neuronowej i modułu analizy dyskryminacyjnej w ocenie zagrożenia bankructwem. Podano przykład hybrydy sieci neuronowej i modułu zbiorów przybliżonych (zarówno o wzorcu horyzontalnym jaki dwuwymiarowym) mającej na celu przewidywanie niepowodzeń biznesowych na podstawie zestawu	

wskaźników finansowych. Omówiono zastosowanie hybrydy sieci neuronowej oraz modułu metody punktowej w ocenie wniosków kredytowych. Przedstawiono hybrydę algorytmu genetycznego oraz sieci neuronowych mającej na celu emulację metody punktowej zgodnie z zasadami inverse classification autorstwa Mannino i Koushika w ocenie wniosków kredytowych. Opisano także hybrydę algorytmu genetycznego i sieci neuronowej której celem było przewyższenie słabszych stron rachunku kosztów działań.

4.4 Zalety i wady poszczególnych technik 4-10

Omówiono zalety i wady poszczególnych technik wykorzystywanych jako elementy składowe struktur hybrydowych. Dokonano podziału technik sztucznej inteligencji na techniki twarde-kontekstowe i miękkie-bezkontekstowe oraz mieszane. Wskazano na potrzebę opracowania metod symbolicznego przetwarzania wiedzy nieustrukturalizowanej. Przedstawiono przesłanki wykorzystania w tym zadaniu narzędzi hybrydowych poprzez translację reprezentacji wiedzy. Wyszczególniono uwarunkowania wsparcia analizy i planowania finansowego w kontekście reprezentacji wiedzy według Zhanga.

4.5 Konekcyjność a symboliczna reprezentacja wiedzy 4-12

Przedstawiono przekonanie o dodatnim synergicznym efekcie połączenia dwóch lub więcej technik jako motywacji projektowania systemów hybrydowych. Omówiono zalety i wady konekcyjności i symbolicznej reprezentacji wiedzy. Podkreślono, że ze względu na złożoność strukturalną i komunikacyjną projektowanie inteligentnych systemów hybrydowych nie jest zadaniem łatwym. Wskazano, że współczesne narzędzia informatyczne nie wspierają tworzenia takich systemów a zatem powstałe projekty są unikatowymi artefaktami, nie są standardowe i notuje się wiele przykładów niepowodzeń w ich wdrożeniu. Omówiono problematykę hybryd sieci neuronowych i logiki rozmytej. Zwrócono uwagę, że zwykle równoległe rozwiązywanie problemu za pomocą dwu niezależnych metod jest bezpieczną metodą i nie zamyka decydentowi drogi podjęcia autonomicznej decyzji. Podkreślono, że celowe jest zapoznanie się ze wszystkimi wnioskami i raportami oraz przeprowadzenie dogrywki lub zastosowanie dodatkowej techniki nadzorującej pracę hybrydy.

4.6 Struktura hybrydy jako funkcja celów analizy 4-14

4.6.1 Uczenie maszynowe a uczenie ludzkie 4-14

Przedstawiono znaczenie modyfikacji struktur hybrydowych w zależności od charakteru zadania. Omówiono postulat Kilgourów dotyczący wykorzystania nisz przez poszczególne moduły. Wskazano argumenty za strukturami hybrydowymi odwzorowującymi postępowanie decydenta i rezygnację ze zbyt obszernych baz wiedzy na rzecz skupienia się na procesie rozumowania człowieka. Przedstawiono postulaty Malhotry za rozważaniem zamiast uczenia maszynowego i teorii przetwarzania informacji - wzorca uczenia ludzkiego i konstruktywistycznego podejścia teorii osobistego wartościowania, na którą składają się: konfuzja, hipoteza, działanie i porównanie. Przedstawiono kryteria taksonomiczne klasyfikacji struktur hybrydowych.

4.6.2 Klasyfikacje struktur hybryd wg kierunku przepływu informacji 4-15

Przedstawiono trzy sposoby integracji hybryd: strukturę sekwencyjną, działającą na zasadzie wspomaganiania oraz strukturę zagnieżdżoną. Omówiono ich elementy, etapy tworzenia, zastosowania oraz wbudowane techniki. Wskazano na rosnącą złożoność hybryd. Przedstawiono możliwości emulacji technik za pomocą funkcjonalnie złożonej pojedynczej techniki. Omówiono pojęcie wirtualnej struktury i wskazano na szczególną przydatność sieci neuronowych w emulacji.

4.6.3 Klasyfikacje struktur hybryd wg typu relacji 4-17

Przedstawiono podział strategii konstruowania systemów hybrydowych według Medskera. Omówiono charakterystykę struktur przechodzących przeobrażenie – działających w trybie aktywacji techniki lub systemu kaskadowego. Zaprezentowano cechy struktur luźno związanych i wskazano je jako najkrótszą drogę utworzenia hybryd. Omówiono tryb technik głównych i uzupełniających, tryb master-slave oraz równouprawnienia. Wskazano, że opisywana w dalszym ciągu pracy implementacja EKANWIN jest przykładem struktury luźno związanej. Omówiono zalety i trudności implementacyjne struktur ściśle związanych. Przedstawiono cechy i typy komunikacji wewnątrz struktur w pełni

zintegrowanych. Przedstawiono przykład w pełni zintegrowanej hybrydy sieci neuronowej i wnioskowania rozmytego.

4.6.4 Klasyfikacje struktur hybryd wg kryterium sterowania 4-20

Przedstawiono podział struktur dwuskładnikowych według Radosińskiego. Wyróżniono struktury hierarchiczne, rozłączne i homomorficzne. Omówiono rolę techniki głównej i uzupełniającej w strukturach hierarchicznych. Przedstawiono typ hybryd rozłącznych, kooperujących ze sobą przez przekazywanie struktur danych, nadzorowanych przez specjalny moduł lub użytkownika. Wskazano na ich przydatność w warunkach problemów dających podzielić się na podproblemy oraz posiadających wieloetapowe schematy rozwiązań. Omówiono hybrydy homomorficzne (jednolite). Wskazano na zróżnicowanie ich składników wyłącznie pod względem funkcjonalnym, ze względu na wykorzystywanie emulacji. Przedstawiono przykłady wymienionych klas hybryd.

4.6.5 Klasyfikacje struktur hybryd wg typu architektury wewnętrznej 4-22

Omówiono klasyfikację struktur hybrydowych wg Wermtera, która dotyczy przede wszystkim struktur w których zasadniczą rolę odgrywa sieć neuronowa: architektury jednolitej, transformacyjnej i modułowej. Przedstawiono założenia jednolitej architektury neuronową, w której sieć neuronowa emuluje technikę twardą – system wnioskowania oparty na regułach. Wskazano że zaletą tej architektury jest uzyskanie możliwości objaśniających i logicznego wnioskowania. Wyróżniono lokalny (przydatny w modelowaniu kognitywnym) i rozproszony (wyodrębnione przestrzennie struktury neuronów przedstawiają poszczególne pojęcia) typ architektury jednolitej. Omówiono architekturę opartą na wymianie reprezentacji wiedzy (transformacyjną). Wskazano na wyróżnienie techniki nadrzędnej i podrzędnej. Cechą charakterystyczną jest tu wprowadzenie wiedzy symbolicznej do sieci neuronowej. Przedstawiono także typ architektury modułowej, w którym poszczególne moduły współdziałają w strukturach luźno związanych, ściśle związanych lub w pełni zintegrowanych.

4.6.6 Klasyfikacje struktur hybryd wg zasad integracji 4-24

Omówiono znaczenie dekompozycji problemów na podproblemy. Przedstawiono hybrydy na bazie funkcji, komunikacji i struktury. Podkreślono że hybrydy na bazie funkcji działają na zasadzie aktywacji poszczególnych technik w razie potrzeby lub posiadają charakter sekwencyjny. Wskazano że projektowanie hybryd na bazie komunikacji, posiadających moduł transmisji danych jest najbardziej naturalne i umożliwia łatwą integrację dowolnej liczby technik. Nadają się one także do realizacji przetwarzania sekwencyjnego. Omówiono struktury polimorficzne (na bazie struktury) których cechą charakterystyczną jest emulacja funkcji i procesów na jednolitej platformie programistycznej (zwykle sieci neuronowej). Podkreślono pokrywanie się niektórych typów struktur w poszczególnych klasyfikacjach.

5 Założenia hybrydowego SWD Ekanwin 5-1

5.1 System wspomagania decyzji – Ekanwin – założenia 5-1

Sformułowano oczekiwania dotyczące inteligentnego SWD. Wyszczególniono i umiejscowiono w strukturze SWD pożądane funkcje i usługi oferowane przez system.

5.2 Przesłanki modułowości hybrydowego SWD Ekanwin 5-2

Wskazano na złożoność analizy ekonomicznej przedsiębiorstwa. Przedstawiono argumenty za zachowaniem modułowej struktury SWD i jej rozszerzeniem ze względu na potrzebę implementacji różnorodnych metod i technologii sztucznej inteligencji i przetwarzania danych. Wymieniono techniki które mogą znaleźć odzwierciedlenie w nowych zintegrowanych modułach. Wskazano na konieczność opracowania interfejsu komunikacji między modułami. Podkreślono kluczową rolę modułu modelu symulacyjnego dla realizacji SWD w architekturze klient-serwer. Wskazano, że moduł symulacyjny umożliwia pełniejsze zintegrowanie teorii, modelowania i eksperymentu.

5.3	Modułowość i hybrydowość SWD Ekanwin	5-4
	Przedstawiono możliwości kombinacji symulacji i technik inteligentnych (w szczególności systemu eksperckiego) w dwuskładnikowych luźno związanych hybrydach. Wskazano równorzędny charakter struktury i rolę modułu języka sterowania eksperymentem. Wyróżniono moduły systemu Ekanwin.	
5.3.1	Zastosowanie interfejsu hybrydowego	5-4
	Opisano hybrydowy (menu, formularze, drzewo decyzyjne, język sterowania eksperymentem) charakter interfejsu między użytkownikiem a zespołem modułów.	
5.3.2	Zadania modułu wizualizacji	5-5
	Przedstawiono rolę wizualizacji w prezentacji wyników przetwarzania wiedzy. Omówiono formy prezentacji obsługiwane przez moduł. Wskazano na możliwość graficznego konfigurowania eksperymentów, animacji i obserwacji zmian w bazie wiedzy. Wskazano na przesunięcie uwagi z algorytmu na bezpośrednie manipulowanie modelem i eksperymenty symulacyjne.	
5.3.3	Aspekty projektowe modułu wizualizacji	5-5
	Wskazano na pożądane cechy przejawiane przez projektanta interfejsu człowiek-komputer. Wskazano na niebezpieczeństwa związane z niską jakością interfejsów SWD. Podkreślono, że kluczową rolę w powodzeniu SWD odgrywają możliwości komunikacyjne oraz interfejs użytkownika	
5.3.4	Moduł wizualizacji w SWD Ekanwin	5-6
	Wskazano na czynnik zgodności na poziomie wewnętrznym i zewnętrznym jako warunku podwyższenia jakości SWD. Przedstawiono wymagania dotyczące możliwości adaptacyjnych dialogu względem stopnia zaawansowania użytkownika. Omówiono potrzebę interaktywności, wykorzystania grafiki, konstruowania zestawów ekonomicznych, wnioskowania eksperckiego, parametryzacji reguł, samodzielnej budowy bazy wiedzy i planu eksperymentu. Podkreślono zgodność założeń interfejsu systemu Ekanwin z klasyfikacją stopni adaptacyjności interfejsu wg Lianga (graficzne prowadzenie dialogu, zarządzanie informacją przez użytkownika oraz samoadaptacyjność). Omówiono charakterystykę typów użytkowników SWD wg Rathnama: projektantów, analityków i członków kierownictwa. Zwrócono uwagę na możliwość dostosowania do stylu rozumowania analityka poprzez poszczególne funkcje interfejsu.	
5.4	Cele hybrydowego systemu wspomaganie decyzji Ekanwin	5-8
	Omówiono dwa tryby wsparcia procesu podejmowania decyzji oferowane przez system Ekanwin. Przedstawiono sterowanie eksperymentem za pomocą języka LEKS w celu optymalizacji parametrów decyzji, otrzymywanie raportów z modułu systemu eksperckiego oraz wykorzystanie graficznego, interaktywnego środowiska konfiguracji eksperymentów. Wymieniono techniki, których moduły mogą być zintegrowane z modułem symulacji w celu uzyskania efektu synergicznego: system ekspercki, wizualizacja, planowanie eksperymentu, proceduralny język skryptowy, optymalizacja itd. Wskazano na hybrydowy charakter modelu przedsiębiorstwa implementowany w SWD Ekanwin.	
6	Projektowanie interfejsu systemu Ekanwin dla potrzeb wizualizacji i wnioskowania	6-1
6.1	Korzystanie z graficznego interfejsu użytkownika	6-1
6.1.1	Moduł wizualizacji i moduł wnioskowania eksperckiego - opis korzystania z interfejsu	6-1
	Omówiono znaczenie analiz wariantowych oraz metody prób i błędów. Wskazano na możliwość sterowania wartościami komórek macierzy zmiennych decyzyjnych za pomocą elementów ekranowych typu suwak. Przedstawiono istotę realizacji eksperymentów typu what-if i zastosowanie graficznego interfejsu użytkownika. Przedyskutowano propozycje konfiguracji pulpitu sterowniczego macierzy zmiennych decyzyjnych. Omówiono związek przyczynowo-skutkowy między analizami porównawczymi i postaciami scenariuszy eksperymentu. Przedstawiono zastosowanie animacji (automatycznej i ręcznej) w procedurach poszukiwania rozwiązań optymalnych. Podkreślono rolę automatycznego rejestrowania i dokumentowania czynności badawczych wykonywanych przez operatora-analityka. Przedyskutowano propozycje strategii poszukiwania rozwiązań. Przedstawiono	

wygląd graficznego interfejsu użytkownika (efektu działania modułu wizualizacji) w różnych konfiguracjach i ustawieniach opcji dotyczących modułów przetwarzających informacje i wiedzę.

6.1.2 Funkcja raportowania systemu Ekanwin

6-6

Przedstawiono funkcję automatycznego raportowania dotyczącą eksperymentów wykonywanych w systemie Ekanwin. Omówiono jej integrację z modułem wnioskowania eksperckiego i działaniami operatora na poziomie pulpitu sterującego. Raportowanie polega na dokumentowaniu informacji o aktywowanych bazach wiedzy, zbiorach (bazach) wskaźników w kontekście architektury tablicowej i różnorodnych źródeł wiedzy, wartościach zmiennych decyzyjnych oraz wynikach (diagnozach) wnioskowania eksperckiego opartego na regułach. Raport odzwierciedla także styl postępowań analityczno-decyzyjnych reprezentowany przez operatora systemu (strategie, skupianie uwagi, rozważane wskaźniki, kryteria optymalizacji i zatrzymania). Przedstawiono przygotowanie graficznego interfejsu użytkownika do uruchomienia pełnego trybu raportowania oraz przykładowy raport.

6.2 Koncepcja wnioskowania i bazy wiedzy

6-11

Przedstawiono etapy konstrukcji bazy wiedzy w systemie Ekanwin. Przedstawiono sposób integracji biblioteki programistycznej regułowego systemu wnioskowania typu shell – CLIPS z hybrydowym SWD. Wskazano za zewnętrzny charakter definiowania wskaźników ekonomicznych z poziomu MS Excel. Opisano fazy automatycznego wypełniania (ukonkretnienia) bazy faktów i konstruowania bazy reguł w trakcie eksperymentów symulacyjnych wykonywanych w trybie online. Przedstawiono odpowiednie przykłady plików zawierających bazy na poszczególnych etapach tworzenia bazy wiedzy. Omówiono zasady prezentacji wyników wnioskowania i filtrowania bazy faktów przed wysłaniem do modułu wizualizacji i raportowaniem. Przedstawiono znaczenie poszczególnych ustawień i opcji dla uzyskiwanych wnioskowań (parametryzacja reguł wnioskowania i korzystanie z architektury tablicowej).

6.3 Moduł metod analitycznych. Obsługa wskaźników ekonomicznych

6-22

Przedstawiono korzystanie ze sprawozdań finansowych, hierarchicznej bazy danych (drzewa decyzyjnego) oraz relacyjnej bazy danych (definicji wskaźników) przez moduły wizualizacji i systemu eksperckiego. Omówiono podmoduł interpretacji definicji wskaźników ekonomicznych jako metawiedzy. Przedstawiono technikę translacji informacji pochodzących ze struktury sprawozdań generowanych przez moduł symulacyjnego modelu przedsiębiorstwa oraz relacyjnej bazy danych (wskaźników) i wsparcie translacji przez graficzny interfejs użytkownika. Przedstawiono sposoby korzystania z bazy wskaźników: udostępnienia gotowego zestawu – utworzonego dla realizacji określonego z góry celu oraz interaktywnej edycji za pomocą interfejsu użytkownika. Omówiono znaczenie uznanego, podręcznikowego zestawu wskaźników. Przedstawiono założenia konstrukcji wskaźników ekonomicznych w systemie Ekanwin. Wskazano argumenty za wykorzystywaniem ograniczonego liczbowo zestawu wskaźników i w razie potrzeby przełączaniem aktywnych zestawów. Umożliwia to równocześnie realizację założeń architektury tablicowej systemu eksperckiego.

6.4 Moduł systemu eksperckiego – charakterystyka języka reprezentacji wiedzy

6-25

6.4.1 Środowisko CLIPS jako przykład systemu shell

6-25

Przedstawiono architekturę bazy wiedzy w systemie eksperckim typu shell – CLIPS. Przedstawiono techniki pozyskiwania wiedzy w systemie CLIPS oraz założenia rozbudowy architektury o podejście tablicowe.

6.4.2 Język reprezentacji wiedzy CLIPS

6-26

Omówiono strategię wnioskowania zastosowaną w systemie CLIPS. Przedstawiono zasady deklarowania faktów i reguł oraz sposoby dołączenia faktów do bazy wiedzy na przykładzie wskaźników ekonomicznych.

7	Koncepcja, projekt i zastosowanie języka sterowania eksperymentem LEKS	7-1
7.1	System wspomagania decyzji – Ekanwin – moduł planowania i sterowania eksperymentem symulacyjnym – koncepcja projektowania	7-1
7.1.1	Koncepcja języka LEKS	7-1
	Przedstawiono założenia dotyczące funkcji interpretowalnego, algorytmicznego języka skryptowego służącego sterowaniu eksperymentem. Zdefiniowano założenia projektowe modułu działającego zgodnie z architekturą typu klient-serwer, zrealizowanego jako warstwa między modelem symulacyjnym a interfejsem użytkownika. Wyszczególniono zadania modułu i dostarczanych przez niego usług. Dokonano klasyfikacji funkcji języka. Zaprezentowano przykład graficznej postaci interfejsu działającego w trybie wykonywania programu w języku LEKS. Wyszczególniono cechy graficznego interfejsu użytkownika (praca krokowa, funkcje debugowania, podgląd wartości zmiennych programowych). Omówiono założenia zbioru rozkazów języka.	
7.1.2	Perspektywa zastosowań języka LEKS	7-4
	Zaprezentowano ogólne założenia algorytmu optymalizacyjnego naśladującego (w określonej kolejności) poszczególne czynności operatorsko-analityczne zgodnie z konstruktywistycznym podejściem teorii osobistego wartościowania. Przedstawiono zarys obszarów zastosowań połączonych technik optymalizacji i symulacji w rozwiązywaniu problemów zarządzania przedsiębiorstwem.	
7.1.3	Wprowadzenie do badań wykonywanych za pomocą języka LEKS	7-5
	Omówiono algorytmiczne cechy strukturę planów eksperymentów formułowanych za pomocą języka LEKS. Przedstawiono propozycje wykorzystania wyników otrzymanych w rezultacie uruchomienia serii eksperymentów. Omówiono sposoby wsparcia i zintegrowania bardziej złożonych fragmentów składni języka z graficznym interfejsem użytkownika. Wyszczególniono techniki analityczne i metody optymalizacyjne korzystające z modułu sterowania eksperymentem, zaprezentowane w dalszej części pracy.	
7.1.4	Elementy języka LEKS	7-6
	Przedstawiono omówienie instrukcji języka LEKS pogrupowanych wg funkcji.	
7.2	Przykłady badań wykonywanych za pomocą języka LEKS wspomaganych interfejsem użytkownika	7-11
7.2.1	Zastosowanie metody bisekcji	7-11
	Sformułowano program badawczy dotyczący problemu optymalizacji funkcji wielu zmiennych poprzez dekompozycje na podproblemy w których zastosować można technikę optymalizacji funkcji jednej zmiennej – bisekcji dziedziny funkcji. W zaprezentowanym przykładzie optymalizowany był zysk netto symulowanego przedsiębiorstwa na koniec badanego okresu (trzy lata) a zmiany dotyczyły wartości cen (trzech produktów). Przedstawiono formalny opis i schemat algorytmu bisekcji. Opracowano graficzno-macierzowy sposób zapisu planu eksperymentu i sformułowano badane strategie poszukiwania rozwiązania optymalnego. Przedyskutowano uwarunkowania rozbudowy strategii (planu eksperymentu) o poszukiwanie optymalnych wielkości zamówień produkcyjnych oraz wykorzystanie możliwości interaktywnej (skokowej) modyfikacji wartości zmiennych decyzyjnych, animacji i analizy wyników wnioskowania eksperckiego. Wyróżniono problemy elementarne związane z doborem punktu startowego, dziedziny poszukiwań i warunku zatrzymania algorytmu oraz strategii (kolejności optymalizacji zmiennych decyzyjnych). Przedstawiono tekst źródłowy programu optymalizacyjnego w języku LEKS skonstruowanego w postaci programu nadzorującego oraz zestawu plików – procedur elementarnych (dotyczących optymalizacji jednej komórki macierzy zmiennych decyzyjnych) wywoływanych z poziomu programu głównego, z odpowiednimi wartościami parametrów.	

7.2.2 Metoda bisekcji – wyniki eksperymentów i wnioski

7-23

Wyniki eksperymentów przedstawiono w tabelach. W celu wyboru najodpowiedniejszej strategii obliczono wskaźnik jakości rozwiązania i wyznaczono najodpowiedniejszą grupę strategii. Przedstawiono wykresy badanej funkcji (zysku netto) dla najlepszego z wyznaczonych rozwiązań i porównano z wynikami prognozowanymi dla warunków początkowych. Wybrano trzy najlepsze rozwiązania, które posłużyły jako punkty startowe przy ponownym uruchomieniu programu metody bisekcji (planu eksperymentu), jednak tym razem zmieniano wielkość zamówień produkcyjnych. Ponownie wybrano najlepsze trzy rozwiązania, które posłużyły jako punkty startowe przy kolejnym uruchomieniu programu implementującego metodę bisekcji (planu eksperymentu), jednak tym razem zmieniano wielkość zamówień surowców. Sformułowano wnioski, przedyskutowano uwarunkowania optymalizacji o dłuższym horyzoncie czasowym i zaproponowano modyfikację postępowania optymalizacyjnego o wykorzystanie interaktywnych elementów ekranowych (suwaki sprzężone z systemem eksperckim).

7.2.3 Wsparcie metody bisekcji interaktywnym interfejsem użytkownika

8-27

Sformułowano modyfikację metody bisekcji polegającą na uwzględnieniu wyników (dostępnych w sprawozdaniach finansowych) takich jak zapasy surowców i wyrobów gotowych, wskaźniki obrotowości, wielkość sprzedaży itp., wyniki diagnoz prezentowanych przez moduł systemu eksperckiego i w razie potrzeby zmianę wartości komórek macierzy zmiennych decyzyjnych leżących poza podmacierzą cen. Wyniki i omówienie eksperymentów przedstawiono w postaci tabeli zawierającej wykresy zysku netto. Sformułowano wnioski.

8 Zastosowanie języka sterowania eksperymentem LEKS. Analiza czynnikowa

8-1

8.1 Rola eksperymentu w badaniach naukowych

8-1

Omówiono założenia hipotetyczno-dedukcyjnej metody badania obiektu. Zdefiniowano pojęcia obserwacji, prognozy i eksperymentu. Przedyskutowano znaczenie analizy wyników badań, w szczególności potrzebę powtarzalności i zgodności z teorią.

8.2 Planowanie doświadczeń. Systematyka planów eksperymentu

8-2

Omówiono pojęcie planowania i planu eksperymentu. Wyszczególniono czynności przygotowawcze wykonywane przed eksperymentem. Przedstawiono typologię planów eksperymentu na podstawie kryteriów: zbioru wartości zmiennych wejściowych (poziomy planów), zastosowanej techniki rozbudowy planu (kompozycyjne, ortogonalne, sympleksowe itd.), liczby doświadczeń (całkowite, ułamkowe itd.).

8.3 Planowanie dwupoziomowe – przykłady zastosowań

8-5

8.3.1 Założenia metody Boxa-Wilsons dla potrzeb dwupoziomowego planu eksperymentu

8-5

Przedstawiono strukturę danych wejściowych, postać funkcji celu, sposób zapisu przebiegu eksperymentu oraz algorytm wyznaczania rozwiązania optymalnego w metodzie Boxa-Wilsons.

8.3.2 Zastosowanie procedury optymalizacji wg metody Boxa-Wilsons

8-6

Wyróżniono etapy algorytmu. Sformułowano pierwsze zagadnienie problemowe: plan dwupoziomowy, eksperyment całkowity o dwu wejściach. Optymalizacja dotyczyła wartości zysku netto, zmianom podlegały ceny zbytu dwu wyrobów; określono parametry punktu startowego i długość kroków próbnych, podano postać funkcji charakterystyki ekstremalnej oraz ograniczenia nałożone na wartości zmiennych. Przedstawiono wykresy wartości funkcji w badanej dziedzinie wartości zmiennych oraz hiperpłaszczyzny ustalającej kierunek poszukiwań, wyznaczonej w wyniku działania procedury. Omówiono standaryzację układu współrzędnych zadania i poszczególne kroki algorytmu oraz macierz wejść planu eksperymentu. Wyznaczono parametry funkcji regresji a następnie iteracyjnie określono punkt ekstremalny. Przedstawiono tabelaryczne i graficzne zestawienie wyników osiągniętych w poszczególnych krokach algorytmu. Następnie zmodyfikowano założenia algorytmu wprowadzając liniowe ograniczenia na wartości zmiennych decyzyjnych. Podobnie jak poprzednio uruchomiono program w języku LEKS i wskazano że nowy znaleziony punkt ekstremalny

leży na granicy przeszukiwanego obszaru i nie pokrywa się z kierunkiem poszukiwania wyznaczonym przez parametry funkcji regresji. Wynika to z postaci funkcji celu. Trzecie zagadnienie problemowe dotyczyło sformułowania planu dwupoziomowego, całkowitego, o trzech wejściach (ceny trzech wyrobów). Wykonano eksperyment i przedstawiono zestawienie tabelaryczne otrzymanych wyników. Czwarte zagadnienie problemowe dotyczyło planu dwupoziomowego, całkowitego, dwuwejściowego (ceny zbytu i wielkości zamówień wyrobu). Przedstawiono wykresy wartości funkcji w badanej dziedzinie wartości zmiennych oraz hiperpłaszczyzny ustalającej kierunek poszukiwań, wyznaczonej w wyniku działania procedury. Przedstawiono tabelaryczne i graficzne zestawienie wyników osiągniętych w poszczególnych krokach algorytmu.

8.4 Badanie przebiegu wartości zmiennych zależnych z wykorzystaniem języka LEKS 8-20

8.4.1 Wyznaczenie trójwymiarowego wykresu wartości funkcji celu za pomocą programu w języku LEKS 8-20

Omówiono założenia zadania sporządzenia zestawienia tabelarycznego wyniku finansowego netto w funkcji cen wyrobów. Przedstawiono trójwymiarowy wykres będący wynikiem serii eksperymentów inkrementacyjnych oraz tekst źródłowy procedury w języku LEKS.

8.4.2 Zastosowanie języka LEKS w problemie wyznaczania hiperpowierzchni progu rentowności 8-23

Sformułowano założenia zadania wyznaczenia trójwymiarowego odwzorowania progu rentowności w funkcji zleceń produkcyjnych poszczególnych wyrobów. Omówiono technikę sterowanego krokowego (inkrementacyjnego) przeszukiwania przedziału wartości zmiennej zależnej. Przedstawiono wykresy wyników eksperymentu oraz zweryfikowano wyniki za pomocą badania wyniku finansowego dla bliskiego otoczenia miejsca zerowego.

8.5 Procedura optymalizacyjna w języku LEKS - metoda Hooke'a-Jeevesa 8-29

Przedstawiono założenia optymalizacyjnej metody Hooke'a-Jeevesa jako przykładu metody przeszukiwania prostego. Wyszczególniono kroki algorytmu. Omówiono zastosowanie metody w celu rozwiązania zadania optymalizacji wartości podzbioru macierzy zmiennych decyzyjnych. Dokonano dekompozycji zadania na trzy etapy i przedstawiono założenia zmodyfikowanej, uproszczonej metody Hooke'a-Jeevesa. Przedstawiono program źródłowy w języku LEKS, zamieszczono wyznaczone wyniki (optymalne wartości cen wyrobów) oraz wykres przebiegu wyniku finansowego dla optymalnych wartości argumentów.

8.6 Procedura optymalizacyjna w języku LEKS – hybryda symulacji, języka sterowania eksperymentem i algorytmu genetycznego 8-37

Przedstawiono założenia zadania optymalizacji wielkości zamówień surowców. Wskazano na możliwości zastosowania graficznego interfejsu użytkownika w celu ustalenia punktu początkowego. Omówiono założenia algorytmu genetycznego zastosowanego w rozwiązywaniu zadania. Przedstawiono wykres wyniku finansowego w zależności od wielkości zamówienia surowców. Wykonano serię eksperymentów i porównano efektywność algorytmu bez i z operatorem mutacji. Przedstawiono tekst źródłowy algorytmu genetycznego w języku LEKS.

9 Dodatki 9-1

9.1 Informacje o wybranych ośrodkach uniwersyteckich prowadzących badania nad hybrydowymi systemami wspomaganie decyzji 9-1

9.2 Wybrane konferencje dotyczące systemów hybrydowych 9-2

9.3 Wybrana literatura poruszająca zagadnienie hybrydowości 9-3

9.4 Ewolucja systemu Ekanwin 9-4

9.5 Opis symulatora Ek_An i przykłady generowanych sprawozdań finansowych 9-5

9.6 Opis języka sterowania eksperymentem LEKS 9-12

Wstęp

Praca dotyczy wspomaganej komputerowo analizy ekonomicznej przedsiębiorstwa przemysłowego. W analizie ekonomicznej korzysta się z metod ilościowych i jakościowych. Podział metod wynika ze stosowania zróżnicowanych technik analitycznych i procedur wnioskowania, formy i źródeł wykorzystywanych informacji oraz postaci i przeznaczenia wiedzy otrzymanej w wyniku zastosowania metod. Oba typy metod są obecnie wspomagane komputerowo. Sformułowano tezę, że połączenie, a nawet interakcja różnorodnych technik (konwencjonalnych i tzw. sztucznej inteligencji) zwiększy możliwości poznawcze i wartość stosowanych procedur podejmowania decyzji.

Dotychczas brakuje jednak uznanych zasad integracji poszczególnych technik stanowiących aparat wyszczególnionych metod. Z tego powodu efekt synergiczny nie jest wystarczająco rozpoznany, mimo że hybrydowe systemy wspomagania decyzji cechują się znaczną złożonością (funkcjonalną, strukturalną, informacyjną, technologiczną, metodologiczną), która jest odpowiedzią na wyzwanie złożoności zagadnienia analizy ekonomicznej przedsiębiorstwa. Także symulacja komputerowa jest zwykle traktowana jako technika obliczeniowa, odseparowana od wnioskowania i przetwarzania symbolicznego. Co gorsza, symulacja jest pozbawiona wymiaru optymalności.

Należy zatem zaproponować rozwiązania technologiczne zapewniające połączenie podstawowych technik analitycznych, stosowanych do tej pory oddzielnie: dynamicznej analizy wskaźnikowej, wnioskowania eksperckiego opartego na regułach, symulacji komputerowej, analizy wrażliwości, grafiki biznesowej, interaktywnego interfejsu użytkownika, sterowania eksperymentem, planowania eksperymentu, algorytmów genetycznych, algorytmów optymalizacji. W ten sposób spełnione zostaną współczesne wymagania dynamiczności, konkretności, elastyczności i operatywności stawiane przed analizami ekonomicznymi.

Problematyka struktur hybrydowych nie jest wystarczająco opracowana. W literaturze przedmiotu przedstawiono jednak szereg kwestii badawczych do rozwiązania:

- W jakim celu i w jaki sposób łączyć poszczególne techniki sztucznej inteligencji aby powstały systemy hybrydowe zwiększające możliwości analityczne w zakresie analizy ekonomicznej przedsiębiorstwa?
- W jaki sposób wykorzystywać, na podstawie jakich kryteriów grupować i jak oceniać wartości wskaźników ekonomicznych stosowanych w analizie ekonomicznej? Jak wykorzystać te informacje w ocenianiu kategoriami rozwoju (kryterium dynamiczności) i w analizie porównawczej?
- Czy możliwe jest opracowanie zautomatyzowanej procedury oceny kondycji finansowej przedsiębiorstwa respektującej zróżnicowane potrzeby analityczne i sposób podejmowania decyzji przez analityka bądź decydenta (kryterium elastyczności) ? W jaki sposób filtrować informacje uzyskiwane w toku wnioskowania eksperckiego i prezentować je analitykowi?

- W jaki sposób zastosować symulacyjny model przedsiębiorstwa i wykorzystać zalety metody symulacji dynamicznej w procesie podejmowania decyzji? Jak zintegrować technikę analizy wrażliwości i eksperyment symulacyjny? Czy można dokumentować przebieg analizy wrażliwości (kryterium konkretności) dla potrzeb opracowania reguł podejmowania decyzji?
- Jakie powinny być zasady konstruowania interfejsu między modułami implementującymi poszczególne techniki analityczne, sztucznej inteligencji a modelem symulacyjnym przedsiębiorstwa reprezentującym istniejący system rejestrowania zdarzeń gospodarczych? Czy można sformułować zasady konfiguracji interfejsu dla konkretnych zadań badawczych? Czy możliwe jest równoczesne korzystanie z wielu modułów w trakcie analizy wrażliwości?
- W jaki sposób sterować przebiegiem eksperymentu w środowisku hybrydowego systemu wspomaganie decyzji? Jak zastosować techniki planowania eksperymentu symulacyjnego? W jaki sposób posłużyć się założeniami teorii osobistego wartościowania? Czy możliwe jest opracowanie wytycznych dotyczących doświadczalnych procedur badawczych mających na celu poszukiwanie optymalnych rozwiązań w dziedzinie produkcji, zbytu, finansowania i inwestycji przedsiębiorstwa (kryterium operatywności)? Czy możliwe jest opracowanie ogólnej, uniwersalnej metody implementowania algorytmów optymalizacji bez, jak i z ograniczeniami, zarówno konwencjonalnych (gradientowych, bezgradientowych) jak i kierowanego przeszukiwania losowego (np. genetycznych). Jakie cechy powinna mieć taka metoda? Jakie są jej związki z analizą wrażliwości?
- W jaki sposób skonfigurować procedury komputerowej wizualizacji danych i informacji z systemu rachunkowości przedsiębiorstwa? Jak wykorzystać informacje z systemu rachunkowości przy generowaniu bazy wiedzy dla regułowego systemu eksperckiego?
- W jaki sposób integrować algorytmy losowego przeszukiwania rozwiązań (alg. genetyczne) i konekcyjną reprezentację wiedzy?

W niniejszej pracy przedstawiłem wyniki badań nad uwarunkowaniami integracji technik inteligentnych dla potrzeb analizy ekonomicznej w hybrydowych systemach (strukturach) wspomaganie decyzji. Celem była odpowiedź na pytanie czy „hybrydyzacja” SWD udostępni nieznane dotychczas możliwości prowadzenia analiz ekonomicznych. Przed przystąpieniem do opracowania i implementacji nowej metodyki systemów wspomaganie decyzji należy odpowiedzieć na szereg pytań. Do najważniejszych należą następujące:

- W jaki sposób, za pomocą jakich metod ocenia się kondycję ekonomiczną przedsiębiorstwa przemysłowego ze szczególnym uwzględnieniem poszczególnych obszarów działalności?
- Jakie techniki komputerowe są stosowane w decyzyjnej analizie ekonomicznej?
- Jakie są uwarunkowania zastosowań symulacji komputerowej w analizie ekonomicznej?

- Czym są hybrydowe SWD? Jakie są przesłanki łączenia technik komputerowych stosowanych w analizie ekonomicznej przedsiębiorstw; jakie typy hybryd można wyróżnić? Jakie są dotychczasowe ograniczenia systemów hybrydowych?
- W jaki sposób konstruować hybrydowe systemy wspomaganie decyzji; jak łączyć poszczególne techniki analityczne?
- W jaki sposób, w jakich problemach decyzyjnych wykorzystać stworzoną implementację hybrydowego SWD, jak ocenić efekty jego zastosowania?

W celu osiągnięcia postawionego celu wykonałem następujące czynności:

- Przeprowadziłem studia literaturowe pomocne w udzieleniu odpowiedzi na postawione kwestie.
- Opracowałem zasady konstrukcji i implementacji hybrydowego inteligentnego systemu wspomaganie decyzji, na który składają się następujące moduły i techniki: symulacja + tablicowy regułowy system ekspercki + planowanie eksperymentu + algorytm genetyczny + wizualna interaktywna symulacja (Visual Interactive Simulation - VIS) + algorytmy optymalizacji + grafika biznesowa.
- Określiłem typy zagadnień (problemów) i technik ich rozwiązywania, w których można wykorzystać hybrydowy SWD zrealizowany w formie modułów programowych. Wykonałem także szereg eksperymentów.

Wyniki programu badawczego są następujące: dokonano przeglądu współczesnych technik komputerowych mogących znaleźć zastosowanie w analizie finansowej przedsiębiorstwa, opracowano koncepcję modułowej integracji dla potrzeb hybrydowego, inteligentnego systemu wspomaganie decyzji, opracowano metody integracji poszczególnych technik i modelowania symulacyjnego. Charakterystyka poszczególnych rozdziałów pracy zawarta została w spisie treści.

Uznałem że zastosowanie nowego hybrydowego podejścia dla potrzeb analizy ekonomicznej przedsiębiorstwa wymaga opracowania rozwiązań informatycznych ogólnego zastosowania (platformy informatycznej). Kluczowymi elementami hybrydy powinny być: moduł modelu symulacyjnego badanego obiektu (dziedzinowy, ekonomiczny), moduł sterowania eksperymentem (ogólnego zastosowania tzn. zdolny do implementacji technik inteligentnych, optymalizacyjnych, planowania eksperymentu) oraz moduł interfejsu komunikacji między aparatami pojęciowymi. Postuluję że należy zintegrować z hybrydą ogólnego zastosowania moduły przetwarzające dane, informacje, wiedzę, procesy, decyzje oraz wykorzystać metodologię dotyczącą badania podmiotu gospodarczego. Wśród nich szczególnie należy wyróżnić użyteczność języka sterowania eksperymentem zgodnego z podejściem guiding cognition. Dzięki opracowaniu i zastosowaniu podejścia hybrydowego w

konstrukcji narzędzia (środowiska informatycznego) wykonano eksperymenty symulacyjne, których wykonanie było dotychczas niemożliwe. Nakreśliłem perspektywy wykorzystania hybrydowego systemu analizy ekonomicznej w badaniach, dydaktyce i praktyce menedżerskiej.

Konkretnym efektem pracy jest komputerowe środowisko (platforma informatyczna) eksperymentów z hybrydowym inteligentnym systemem wspomagania decyzji w zakresie analizy finansowej przedsiębiorstwa, implementującym metody analizy takie jak analiza wskaźnikowa i analiza porównawcza oraz grafika biznesowa. Hybryda wyposażona jest w moduły implementujące technologie: regułowe systemy eksperckie z architekturą tablicową, interaktywny graficzny interfejs użytkownika oraz język sterowania eksperymentem za pomocą którego możliwe jest zintegrowanie praktycznie dowolnej techniki optymalizacyjnej. Moduł języka planowania eksperymentu pozwala na prowadzenie analiz czynnikowych z wykorzystaniem dowolnych planów eksperymentu wraz z procedurami poszukiwania rozwiązań optymalnych.

W pracy przedstawiłem zastosowania metody hybrydowej w następujących problemach analizy finansowej (badałem zachowanie się wyniku finansowego).

- Technika symulacji inkrementacyjnej w badaniu wpływu parametrów decyzji (cen) na podstawie całkowitego planu eksperymentu
- Dynamiczna symulacja inkrementacyjna dla potrzeb wyznaczenia trójwymiarowej powierzchni progu rentowności (w funkcji zleceń produkcyjnych) na podstawie całkowitego planu eksperymentu z poszukiwaniem miejsc zerowych
- Optymalizacja za pomocą metod Hooke'a-Jeevesa i bisekcji dla potrzeb poszukiwania zadowalającej strategii cenowej i optymalnej macierzy cen zbytu. Wykorzystałem także raporty z systemu eksperckiego
- Wykorzystanie techniki planowania eksperymentu Boxa-Wilsona dla potrzeb wyznaczenia optymalnych cen zbytu przy ułamkowym planie eksperymentu i weryfikacja na podstawie pełnego planu eksperymentu. Rozważałem przypadki z nieaktywnymi i aktywnymi ograniczeniami na wartość cen
- Poszukiwanie optymalnych wielkości zamówień surowców dla funkcji celu z wieloma ekstremami lokalnymi. Zaprojektowałem i zastosowałem algorytm genetyczny zaimplementowany w języku sterowania eksperymentem.

Wśród wyników zastosowania metody wyróżnić można:

- Uzyskanie możliwości wszechstronnej analizy ekonomicznej, swobodnego formułowania problemów oraz doboru metod rozwiązania i oceny wyników z obszaru analizy ekonomicznej (architektura otwarta, szczegóły w pracy) za pomocą SWD
- Rozszerzenie symulacji komputerowej o planowanie eksperymentu
- Wyznaczenie relacji między zmiennymi w modelu symulacyjnym przedsiębiorstwa, które nie były możliwe do odkrycia do tej pory
- Rozszerzenie analiz ekonomicznych o tekstowe raporty wykorzystujące tablicową architekturę systemu eksperckiego
- Zintegrowanie technik inteligentnych i grafiki biznesowej z techniką analizy wrażliwości

1 Analiza decyzyjna

1.1 Określenie przedmiotu badań. Systematyka podejmowania decyzji oraz znaczenie kapitału intelektualnego

Decyzje menedżerskie, wg Szapiro [293] posiadają następujące cechy (niekoniecznie wszystkie):

1. Stanowią wynik procesu czyli [52] strukturalnego, mierzalnego zbioru działań wewnątrz organizacji zaprojektowanego dla osiągnięcia określonych skutków ekonomicznych mającego prowadzić do osiągnięcia celu.
2. Towarzyszy im występowanie wielu kryteriów oceny stopnia osiągnięcia tego celu.
3. Decyzje podejmowane są w sytuacji niepewności związanej z przyszłym stanem środowiska (a także obiektu podlegającego decyzjom).

Współczesna generacja **systemów wspomagania decyzji**, jak wskazuje Keen [144], dzięki postrzeganiu danych wg zróżnicowanych perspektyw (w tym danych niepewnych, niekompletnych) i możliwością radzenia sobie z jakościową reprezentacją wiedzy jest w stanie przetworzyć je w informacje przydatne w procesie podejmowaniu decyzji. Noblista Simon [275] uważa podejmowanie decyzji za główną funkcję organizacji, podkreślając istnienie szerokiego spektrum problemów decyzyjnych: od łatwych, dających się rozwiązać za pomocą „zaprogramowanych” działań (powtarzalność, rutynowość, dobrze zdefiniowana struktura) do trudnych, nie dających się „zaprogramować” (mających źle zdefiniowaną strukturę, konieczne opracowanie nowatorskich metod rozwiązań). Podejmowanie decyzji [72] nie jest łatwym zadaniem, gdyż wysoka jakość decyzji pożądana jest w celu osiągnięcia większej wydajności i lepszych wyników organizacji oraz szybszemu rozwiązywaniu problemów menedżerskich. Jednak problemy te zazwyczaj są złożone, źle ustrukturalizowane, decydent musi respektować zdanie wielu zainteresowanych podmiotów i grup. Najczęściej charakteryzują się występowaniem niepewności - zarówno na etapie interpretacji zebranych danych źródłowych jak i oczekiwanych wyników realizowanego wariantu decyzyjnego. Dlatego też czynione są próby opracowania metodologii mierzenia i badania nieuchwytnych dotychczas wymiarów działalności firmy takich jak:

1. Personel - jego umiejętności oraz doświadczenie.
2. Relacje z klientami – baza danych klientów, marka i zaufanie do niej.
3. Procesy – przebieg działalności wewnątrz organizacji oraz sprzężenia zwrotne z otoczeniem.
4. Odnowienie i rozwój – czy przedsiębiorstwo wykazuje wzrost, czy rokuje nadzieje na rozwój oraz czy docenia przygotowania się na przyszłość i prowadzenie badań.

Powyżej wymienione cztery wymiary uzupełniane są w modelu **kapitału intelektualnego** [149] o piąty (finansowy), którego odwzorowaniem są sprawozdania finansowe. Kapitał intelektualny rozumiany jako nieuchwytna aktywa firmy odwzorowuje różnicę między wartością księgową, obejmującą widoczne, materialne aktywa firmy a jej wartością rynkową. Powyższą zależność

Malhotra [183] zapisał w postaci wzoru: wartość rynkowa = kapitał finansowy + kapitał intelektualny. Wyraził także krytykę **technologicznego podejścia** do reprezentacji i zarządzania wiedzą charakterystycznego w obowiązującym cyklu życia systemów inteligentnych (w szczególności systemów informatycznych oraz systemów eksperckich). Wydaje się, że jasno zdefiniowanym celem podejścia technologicznego jest lepsze przewidywanie przyszłości (a w konsekwencji opracowanie przyszłych decyzji) na podstawie historycznych danych i historycznych doświadczeń. Zgodnie z tym podejściem reprezentacja wiedzy jest wydobywana z baz danych w następujących etapach: analiza danych historycznych, analiza przypadków historycznych, generalizacja prowadząca do sformułowania reguł. Podejście technologiczne korzysta z terminologii technik przetwarzania informacji. Stąd wynika jedna z jego głównych wad: różnice w rozumieniu pojęć przez użytkowników systemów SWD. Motywacją do stworzenia teorii kapitału intelektualnego była potrzeba uwzględniania w badaniach nad wartością firmy innych mierników niż tylko wskaźniki finansowe oraz pomiar efektów wdrażania wizji i strategii firmy. Malhotra [183] jako zwolennik teorii kapitału intelektualnego i jej zastosowań w systemach wspomagania decyzji uważa że **to czego naprawdę potrzebują osoby odpowiedzialne za podejmowanie decyzji to nie jest "przewidzenie przyszłości" ale "antycypowanie niespodzianek"**. Stąd wynika konieczność priorytetowego traktowania badań wyjaśniających czym jest znaczenie informacji (jej interpretacja) i w jaki sposób wygenerować informacje oraz często obserwowane rozczarowanie systemami informatycznymi (w szczególności systemami eksperckimi), które wynika z pomijania przez nie znaczenia zjawisk (w przeciwieństwie do ludzkich ekspertów).

Badając firmę ubezpieczeniową Skandia AFS Edvisson i Malone [70], korzystali z metodologii kapitału intelektualnego i sformułowali (obok 73 wskaźników finansowych) 91 dodatkowych miar (wskaźników i wartości bezwzględnych) mających wpływ na określenie wartości firmy. Wśród nich warto wyróżnić: (1) dotyczące personelu: umiejętność obsługi komputera i technologii informacyjnych, rotacja pracowników, odsetek zatrudnionych w kierownictwie, (2) dotyczące relacji z klientami: liczbę dni spędzonych u klienta, odsetek klientów zadowolonych i utraconych (3) dotyczące obsługi procesów w firmie: wydatki na technologie informacyjne przypadające na jednego pracownika, zgodność z normami jakości (4) dotyczące rozwoju: wydatki na szkolenia przypadające na jednego pracownika, wydatki na badania i rozwój do wydatków administracyjnych, odsetek zadowolonych pracowników oraz (5) finansowe, takie jak: zysk przydający na jednego pracownika, wartość rynkowa firmy przypadająca na jednego zatrudnionego itd. Znane są także metodologie: ICM Group [175], Roosa [251], CMA oraz Balanced Score Card [142] reprezentowany przez „deskę rozdzielczą” w postaci trójki: cel - miara (osiągnięcia celu) - inicjatywy (przedsięwzięcie dla osiągnięcia celu) i czwórki perspektyw: finansów – klienta - procesów biznesowych - rozwoju i uczenia się organizacji oraz szwedzki Intangible Asset Monitor [289] oparty na teorii przepływów i zbiorników (*stocks and flows*).

1.2 Analiza decyzyjna jako proces

Podjęcie decyzji (służących realizacji zadań które mają stanowić rozwiązanie problemów występujących w działalności organizacji) w gospodarce rynkowej musi być poprzedzone wykonaniem analizy decyzyjnej. Potwierdzają to Shim i inni [270] powołując się na Gorry'ego i Mortona [100], którzy wychodząc od prac Simona [275] i Anthony'ego [11] wyróżnili trzy **kategorie działań badawczych** prowadzonych podczas podejmowania decyzji:

(1) Postrzeganie, uświadomienie, systematyzacja i sformułowanie istniejących w przedsiębiorstwie problemów (możliwe do wykonania jedynie dzięki inteligencji, wyobraźni, doświadczeniu i wiedzy analityka – przedsiębiorcy, działanie o charakterze strategicznym),

(2) Sięganie do opracowanych metod rozwiązywania problemów, formułowanie wariantów decyzyjnych w języku modeli, generowanie nowych (nieszablonowych) rozwiązań (praca charakterystyczna dla projektanta, o charakterze taktycznym),

(3) Analizowanie wariantów, symulacja i dokonywanie wyboru spośród postawionych wariantów (analiza wykonalności, optymalizacja, kontrolowanie realizacji – praca o charakterze operacyjnym, dająca się (w największym stopniu) programować).

Wykonywanie analizy decyzyjnej polega na opracowaniu oceny poszczególnych działań i ich wariantów które powinny prowadzić do realizacji celów stawianych przez kierownictwo przedsiębiorstwa. Oceny muszą być oparte na jasno określonych kryteriach (właściwych dla polityki kierownictwa), które pozwalają na określenie w jakim stopniu poszczególne działania (jak przewidujemy) i ich następstwa będą wypełniały cele kierownictwa. Jeśli zdecydujemy się na używanie języka matematyki, to możemy stwierdzić że analiza decyzyjna jest stochastycznym problemem optymalizacji. Celem metodologii analizy decyzyjnej jest **ustrukturalizowanie** sposobów poszukiwania rozwiązań optymalnych. Należy zauważyć, że w analizie decyzyjnej mamy do czynienia z dwoma typami zadań: zawierających czynnik losowy (np. ubezpieczenie się w firmie gdy nie wiemy z jakich odszkodowań będziemy korzystać) i nie zawierających czynnika losowego (np. wybór kontrahenta na podstawie odległości od jego siedziby). Już w przypadku problemów decyzyjnych Markowa (procesu stochastycznego w którym przejście ze stanu poprzedniego do stanu bieżącego zależy wyłącznie od stanu poprzedniego; stan procesu nie zależy zatem od trajektorii stanów, gdyż cała potrzebna informacja znajduje się w stanie poprzednim) opisanych w pracy Cassandrasa [35] jednym z zalecanych sposobów rozwiązań jest symulacja. Analiza decyzyjna w przypadku przedsiębiorstwa może być zatem próbą symulacyjnego zweryfikowania domniemanego potencjału firmy.

1.3 Analiza wielokryterialna, modelowanie wariantów decyzyjnych, ocena ryzyka

Możliwe konflikty między realizowanymi celami, ujętymi w oddzielnych kryteriach mogą stanowić przesłankę do powtórnego sformułowania problemu decyzyjnego (w tym prowadzenia analizy wielokryterialnej). Jest to konieczne, gdyż jak twierdzi Szapiro [293] „**konflikt kryteriów uniemożliwia wybór**”. Analiza decyzyjna zajmuje się formułowaniem poszczególnych wariantów decyzyjnych (reprezentowanych przez opisy planowanych przedsięwzięć, różniących się między sobą; nazywanych także działaniami). Wspomaga także rozważanie argumentów za i przeciw dotyczących wariantów decyzyjnych (dokonuje ich oceny). Podobne znaczeniowo cechy analizy decyzyjnej wyszczególnili Barthelemy i inni [18]:

- (1) Podejmowanie decyzji związane jest z całościową **ewaluacją** (wartościowaniem) poszczególnych alternatyw (wykluczających się wariantów) korzystającą z krótkoterminowo dostępnej pamięci roboczej oraz dokonujących porównań na przyjętej skali;
- (2) Analiza decyzyjna jest procesem **stopniowego pozyskiwania wiedzy** (aktualizacji mentalnej reprezentacji problemu) przez decydenta w celu określenia, która decyzja jest najlepsza jeśli chodzi o rozważaną pulę wariantów.

Z powyższego wynika, że **w trakcie analizy decyzyjnej** możliwa jest **zmiana preferowanych wariantów**. W ekonomicznej analizie decyzyjnej największe znaczenia mają jednak argumenty (czynniki) ekonomiczne: **spodziewane** (określone przez wartość prawdopodobieństwa przy poszczególnych stanach natury czyli stanach które mogą się zdarzyć) **zyski** oraz **ponoszone koszty** (rozumiane łącznie jako tzw. „wyłaty”). Zebranie ocen poszczególnych wariantów powinno umożliwić wybór jednego spośród opracowanych wariantów (także brak działań lub brak zmian można uważać za wariant decyzyjny). Przesłanką opracowania ocen poszczególnych wariantów oraz dokonywania spośród nich wyboru najlepszego (optymalnego wg przyjętego kryterium) postępowania jest założenie [293], że lepszy skutek (lepsze wypełnienie celów założonych przez kierownictwo) wynika z lepszego przygotowania materiału badawczego oraz wyboru właściwego kryterium oceny (porównywania poszczególnych działań).

Do przeprowadzenia analizy decyzyjnej niezbędne jest [9] **zdobycie i posiadanie przez decydenta informacji**. Zakres, rodzaj, ilość, pewność, aktualność, koszt, sposób pozyskania informacji zależne są od rodzaju podejmowanej decyzji (skali przedsięwzięcia, zaangażowanych środków, wymagań dotyczących szybkości podjęcia decyzji, akceptowanego poziomu błędu decyzyjnego, skłonności decydenta do ryzyka, liczby wariantów decyzyjnych). Wyróżnić można następujące **fazy analizy decyzyjnej**:

- (1) Przygotowanie (zbieranie informacji o konkurencji, analiza SWOT, porównanie stanu bieżącego z postulowanym)
- (2) Określenie i sprecyzowanie celu zamierzeń (formułowanie celów i środków ich osiągnięcia,

specyfikacja wymagań, które powinien spełniać wariant decyzyjny wybrany jako rozwiązanie, określenie posiadanych zasobów oraz sposobów ich zewnętrznego pozyskania, konstruowanie wariantów decyzyjnych, określenie horyzontu czasowego realizacji przedsięwzięcia)

(3) Sformułowanie kryterium wyboru rozwiązania (określenie zasad rankingu, przeprowadzenie obliczeń – sporządzenie prognozy, rozważanie pewności osiągnięcia wymaganego celu).

Omówione powyżej fazy analizy decyzyjnej są zasadniczo zgodne z **etapami budowy modelu** prowadzącego do wyznaczania optymalnych decyzji podawanymi przez Szapiro [293] czyli:

(1) Określenia pojęcia decyzji, tj. sieci, grafu, wektora/macierzy definiujących przestrzeń podejmowanych decyzji wraz z opisem poszczególnych składowych, parametrów, zmiennych stanu, zmiennych wejściowych

(2) Określenia relacji definiujących zbiór decyzji dopuszczalnych (warunki ograniczające nałożone na składowe)

(3) Skonstruowania i przyjęcia miernika służącego do oceny realizacji postawionego celu dla poszczególnych decyzji wygenerowanych w trakcie przeszukiwania przestrzeni rozwiązań dopuszczalnych

(4) Określenia terminu decyzji optymalnej tj. przekształcenia wartości miernika oceny realizacji celu na ocenę optymalności rozwiązania (maksymalizacja, minimalizacja, zbieżność rozwiązania, zakres stabilności.

Reasumując, problemy decyzyjne zawierające **czynnik niepewności definiowane** są przez następujące wielkości:

(1) Działania czyli opis wariantów postępowania które posiada do wyboru decydent

(2) Stany natury czyli stany, które mogą mieć miejsce w przyszłości; określane często (jeśli to możliwe) za pomocą prawdopodobieństwa (subiektywnego, a priori) – liczbowego lub opisowego (w kategoriach logiki rozmytej)

(3) Tabelę (zestawienie) potencjalnych wypłat (w szczególnym przypadku - wartości oczekiwane) która może być zastąpiona tabelą potencjalnych strat.

Tabela potencjalnych wypłat [9] skonstruowana jest na podstawie obliczonych (wyznaczonych na podstawie doświadczenia, wiedzy eksperckiej, informacji gospodarczych) korzyści (zysków) dla poszczególnych kombinacji wariantów działań i stanów natury. Możliwa jest zatem sytuacja że spodziewana wypłaty **nie ulega zmianom** mimo zmian stanów natury – dzieje się tak np. dla inwestycji o dużym stopniu bezpieczeństwa; natomiast **duże ryzyko** przejawia się dużymi różnicami wartości wypłat. Tabela potencjalnych strat powstaje z przekształcenia (normalizacji) tabeli wypłat (jest zatem jej interpretacją) – dla każdego ze stanów natury wyznacza się największą wartość wypłaty a następnie dla każdego działania wylicza różnicę między tą (największą) wartością a wartością związaną z określonym działaniem. **Strata** interpretowana jest jako osiągnięcie zysku (wypłaty) mniejszego od maksymalnego, możliwego do osiągnięcia przy danych warunkach (stanach natury)

zachodzących (prognozowanych) w przyszłości.

Jeżeli problem decyzyjny dotyczy wielu wariantowych działań wygodniejszą niż tabela wypłat ilustracją (reprezentacją) jest **drzewo decyzyjne**. Bezpośredni potomkowie (węzły) korzenia wskazują na poszczególne stany natury. Węzły reprezentujące stany natury posiadają potomków (wierzchołki) przedstawiających wartości wypłat (nieznormalizowane). Niezależnie czy posługujemy się reprezentacją problemu decyzyjnego za pomocą tabeli czy drzewa, poszukiwać będziemy wariantu, dla którego wierzchołki-potomkowie (po zagregowaniu) spełniają określone kryterium oparte na wartości wypłaty (zespole komórek w tabeli, wierzchołków drzewa). Może to być kryterium maksymalizujące wartości wypłaty (należy zauważyć, że wypłata reprezentowana przez wierzchołek może nie być wartością możliwą do osiągnięcia – np. gdy kryterium ujmuje średnią wartości wypłat) czy minimalizujące wartości straty (uwagi jak powyżej). Ponieważ **brak pewności** co do zajścia w przyszłości określonego stanu natury skutkować może **podjęciem decyzji nieoptymalnej** (a posteriori), to o ile to możliwe należy wykonać próby uzyskania informacji pozwalających zmniejszyć ryzyko. Jak twierdzi Bookstaber [23] w rzeczywistości **największe ryzyko** rynkowe ma miejsce tam, gdzie jest **niewidoczne** lub zaniedbuje się jego obserwacji (monitorowania). **Wpływ ryzyka** jest bowiem **dalekosiężny** a skutki ryzyka mają **naturę wybuchową**, gdyż mogą nie manifestować się w codziennym zachowaniu np. pozycji cen (przykładem może być wprowadzenie na rynek nowego modelu konkurencyjnego produktu).

Przesłanką tworzenia **drzew decyzyjnych** jest potrzeba jednoznacznego wyrażenia zależności hierarchicznych, często mających miejsce w procesie podejmowania decyzji. Wartościową cechą drzew decyzyjnych jest ich **czytelność** i stosunkowa **łatwość weryfikacji reguł postępowania** poprzez przeglądającego je eksperta. Drzewa decyzyjne to struktura, w której wszystkie ścieżki (od strony lewej – korzenia do prawej – liści, węzłów terminalnych) reprezentują sekwencje zdarzeń (np. podejmowanie decyzji). Natomiast poszczególne gałęzie (linie łączące węzły) reprezentują alternatywne decyzje. Korzeń przedstawia bieżącą chwilę. Poszczególne węzły przedstawiają chwile podejmowania decyzji oraz zmianę stanów rzeczy. Drzewa decyzyjne stanowią zatem narzędzie umożliwiające prezentację i analizę decyzji (sekwencji decyzji) podejmowanych w warunkach występowania ryzyka [206] – np. nieznanego poziomu popytu charakteryzowanego przez dwa stany natury: niski lub wysoki popyt. Drzewa decyzyjne [160], [207] składają się z **węzłów, gałęzi i danych skojarzonych z gałęziami**. **Węzły decyzyjne** oznaczane są zwykle symbolem kwadratu, z którego wychodzą linie reprezentujące **dopuszczalne decyzje**. Inny typ węzłów (symbol okręgu) przedstawia **poszczególne stany natury**, których zajście nie jest pewne (tzw. zdarzenia). Z węzłów zdarzeniowych wychodzą linie opisane przez **wartości możliwych wypłat** (wyników). W modelu drzew decyzyjnych występują **gałęzie**; mogą one odwzorować wybór decyzji spośród dostępnych wariantów oraz możliwych stanów natury (są one następstwem podjęcia decyzji lub poprzednich stanów natury). **Dane** służą do wyjaśnienia znaczenia gałęzi – po pierwsze mogą przedstawiać

korzyści lub **koszty** związane z **danym wariantem decyzyjnym** lub poprzedzają określony stan natury (zdarzenie) a po drugie określać **wartość prawdopodobieństwa** wystąpienia danego stanu natury (tożsame z wyborem danego wariantu decyzyjnego). Trenowanie (strojenie parametrów - danych) drzew decyzyjnych odbywa się w trybie **uczenia nadzorowanego**. Dzięki temu struktury drzewiaste zdolne są do formułowania pojęć na podstawie ciągu uczącego. Po wytrenowaniu drzewa decyzyjne można wykorzystać jako **klasyfikator nowych przypadków** (struktura drzewa tworzy zbiór reguł wnioskowania).

Informacja (pełniejsza) o mogących wystąpić w przyszłości stanach natury związana zapewne będzie z zainwestowaniem odpowiedniej kwoty na jej pozyskanie. W przypadku pewności (idealizacja) o konkretnym stanie natury który nastąpi w przyszłości można mówić o wartości doskonałej informacji. Posiadając ją można wyliczyć oczekiwaną wypłatę przy doskonałej informacji (biorąc pod uwagę kolejno poszczególne możliwe stany natury). Natomiast różnicę między tą wypłatą a wypłatą oczekiwaną bez posiadania informacji (dodatkowej, za którą zapłacono) nazywa się **oczekiwaną wartością doskonałej informacji**. Jeśli mamy informację o charakterystyce trafności prognoz dokonywanych przez dany podmiot (doradcę) a dotyczących stanów natury, to można prowadzić dalsze badania nad sensownością zdobywania (odpłatnego) informacji o prognozowanych stanach natury (np. doradca przewiduje zajście jednego ze stanów natury) z wykorzystaniem twierdzenia Bayesa. Dotyczy ono sposobu wyznaczenia prawdopodobieństw warunkowych (rzeczywistych stanów natury przy stanach prognozowanych przez doradcę) nazywanych prawdopodobieństwami a posteriori, przy znanych wartościach prawdopodobieństw a priori (dotyczących stanów natury i sformułowanych przed uzyskaniem nowych informacji). Wyznaczyć można potem oczekiwaną wartość wypłaty dla poszczególnych wariantów decyzji przy prognozowanych przez doradców stanów natury (podstawiając wyliczone prawdopodobieństwo a posteriori w miejsce prawdopodobieństwa a priori). Wartość informacji (prognozy) dostarczanej przez (płatnych) doradców można oszacować (w długim okresie czasu, zakładając dłuższą współpracę z danym doradcą lub znając historię opinii wydawanych przez doradcę) wyznaczając **kwotę** którą opłaca się **zapłacić za uzyskaną informację** (o wynikach eksperymentu wykonanego przez doradcę) – jest to tzw. analiza *pre posteriori* mająca na celu obliczenie oczekiwanej wartości próbki informacji (różnicy między wartością wypłaty z opłaconą dodatkową informacją a wypłaty bez takiej informacji).

Do wykonania analizy *pre posteriori* niezbędne są następujące dane:

- (1) Warianty decyzyjne określone przez tabele wypłat i szacowane prawdopodobieństwa stanów natury
- (2) Tabela charakteryzująca firmę doradcą - zawierająca (dla poszczególnych wariantów decyzyjnych) wartość prawdopodobieństw warunkowych (prawdopodobieństwa a posteriori stanów natury pod warunkiem prognozowanych przez doradcę stanów natury)
- (3) Proponowana przez firmę opłata za dostarczenie informacji (opinii).

Informacje liczbowe potrzebne do wykonania obliczeń oraz ich wyniki przedstawiane mogą być za pomocą tabeli lub w formie graficznej: drzewa losowego (stosowane do wyliczenia prawdopodobieństwa zdarzeń a posteriori) oraz drzewa decyzyjnego (całościowe ujęcie problemu, często wielopiętrowe, składające się z wielu wierzchołków i węzłów), które nadaje się także do analiz *pre posteriori* – porządkuje informacje o wartościach prawdopodobieństw, kwotach wypłat i poszczególnych wariantach. Dalsze badania polegać mogą na **analizie wrażliwości** czyli obserwacji zmian wyników (wypłat) które są efektem manipulacji wartościami parametrów modelu (np. wynagrodzenia dla firmy doradczej, wartości prawdopodobieństw zajścia poszczególnych stanów natury, powiązanych z nimi wartości wypłat a nawet wyłączenie niektórych gałęzi drzewa – poprzez zadeklarowanie zerowego prawdopodobieństwa). Powyższe działanie wymaga jednak posiadania komputerowego modelu symulacyjnego, stworzonego chociażby w arkuszu kalkulacyjnym [293] w którym możliwe jest dokonywanie takich zmian czyli **strojenie modelu** na bieżąco przez edycję wartości poszczególnych komórek arkusza.

1.4 Metodologia i instrumentarium analizy decyzyjnej

Analiza ekonometryczna służy **prognozowaniu** (przewidywaniu) zjawisk (oraz ich parametrów) ekonomicznych na podstawie opracowanych modeli ekonometrycznych. Polega na **wyborze** odpowiedniego (wg przyjętego kryterium i typu problemu) rodzaju **modelu matematycznego**, **dostrojeniu** parametrów modelu oraz **wykonywaniu obliczeń** (sporządzeniu prognozy) – wyznaczenia odpowiedzi (wartości zmiennych wyjściowych) modelu na podawane wymuszenie (czyli ustalone wartości zmiennych wejściowych) przy określonym przebiegu zmiennych (stanu) modelu. **Analiza ekonometryczna i modelowanie symulacyjne** w zarządzaniu charakteryzują się następującymi **cechami wspólnymi**: wykorzystują modele matematyczne wyjaśniające zjawiska ekonomiczne (szczególnie jest to widoczne w analitycznych modelach w modelowaniu ciągłym), służą do weryfikacji hipotez dotyczących charakteru tych zjawisk oraz powinny wykazać przydatność w przewidywaniu przyszłych zjawisk na podstawie znajomości ich modelu i wartości aktualnych (lub hipotetycznych – takie postępowanie jest charakterystyczne dla analizy decyzyjnej. Należy zauważyć [293] **wieloetapowość** postępowania zgodnie z zasadami analizy ekonometrycznej i w tym kontekście wyróżnić w nim trzy etapy:

- (1) **Przygotowanie opisu** zjawiska w języku danego rodzaju modelu ekonometrycznego oraz sformułowanie hipotez (dotyczących przeszłości i teraźniejszości) mających na celu wyjaśnienie charakteru zjawiska z wykorzystaniem danego modelu
- (2) **Oszacowanie parametrów** danego modelu (w przypadku stochastyczności wyznacza się regresję, korelację) w taki sposób aby wyjaśniał dane zjawisko (lub proces) i mógł służyć jako weryfikator postawionych hipotez (zatem dobór modelu powinien być prowadzony ze względu na przydatność w weryfikacji hipotez)
- (3) **Po uznaniu** wyznaczonego **modelu** za godny zaufania wykorzystanie go do prognozowania przyszłego stanu zjawisk i procesów (objaśnianie za pomocą modelu ekonometrycznego).

Hipotezy do rozstrzygnięcia w których zastosować można modele ekonometryczne dotyczą poglądów na występowanie związków (zależności, relacji przyczyna-skutek) między poszczególnymi wyróżnionymi kategoriami ekonomicznymi np. wielkość aktywów a zysk przedsiębiorstwa. Weryfikacja tak postawionych hipotez odbywa się z wykorzystaniem modeli ekonometrycznych, czyli opisanych za pomocą relacji matematycznych (w szczególnym przypadku: równań, układu równań) zależności (w szczególnym przypadku zależności funkcyjnych) między zmiennymi objaśnianymi (których wartości wynikają z postaci relacji, wartości zmiennych wejściowych – objaśniających i parametrów relacji) oraz zmiennymi objaśniającymi (wejściowymi) a także dodatkowego czynnika nazywanego składnikiem losowym, ujmującym zależności które nie zostały zidentyfikowane i opisane w zasadniczej części relacji (zależnej wyłącznie od zmiennych objaśniających). **Weryfikacja merytoryczna** modelu [293] dotyczy **zgodności modelu i teorii** na podstawie której go zbudowano

oraz przydatności modelu do weryfikacji hipotezy badawczej (pretendującej do opisu zjawiska – a nie teorii, gdyż argumenty teorii wykorzystywane są do budowy modelu). Wynika stąd możliwość zmian postaci modelu, liczby i interpretacji zawartych w nim zmiennych objaśniających (zwykle dąży się do wyboru zmiennych mierzalnych, ilościowych) oraz zmiany wartości jego parametrów (strojenie modelu). Pełna weryfikacja hipotezy (gdy w zależności zmienna objaśniana – zmienne objaśniające występuje czynnik losowy) możliwa jest po przeprowadzeniu testów statystycznych (biorących pod uwagę przygotowane – zebrane z rzeczywistości lub opracowane – szeregi czasowe oraz tzw. dane przekrojowe). Dane przekrojowe [293] dotyczą wycinków pewnych zbiorowości.

Proces podejmowania decyzji składa się ze **zidentyfikowania alternatywnych metod działania** i wyboru między nimi. Ze względu na regularność decyzje można podzielić na **programowe** (podejmowane zgodnie z ustalonymi procedurami) oraz **nieprogramowe** (ich przesłanką jest zmiana warunków środowiska: nowy konkurent, wejście na nowy rynek zbytu, wdrożenie nowej technologii).

1.5 Metody optymalizacyjne w analizie decyzyjnej

Optymalizacja rozumiana formalnie polega na **wyszukiwaniu najlepszego rozwiązania** (wartości zmiennych niezależnych - decyzyjnych) przy czym można podać funkcję celu (wyrażenie które ma być optymalizowane) oraz zbiorem ograniczeń (na zmienne decyzyjne). Chociaż liniowe i nieliniowe techniki optymalizacji (programowania) są powszechnie znane to w praktyce pomija się statystyczne fluktuacje w badanym systemie [95]. Zamiast tych technik proponuje się zastosowanie metod heurystycznych. Polegać mogą one [95] na optymalizacji uproszczonych problemów lub wyznaczaniu rozwiązań suboptymalnych (np. zastąpienie w warunkach stochastycznych wyszukiwania pojedynczej optymalnej wartości funkcji celu wyszukiwaniem średnio najwyższej funkcji celu).

Optymalizacja wielokryterialna dotyczy wielu rzeczywistych problemów decyzyjnych. Wielokryterialność oznacza konieczność optymalizacji (sporządzenia rankingu) względem wielu kryteriów jednocześnie (każde kryterium odwzorowuje pogląd lub cel artykułowany przez ten sam podmiot). Uwaga: jest to problem odmienny od pojęcia konfliktu znanego w teorii gier. Cel prowadzenia optymalizacji wielokryterialnej kryje się zwykle pod potocznymi wyrażeniami: „duży efekt małym kosztem”, „coś za coś”. Oznacza to, że w przypadku wielokryterialności poszczególne kryteria są ze sobą skonfliktowane; innymi słowy nie można osiągnąć maksymalnej wartości każdego z kryteriów jednocześnie.

Podstawowa metoda rozwiązywania problemów wielokryterialnych korzysta z pojęcia **optymalności** w sensie **Pareto**. Oparta jest ona na założeniu sformułowanym w [262] stanowiącym, że najlepsze rozwiązanie musi być osiągnięte bez poszkodowania jakiegokolwiek grupy (kryterium, reprezentującego dany punkt widzenia). Wynikiem optymalizacji jednokryterialnej jest zwykle (choć nie zawsze) pojedyncze rozwiązanie, nazywane zbiorem granicznym Pareto. O optymalności rozwiązania w sensie Pareto mówimy, gdy nie można poprawić wartości żadnego z kryterium bez pogorszenia któregoś z pozostałych kryteriów.

W problemach wielokryterialnych można także stosować skalaryzowanie. Polega ono na wyznaczeniu nowego kryterium uwzględniającego wszystkie kryteria cząstkowe. Model matematyczny nowego kryterium konstruuje się poprzez zsumowanie iloczynów wartości poszczególnych kryteriów i arbitralnie przyjętych współczynników wagowych. W wyniku takiego postępowania otrzymuje się zagadnienie optymalizacji rozważane analogicznie do problemów optymalizacyjnych z pojedynczą wartością funkcji kryterium (problem skalarny). Optima wyznaczone dzięki skalaryzacji i podejścia Pareto związane są ze sobą ważną własnością: zmieniając **wartości współczynników wagowych** (w problemie skalaryzacji) wyznaczamy **zbiór graniczny Pareto** (tzn. że zbiór rozwiązań wyznaczonych za pomocą skalaryzacji, przy różnych współczynnikach wagowych jest równoważny zbiorowi granicznemu Pareto).

1.6 Problemy na jakie napotyka analiza decyzyjna

Powodzenie w zarządzaniu przedsiębiorstwem przemysłowym uwarunkowane jest **kompetencją menedżerów i pracowników**. Poszczególne kompetencje składają się na **kompetencje przedsiębiorstwa** określane jako „zdolność do świadomego budowania swojej pozycji rynkowej” [286]. Do przejawów kompetencji należą: umiejętność prawidłowej kontroli kosztów własnych, utrzymanie pozycji konkurencyjnej na rynku, sformułowanie i realizowanie właściwej strategii rozwoju (wykorzystania i alokacji posiadanych środków). Mimo że - jak podaje Strużycki [286] – „proces kontrolowanego kształtowania rentowności, (...) racjonalny wybór technologii i asortymentu produkcji, dystrybucji itd. powinien być zorganizowany, a nie (być) rezultatem prób i błędów” to wydaje się że **dzięki integracji symulacji i odpowiednio dobranych technik poszukiwania rozwiązań na platformie systemu informatycznego** te dwa podejścia **nie będą przeciwstawiane**.

W celu eksperymentalnej weryfikacji tej hipotezy zaproponowałem implementację uznanych w planowaniu strategicznym metod **analizy wrażliwości i analizy scenariuszowej**, ukierunkowanych na wyznaczenie prognozy rentowności, granicznego poziomu zaangażowania środków, wielowariantowość przyszłości i logiczne odwzorowanie hipotetycznych przyszłych zdarzeń (stanów). Analiza wrażliwości polega na obserwacji wpływu modyfikacji parametrów modelu na otrzymywane wnioski-rozwiązania, zmieniając się zatem wartości badanej zmiennej w określonym zakresie, z określonym krokiem. **Wybór wariantu** rozwiązania problemu jest bowiem jednym z etapów **procesu podejmowania decyzji** [61]. Etap ten wymaga **obrony kryteriów oceny** poszczególnych wariantów, **prognozowania** przyczynowo-skutkowego i przemyślanego (np. z wykorzystaniem metody punktowej) **wyboru tego wariantu**, który w największym stopniu będzie – jak sądzimy – spełniał oczekiwania decydenta. **Symulacja** odgrywa tu rolę **metody weryfikującej ex ante** podjęte decyzje. Symulacja rozumiana jako wyznaczenie wyników na podstawie modelu przedsiębiorstwa i zbioru wartości zmiennych decyzyjnych stosowana jest także **na etapie dodatkowej pętli (zewnętrznej) sprzężenia zwrotnego**. Sprzężenie zwrotne służy kontrolowaniu następstw podjętej decyzji i przygotowaniu na ich podstawie następnej decyzji lub (w przypadku symulacji) modyfikacji analizowanego wariantu. **Programowanie zewnętrznej pętli sprzężenia zwrotnego** powierzyć można **metodom klasycznym** (optymalizacyjnym) – przydatnym w podejmowaniu decyzji programowanych, **technikom inteligentnym** (np. algorytmom genetycznym) jak i **interaktywnemu systemowi wspomaganie decyzji** (decyzje nieprogramowane, podejmowane w warunkach nietypowych) czy metodzie delfickiej, w której korzysta się z opinii różnych ekspertów (wykorzystanie systemu eksperckiego zrealizowanego w architekturze tablicowej). Jak się okazało, pozwoli to na osiągnięcie zadowalających rezultatów mierzonych za pomocą uznanych wskaźników ekonomicznych.

System innowacyjny, którym jest hybrydowy system wspomaganie decyzji, napotyka w

swoim cyklu życia następujące kategorie problemów [233]: informacyjne (dotyczące wyboru tylko takiej wiedzy która jest potrzebna), innowacyjne (jak osiągnąć założony cel standardowymi, prototypowymi lub hybrydowymi metodami) i decyzyjne (jakie należy przyjąć kryteria optymalizacji i jakie do nich przyporządkować metody rozwiązań).

Wydaje się że **interakcja między modułami** hybrydowego systemu wspomagania decyzji jest **realizacją grupy reguł „4F”** mających z założenia „sprzyjać efektywności i wygranej z konkurencją oraz podnosić jakość procesów decyzyjnych w przedsiębiorstwie” [278]. Wymienione reguły to *focus* czyli skupianie się na najlepiej znanych i opanowanych rodzajach czy przejawach działalności (np. działalności produkcyjnej i zbycie wyrobów), *fast* czyli szybkie reagowanie poprzez podejmowanie decyzji asortymentowych, *first* oznaczające najlepsze zaspokojenie potrzeb klientów i akcjonariuszy oraz ukierunkowanie na zajęcie pozycji lidera, *flexibility* czyli przygotowanie na ustawiczne wprowadzanie zmian w wykorzystaniu zasobów i korzystanie z innowacji.

Organizacje gospodarcze funkcjonujące w warunkach konkurencji zwiększają swoją wielkość i stają się coraz bardziej skomplikowane i złożone [293]. Następstwem tego jest potrzeba bardziej odpowiedzialnego i podbudowanego modelami matematycznymi podejmowania decyzji. Ze złożoności organizacji (wielkości zatrudnienia, rozmiarów produkcji, zróżnicowania klientów) [293] wynikają następujące cechy: (a) zwiększa się zakres przetwarzanych informacji (liczba danych źródłowych i wariantów decyzyjnych) – wynika to ze zwiększenia się skali działania organizacji, (b) zwiększa się liczba i złożoność związków między poszczególnymi elementami organizacji – wynika to ze zmian dotyczących zadań pracowników, rozszerzenia oferty, wejścia na nowe rynki itd. (c) zwiększone znaczenie czynników niepewności w warunkach konkurencji wymusza szybkie reagowanie (presja czasu) na zmiany w otoczeniu. Stąd także znacznie poważniejsze dla firmy stają się skutki podejmowania decyzji nieoptymalnych (utracone możliwości, wykorzystanie błędnych decyzji przez konkurencję, straty związane z decyzjami błędnymi).

1.6.1 Analiza decyzyjna a pojedynczy podmiot decyzyjny

Barthelemy i inni [18] dowodzą, że ekspert będący jednocześnie decydentem zwykle kieruje się utylitarną **zasadą racjonalności** posiadającą cztery wyróżniki:

- (1) Opracowane przez decydenta-eksperta warianty decyzyjne wyczerpują wszystkie możliwe rozwiązania problemu
- (2) Decydent-ekspert potrafi ocenić wszystkie warianty decyzyjne
- (3) Potrafi porównać je wzajemnie ze sobą i sformułować syntetyczne wnioski
- (4) Decydent-ekspert po przeanalizowaniu wniosków wybiera wariant najbardziej mu odpowiadający.

Rozważając powyżej wymienione wyróżniki, biorąc pod uwagę założenia klasycznej teorii użyteczności której podstawy opracowali Neumann i Morgenstern [214] oraz prezentowane w pracy

Andrzejczaka [9] przykłady problemów decyzyjnych wyróżnić można cztery etapy procesu podejmowania decyzji opartego na modelach matematycznych; jest to nieunikniony ale i przewidywalny skutek:

(1) Zebranie danych źródłowych i skonstruowanie wariantów decyzyjnych uwzględniających czynnik niepewności osiągnięcia danego stanu końcowego. Według Rynkiewicza [255] „Właściwe zebranie danych jest połową sukcesu w konstruowaniu systemu analiz controllingowych. Drugą połowę stanowi poprawne zaprezentowanie tych danych użytkownikom”

(2) Wyznaczenie wartości funkcji celu (szerzej: funkcji użyteczności) dla każdego wariantu decyzyjnego (w sposób deterministyczny)

(3) Wyznaczenie syntetycznej (uwzględniającej wartość prawdopodobieństwa) oceny (np. wartość oczekiwana) poszczególnych wariantów przedsięwzięć

(4) Sformułowanie kryterium wyboru (korzystającego z wyznaczonych syntetycznych mierników oraz wyników analizy wielokryterialnej) i wybranie wariantu optymalnego (o największej wartości oczekiwanej).

Należy jednak zauważyć słabe punkty omówionego racjonalistycznego podejścia, krytykowane z pozycji behawiorystycznych a konkretnie **psychologii kognitywnej** [276] wg której finalny wybór dokonywany przez decydenta (nawet zdolnego do przeprowadzenia ekspertyzy) jest dość elastycznie związany z formalnymi kryteriami wyboru, gdyż jest **ograniczony** przez **reprezentację problemu** w umyśle decydenta (korzystającego z **heurystyk**, porównującego warianty na podstawie **nieostrych kryteriów**, wybierającego wariant wygrywający swego rodzaju pojedynek którego zasady cechują się wielokryterialnością, zmuszanego do wagi szczegółowości i całościowości oceny). Konwencjonalne systemy eksperckie oparte na regułach, działające przecież na platformie systemów komputerowych nie są zatem w stanie posługiwać się semantyką zbieranych informacji. Jest to oczywiste gdy przyjrzymy się sesji pozyskiwania wiedzy w której uczestniczą ekspert dziedzinowy, inżynier wiedzy oraz system ekspercki w której ekspert jest przymuszany [18] do formułowania generalizacji swoich metod poszukiwania rozwiązania i dostosowania się do sposobu „rozumowania” komputera i prowadzenia z nim dialogu za pomocą interfejsu użytkownika.

1.6.2 Uwzględnienie grupowego podmiotu decyzyjnego

Pojawienie się zagadnienia komputerowego wspomaganie grupowego podejmowania decyzji należy widzieć w kontekście ogólnego trendu ostatnich 20 lat **postępu technologicznego** polegającego na przechodzeniu z użytkowania wolnostojących komputerów przez pojedyncze osoby do złożonych sieci telekomunikacyjnych w których urzeczywistnione zostało dzielenie i łączenie zasobów informacji, wiedzy i idei całych zespołów ludzkich działających w biznesie [270]. Proces grupowego podejmowania decyzji jest jednym z przejawów „współpracy wielu osób podejmowanej zgodnie z

opracowanym planem w tym samym procesie produkcyjnym lub w powiązanych ze sobą procesami produkcyjnymi” [263]. Zasadniczym wyróżnikiem grupowego podejmowania decyzji jest konieczność **kierowania się kompromisowym rozwiązaniem uwzględniającym** [18] punkty widzenia a także zróżnicowane priorytety i interesy osobiste (koordynowania). Grupowe podejmowanie decyzji uwarunkowane jest **poziomem komunikacji** między poszczególnymi członkami grupy [58]. Wśród **zalet grupowego podejmowania decyzji** wyróżnić można [284]: powiększenie wykorzystywanego zasobu wiedzy (w porównaniu z pojedynczym decydem), spojrzenie na problem z różnych perspektyw oraz szansę na większą akceptację decyzji w środowisku. Natomiast wśród **wad** podkreślić należy możliwość zdominowania rozwiązań przez priorytety mniejszości, rozproszenie przyjętych celów, presję na przyjmowanie wariantów nośnych społecznie, konieczność zabiegania o udział i odpowiedzialność wszystkich uprawnionych podmiotów. Jak widać z powyższego proces podejmowania decyzji rozpocząć należy od określenia wyboru trybu podejmowania decyzji spośród jednopodmiotowego oraz grupowego. Model umożliwiający rozwiązanie tego problemu został opracowany przez Vrooma i Yettona. Podejście Vrooma-Yettona [315] oparte jest na zebraniu odpowiedzi na zbiór siedmiu pytań tworzących drzewo decyzyjne. Model Vrooma-Yettona znajduje zastosowanie w sterowaniu procesem podejmowania decyzji w dużych korporacjach, przykłady których można znaleźć w [232].

Modele procesów grupowego podejmowania decyzji uwzględniać muszą **wymiar społeczny** (wybór decyzji przez głosowanie, doprowadzanie do konsensusu, modele preferencji, możliwości wpływania na oddawane głosy, realizacja zadań poszczególnych członków), **teoriogrowy** (modelowanie stosunków autonomicznych, skonfliktowanych, konkurencyjnych i posiadających zróżnicowaną wiedzę graczy na rynku będącym wspólnym obiektem zainteresowania), oraz **sieci komunikacji społecznych** (modelujących konfigurację kanałów komunikacyjnych, naturalny język komunikacji, podkreślając znaczenie komunikowania się między podmiotami decyzyjnymi – pojedynczymi decydentami lub grupami decydentów). Tryb **komunikacji** z którego korzystają członkowie zespołu (grupy) odpowiedzialnego za podejmowanie decyzji określa **wymagania technologiczne** ze względu na czas i miejsce podejmowania decyzji [270]. Biorąc pod uwagę **kryterium odstępu czasu** między wyrażeniem opinii przez poszczególnych członków zespołu wyróżnić można komunikację synchroniczną i asynchroniczną. Komunikacja **synchroniczna** odbywa się poprzez rozmowy telefoniczne, chaty internetowe, spotkania bezpośrednie, telekonferencje. Komunikacja **asynchroniczna** może (a czasami musi) być zastosowana gdy konieczne jest przygotowanie dokumentów (opracowań, raportów) przez poszczególnych członków zespołu lub niektórzy członkowie przebywają w miejscu innym niż pozostali i czas wymiany informacji nie jest czynnikiem krytycznym. Ten tryb komunikacji wymaga przestrzegania zasad wymiany informacji (kalendarza spotkań) i może być realizowany przez posługiwanie się pocztą elektroniczną, internetowym forum dyskusyjnym, listą dystrybucyjną.

Uwzględniając powyższe uwagi (a zwłaszcza nadrzędne znaczenie sieci komunikacji społecznych i coraz powszechniejsze rozproszone podejmowanie decyzji) można wyodrębnić (co najmniej) **trzy poziomy komunikacji**: komunikatów, zdań i wypowiedzi (interpretacji). Na poziomie **komunikatów** (porównywanym przez Barthelemy'ego do teorii Shannona) odbywa się nadawanie, przesyłanie i odbiór, kodowanie i dekodowanie krótkich i prostych sekwencji znaków, co najwyżej symboli (odpowiednika wyrazów), bez uwzględniania znaczenia i struktury wypowiedzi (gdyż sąsiedztwo poszczególnych grup binarnych wynika przede wszystkim z aspektów transmisyjnych, korekcji błędów, protokołu transmisji, zabezpieczeń przed przechwyceniem wiadomości itp.). Na poziomie **lingwistycznym** rozpatruje się sekwencje komunikatów (grup znaków i pojedynczych wyrazów) kojarząc je ze sobą zgodnie z ustalonymi zasadami składniowymi. Nie uwzględnia się jeszcze semantyki zdań – nie ma mowy o rozumieniu znaczenia. Najwyższy poziom komunikacji, **interpretacyjny**, uwzględnia już znaczeniowy aspekt przesyłanych komunikatów i zdań. Rozpatrywać można tu zatem cel komunikacji (współpraca międzyludzka oraz człowiek-komputer, CSCW), stopień zrozumiałości (adresowanie do szerokiego kręgu odbiorców) i skuteczności (czy wyartykułowanie treści odniosło zamierzony skutek i w jakim czasie), dobór środków komunikacji (protokołów) człowiek-komputer.

1.6.3 Podejmowanie decyzji – aspekty psychologiczne – strategie

Stockdale [284] twierdzi że podejmowanie decyzji zgodnie z paradygmatem racjonalności ekonomicznej polega na założeniach, że **decydent podejmuje decyzje racjonalne**, wybór wariantu decyzyjnego skutkuje przedsięwzięciem przez realizatora decyzji sekwencji kroków prowadzących do celu, a model sytuacji decyzyjnej tworzony jest w celu optymalizacji przyjętego kryterium. Ze względu na stopień zaangażowania czynnika nadzorującego wyróżnić można następujące strategie podejmowania decyzji [284]: **uczestniczenia (partycypowania) oraz autokratyczne**. Strategia partycypowania polega na zaufaniu udzielanemu przez decydentów podwładnym, delegowaniu przez nich uprawnień do podejmowania decyzji. **Strategia partycypowania** sprawdza się gdy oceniane są głównie jakościowe aspekty decyzji.

Sposób wcielenia w życie decyzji nie jest sztywno określony, gdyż problem decyzyjny jest słabo- lub nieustrukturalizowany. **Autokratyczna strategia** swoje zalety może wykazać w warunkach konieczności podjęcia szybkiej decyzji, dostępnego opracowania strukturalnego modelu sytuacji problemowej, gdy nie ma potrzeby delegowania uprawnień decyzyjnych podwładnym i ocenia się głównie ilościowe efekty decyzji. Sant [259] wyróżnia **cztery charakterologiczne rodzaje decydentów**: analityków (dążący do zdobycia pełnej i pewnej informacji o uwarunkowaniach problemu), poszukujących konsensusu (ukierunkowani na zaspokojenie wszystkich punktów widzenia,

mniej zwracający uwagę na szczegóły techniczne), wizjonerów (ich rozwiązania oparte są na inteligencji, kreatywności, elastyczności w projektowaniu przedsięwzięć w oparciu o dostępne zasoby) oraz pragmatyków (ukierunkowani na realizację kontroli wykonywania decyzji oraz badanie jej efektywności, czyli jej jakości, akceptacji i wykonaniu na czas). Analogiczną **klasyfikację typów** procesów podejmowania decyzji podaje Stockdale [284] wyróżniając dwa rodzaje podejść opartych na **autokratyzmie** (decyzja podejmowana osobiście na podstawie samodzielnie zgromadzonych informacji albo informacji pozyskanych od podwładnych), dwa rodzaje podejść opartych na prowadzeniu **konsultacji** (**zebranie poglądów** od wyizolowanych podmiotów lub **uczestnictwo w dyskusji** a następnie podjęcie decyzji na własny rachunek) oraz **grupowe podejmowanie decyzji** (po przeprowadzeniu narady grupa doradców lub decydentów wybiera wariant decyzyjny do realizacji).

1.6.4 Zebranie informacji o procesach gospodarczych

Decyzja podejmowana jest na podstawie informacji o istotnych dla przedsiębiorstwa procesach gospodarczych. Właściwa (dobra) decyzja [14] **nigdy nie powinna być podejmowana przypadkowo**, gdyż jest skutkiem zamierzonych **wysiłków intelektualnych**, studiów nad poszczególnymi **wariantami** (ich budżetami oraz czasem realizacji) oraz umiejętności jej **wdrożenia**. Na etapie zbierania informacji analityk formułuje pytania typu „kto”, „co”, „gdzie”, „ile” a następnie przystępuje do analizy danych identyfikując wśród nich parametry (wynikające ze struktury badanego systemu, czyli uznawane za stałe w rozważanym horyzoncie decyzyjnym), zmienne na które decydent będzie miał wpływ (sterowalne, określające projektowane warianty decyzyjne) oraz takie na które nie będzie miał wpływu (niesterowalne, wynikające z aktualnego stanu otoczenia systemu).

Opracowanie informacji odbywa się w dwu głównych procesach: **ewidencji gospodarczej** i **sprawozdawczości**. W trakcie pierwszego procesu pozyskiwane są dane o działalności gospodarczej powstające w wyniku operatywnej i księgowej **ewidencji** elementarnych **zdarzeń gospodarczych** w zasobach ewidencyjnych (nie ma mowy o zarządzaniu gdy nie ma dostępnych pomiarów informacji gospodarczych) a drugiego - **okresowo** wykonywanej **sprawozdawczości**. Ewidencja **operatywna** polega na rejestracji pojedynczych zdarzeń (istotnych dla procesu analitycznego) w odpowiednich zasobach urządzeniach i zasobach informacyjnych. Podczas rejestracji wykorzystuje się odpowiednie mierniki ekonomiczne przeważnie o charakterze ilościowym, za wyjątkiem ewidencji wydatków płacowych i przychodów ze sprzedaży. W zasobach informacyjnych przechowywane są **dokumenty źródłowe** poszczególnych zdarzeń oraz proste ich agregaty w rodzaju **zbiorczych zestawień** (np. liczby sprzedanych wyrobów uporządkowanych wg rodzajów), które mogą zostać wykorzystane w procesie sprawozdawczości. Ewidencja **księgowa** posługuje się już pojęciem **planu kont** bilansowych, wynikowych i rozliczeniowych syntetycznie ujmujących zdarzenia w postaci stanów i wyników

działalności, np. należności i zobowiązań, informacji o posiadanych przez przedsiębiorstwo zapasach, środkach pracy, zyskach i stratach, przychodach i kosztach, wpływach i wydatkach i rozliczeń z innymi podmiotami gospodarczymi. Dane z ewidencji operatywnej i księgowej wykorzystywane są w procesie okresowej sprawozdawczości której zadaniami są umożliwienie **prezentacji stanu faktycznego** przedsiębiorstwa i **porównanie** z wartościami zaplanowanymi oraz ocena zmian w kolejnych okresach. Merytoryczna zawartość opracowywanych sprawozdań dotyczy działalności gospodarczej (np. produkcji), sprzedaży i zapasów, sprawozdań finansowych. Sprawozdania posiadają formę zestawień tabelarycznych. Analityczne zadania sprawozdawczości określają cechy prawidłowo prowadzonej sprawozdawczości: **porównywalność danych**, **terminowość** i **poprawność**. Pierwsza cecha wynika z potrzeby porównywania danych uzyskanych w różnych okresach a jej zapewnienie napotyka na trudności w razie zmieniających się parametrów gospodarczych, rozwoju organizacyjnego i technicznego a nawet zmian cen. Potrzeba wykorzystywania **aktualnych danych**, obrazujących okres jak najbardziej bliski momentowi wykonywanej analizy związana jest z wymogiem terminowości danych. Należy więc zapewnić przestrzeganie założonych terminów sporządzania i dostarczania sprawozdań właściwym adresatom. Czynnikiem warunkującym przydatność wykonywanych analiz jest **poprawność** wykonania sprawozdania dotycząca strony **formalnej** (zgodność z ujednoliconymi w skali kraju przepisami, wymogami i wzorcami sprawozdawczości, wydanymi przez instytucję GUSu), **rachunkowej** (brak błędów obliczeniowych, właściwe użycie i interpretacja jednostek miar) i prawdziwości, wiarygodności, czyli zgodności zawartości merytorycznej ze **stanem faktycznym** (prawidłowy pomiar zdarzeń ekonomicznych, kompletność, zapewnienie minimalizacji możliwości przekłamań, zamieszczenie niezbędnych opisów). We właściwym postępowaniu analitycznym następuje wybór oraz weryfikowanie przydatności danych ewidencyjnych w trakcie podejmowania decyzji dotyczących działań podmiotu gospodarczego. Przedmiotem analizy ekonomicznej przedsiębiorstwa jest także przygotowywanie informacji uzyskanych poprzez przetwarzanie danych ewidencyjnych aby można było je wykorzystać na różnych etapach kierowania podmiotem gospodarczym.

1.7 Podejmowanie decyzji w warunkach niepewności

1.7.1 Zastosowanie podejścia teoriogrowego oraz wiedzy eksperckiej

Najbardziej złożone jest modelowanie procesu podejmowania decyzji w warunkach niepewności. W takiej sytuacji wyróżnić można dwa podejścia [319]: wywodzące się z **teorii gier** (podejmowania decyzji w sytuacji występowania konfliktu lub konkurencji) oraz korzystające z **wiedzy eksperckiej** (doświadczenia). Pierwsze podejście oparte jest na **umownych regułach** np. Laplace'a (utożsamiającą niepewność z równym prawdopodobieństwem wszystkich rozpatrywanych stanów natury), MAX-MIN (nazywanej regułą Walda), polegającej na wyborze decyzji, której najgorszy możliwy wynik nie jest gorszy niż najlepszy wynik związany z inną decyzją. Reguła (kryterium) Laplace'a [207] stanowi że rozważanie niepewności można zastąpić metodami podejmowania decyzji w obliczu ryzyka, tzn. wyznaczyć decyzję oferującą najwyższą sumę wypłat przy poszczególnych możliwych stanach natury.

Kryterium **MAX-MIN** [162] można określić jako wersję pesymistyczną gdyż polega na tym, że niezależnie jaką decyzję podejmiemy, to zawsze **następstwem będzie najmniej korzystny stan rzeczy**. Podejście MAX-MIN (opis dotyczy przypadku optymalizacji zysku) składa się z **dwu etapów**. W pierwszym **przeglądamy listę wszystkich alternatyw** i dla każdej z nich wyszukujemy **minimalną wypłatę** (interesujące dla nas od tej pory są tylko te minimalne wypłaty). Drugi krok polega na wyszukiwaniu **spośród pozostawionych minimalnych wypłat** tej która jest **najwyższa** (maksymalna). W teorii gier znana jest także reguła **MIN-MAX** (optymalizująca, czyli minimalizująca koszt), polegająca na wyborze wariantu który **minimalizuje najbardziej niekorzystne możliwe efekty** (w pewnym sensie **symetryczna** do MAX-MIN). Istnieje także odmiana reguły MIN-MAX uwzględniająca zamiast kosztów **utraczone korzyści** (nazywana regułą Savage'a). W metodzie tej optymalizuje się wartość utraconych korzyści (*regret value*) określanej dla danej decyzji jako **różnica między wartością wypłaty i wartością optymalną** – maksymalnej wypłaty lub minimalnego kosztu. Można także wykorzystywać **regułę Hurwicza** (uwzględniającą współczynnik ostrożności) która jest rozszerzeniem reguły MAX-MIN (Walda) oraz **kryterium Bayesa** (wybór wariantu charakteryzującego się największą wartością oczekiwaną wypłaty).

Dla osób lub zespołów o **optymistycznych oczekiwaniach** (tzn. zakładających wystąpienie najkorzystniejszych z możliwych stanów rzeczy) odpowiednia jest reguła MAX-MAX (dla optymalizowanych zysków) i jej odmiana MIN-MIN (dla optymalizowanych kosztów). Natomiast dla osób o **neutralnym nastawieniu** zaproponować można regułę polegającą na założeniu równego prawdopodobieństwa wszystkich możliwych stanów rzeczy. W takiej sytuacji wybór pada na wariant decyzyjny posiadający największą sumę wypłat odpowiadających poszczególnym stanom rzeczy. Gdy dostępna jest wiedza (w sensie możliwości przewidzenia zachowania w przyszłości) pochodząca od

eksperta dziedzinowego lub dane o następstwach decyzji podejmowanych w przeszłości, ale podobnych warunkach, wtedy wykorzystując informacje (szacunki) o prawdopodobieństwach można **sprowadzić problem niepewności do warunków ryzyka**. Warto jednak zdać sobie sprawę, że reguły wnioskowania pozyskane od eksperta dziedzinowego nie mogą pretendować do roli jedyne go źródła wiedzy dla potrzeb prognozowania. Dlatego też **wiedza ekspercka** (połączenie wiedzy podręcznikowej i nabytej w trakcie własnych doświadczeń lub obserwacji innych ekspertów, metodą prób i błędów) jest uzupełniana i aktualizowana zgodnie z dostępnymi danymi historycznymi. Połączenie eksperckich reguł wnioskowania i procedur statystycznych stanowią tzw. *rule-based forecasting systems*. W prognostycznych systemach eksperckich opartych na regułach [2], reguły wnioskowania ulegają **dynamicznym zmianom w trakcie prognozowania**. Zastosowanie tej **hybrydowej techniki** wymaga jednak spełnienia szeregu warunków: posiadanie dobrze zdefiniowanej **wiedzy dziedzinowej**, zidentyfikowanie zależności między **przesłankami reguł** a **przyszłymi stanami natury** oraz **zgodność danych z trendami** dla których opracowano wiarygodne prognozy. Część **przesłankowa** reguły typu „warunek-działanie” wynika z przetworzenia obserwowanych danych (szeregów czasowych) oraz wiedzy dziedzinowej. Wydaje się że taka hybryda będzie generowała bardziej dokładne wyniki niż zwykła ekstrapolacja obserwowanych wartości lub jej algebraiczne przetworzenie (kombinacja ekstrapolacji ze stałymi wagami). Można wskazać na **trzy zasadnicze zalecenia** [337] dotyczące postępowania w warunkach niepewności (związanych nierozdzielnie z ryzykiem):

- (1) Ryzyko nie powinno przewyższać możliwości przedsiębiorstwa (wynikających z ograniczonych możliwości finansowych oraz możliwości przeniesienia ryzyka na ubezpieczyciela)
- (2) Poszczególne rodzaje ryzyka należy traktować odpowiednio do ich charakterystyk (prawdopodobieństwo wystąpienia i maksymalne rozmiary)
- (3) Należy zachować proporcjonalność między skalą ryzyka a osiągalnymi korzyściami oraz występującymi zagrożeniami (nie ryzykować zbyt wiele przy spodziewanych nieznacznych korzyściach i nie stosować zbyt kosztownych zabezpieczeń przy niskim prawdopodobieństwie ryzyka).

Lawrence i Pasternack [162] wyszczególnili dwa podstawowe ujęcia wartości prawdopodobieństwa poszczególnych przyszłych stanów natury (rezultatów decyzji): największej oczekiwanej wartości wypłaty i najmniejszej oczekiwanej utraconej korzyści. Nie uwzględniają one nastawienia decydenta możliwości ponoszenia strat (ryzyka). Porównując osiągnane rezultaty dla decyzji podejmowanych w warunkach pewności i optymalizowanych decyzji w warunkach ryzyka można wyznaczyć **oczekiwaną wartość doskonałej informacji**. Jest to różnica między **oczekiwaną wypłatą** która będzie osiągnięta przy **doskonałej informacji** (uwzględniająca wszystkie możliwe stany natury) a oczekiwaną wypłatą **przy braku dostępu do pewnej informacji** o wystąpieniu danego stanu natury. Pierwsza zaleca **wybór decyzji maksymalizującej** oczekiwaną wypłatę a druga

minimalizuje utraconą korzyść czyli różnicę między wartością wypłaty a najlepszą dostępną wypłatą dla danego wariantu decyzyjnego.

1.7.2 Symulacja i analiza statystyczna

Proces podejmowania decyzji w rzeczywistych warunkach musi uwzględniać występowanie niepewności (tzn. brakiem informacji i wiedzy przyszłym stanie natury). Jedynym czego możemy być pewni w ekonomii [219] jest niepewność a wiedza i umiejętność korzystania z niej są jedynymi trwałymi źródłami przewagi konkurencyjnej. Niepewność spowodowana jest **brakiem dostępu do informacji** o czynnikach oddziałujących na wyniki podejmowanych decyzji, ale także brakiem modeli umożliwiających bezbłędne prognozowanie. Potwierdza to Prusak [240] pisząc że „wiedza jako czynnik konkurencyjności w danej organizacji jest tym o czym organizacja wie, jak to wykorzystuje i jak potrafi dowiedzieć się czegoś nowego”. Można jednak radzić sobie w warunkach niepewności (pod warunkiem opracowania modeli oferujących akceptowalny poziom błędów) stosując analizę decyzyjną **ujmującą niepewność w kategoriach liczbowych** (traktując niepewność jako stały składnik formuł wyliczających syntetyczne mierniki kryterium decyzyjnego).

Kluczową kwestią w analizie decyzyjnej jest zastosowanie odpowiedniej **techniki ilościowego** rozważania opisu wariantu decyzyjnego i jego prawdopodobnych rezultatów w warunkach niepewności. Jest to możliwe dzięki opisowi niepewności w terminologii matematycznej teorii prawdopodobieństwa, ujmującej prawdopodobieństwo (przyjmowania przez zmienną losową pewnych wartości) w kategoriach liczbowych. Bardzo użytecznym rozwinięciem narzędzi powstałych w zakresie teorii prawdopodobieństwa (statystyce) jest **symulacja komputerowa**. Eksperymenty symulacyjne korzystają z wylosowanych wartości (realizacji zmiennych losowych) dostarczanych przez **generator liczb pseudolosowych**. Dzięki prawidłowo działającemu generatorowi liczb pseudolosowych otrzymujemy wylosowany zbiór wartości, który rozważany kategoriami statystycznymi jest zgodny z wynikami teoretycznego (idealnego) generatora losowego, dostarczającego odpowiednie wartości zgodnie z założonym rozkładem teoretycznym. W **badaniach symulacyjnych i statystyce** posługujemy się terminami prawdopodobieństwa, prawdopodobieństwa warunkowego (prawdopodobieństwa przybrania przez badaną zmienną pewnej wartości w danych warunkach, np. wartości innej zmiennej), prawdopodobieństwa a priori (przekonanie o prawdopodobieństwie przed przeprowadzeniem eksperymentu i pomiaru (obserwacji), a posteriori (prawdopodobieństwo wyliczone na podstawie obserwacji, przydatne podczas wyznaczania prawdopodobieństw warunkowych). O ciągłym charakterze prawdopodobieństwa mówimy wtedy gdy możemy określić prawdopodobieństwo (dodatnie) że zmienna losowa przybierze wartość większą od X , ale nie możemy określić prawdopodobieństwa (dodatniego) że wyniesie dokładnie X . Istotne jest pojęcie niezależności zmiennych oznaczające, że na podstawie znajomości przebiegu wartości jednej

zmiennej nie możemy nic nowego powiedzieć o wartościach drugiej zmiennej; dla niezależnych zmiennych losowych rozkład prawdopodobieństwa łącznego wyraża się wzorem $P(X=x \cap Y=y) = P(X=x) * P(Y=y)$ oraz niezależności warunkowej dwu zmiennych losowych A względem B, przy określonej zmiennej losowej C: jeśli posiadamy wiedzę o zmiennej losowej C, to z pozyskania wiedzy o A nic nowego nie dowiemy się o zmiennej B (i podobnie o B względem A). **Podejście symulacyjne** w modelowaniu niepewności polega na przyjęciu że rzeczywiste wartości zmiennych losowych mogą (dla potrzeb rozwiązywania problemów) być zastąpione przez wygenerowane komputerowo (symulowane) wartości losowe charakteryzowane przez określone wartości parametrów. Jest tak, jeśli prawdopodobieństwo wygenerowania wartości zmiennej losowej poniżej pewnego progu t jest równe prawdopodobieństwu zajścia zdarzenia (w rzeczywistym procesie), reprezentowanego przez zmienną losową o wartości poniżej progu t. Jeśli powyższa zgodność zachodzi dla wszystkich wartości t które może przyjmować zmienna losowa to powiedzieć można, że rozkłady prawdopodobieństw są takie same. Innymi słowy **niepewności dotyczące kolejnej wartości zarówno symulowanej zmiennej losowej jak i rzeczywistej zmiennej losowej są jednakowe**. Powyższe rozumowanie jest prawidłowe zarówno dla zmiennych losowych o charakterze ciągłym jak i zmiennych losowych dyskretnych, przybierających wartości ze zbioru o ograniczonej liczbie elementów. Nie zawsze jednak możliwe jest określenie dokładnego przebiegu rozkładu zmiennych losowych (rzeczywistych). W takiej sytuacji można go przynajmniej w jakimś ograniczonym zakresie zastąpić (podczas konstruowania modelu symulacyjnego) takimi dobrze znanymi pojęciami (miernikami zmienności) jak wartość oczekiwana, średnia, odchylenie standardowe, średni błąd bezwzględny [211].

1.7.3 Drzewa prawdopodobieństw

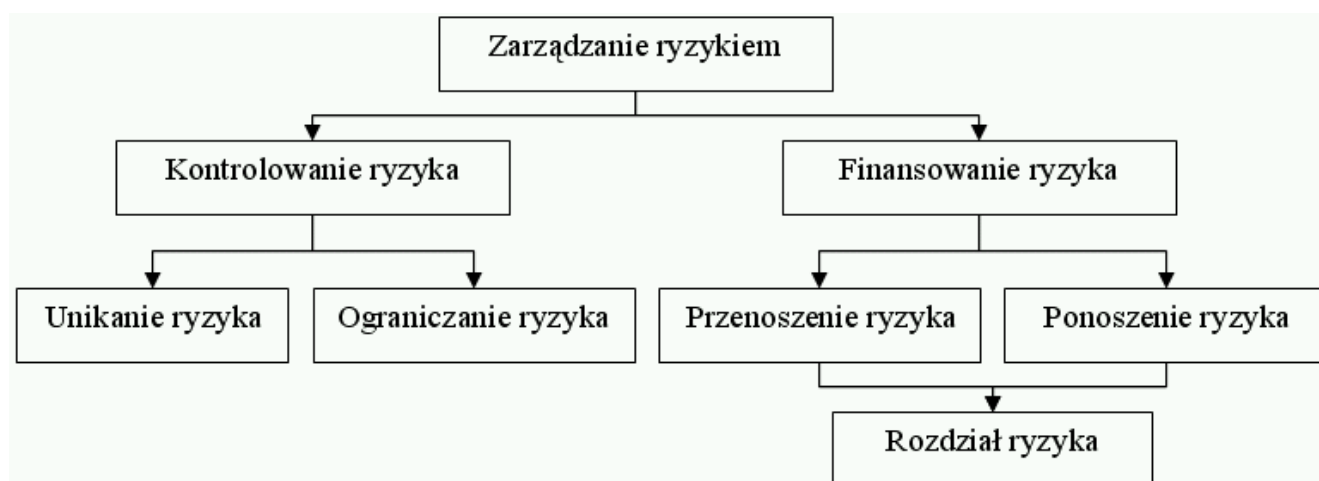
Drzewa prawdopodobieństw posiadają wiele cech wspólnych z drzewami decyzyjnymi znajdującymi zastosowanie w teorii decyzji. Są grafami bez cykli, składają się z węzłów i połączeń. Połączenia wychodzące z poszczególnych węzłów reprezentują wszystkie możliwe zdarzenia (rozumiane jako odpowiedzi na pytania – usuwające niepewność) wraz z odpowiadającym im prawdopodobieństwem (warunkowym). Następniki poszczególnych węzłów wzajemnie wykluczają się (tzn. że może zajść jedno i tylko jedno zdarzenie – w danej iteracji). Oczywiście suma wszystkich prawdopodobieństw warunkowych dla następników danego węzła (na tym samym poziomie hierarchii) wynosi 1 oraz suma prawdopodobieństw wszystkich zdarzeń reprezentowanych przez wszystkie terminalne węzły (liście, nie posiadające następników) w drzewie wynosi 1. Drzewa prawdopodobieństw cechują się także własnością polegającą na tym, że **prawdopodobieństwo wszystkich zdarzeń** reprezentowanych przez gałąź (drogę wychodzącą z jednego z węzłów do której należą wszystkie pozostałe węzły) od korzenia do rozważanego węzła jest iloczynem wszystkich

prawdopodobieństw warunkowych węzłów należących do rozważanego fragmentu gałęzi. Prawdopodobieństwo to nazywamy **prawdopodobieństwem przecięcia**, czyli jednoczesnego zajścia co najmniej dwu zdarzeń [211]. Z prawdopodobieństwem przecięcia związana jest reguła Bayesa, w której przedstawiono związek $P(A | B) = P(A \cap B) / P(B)$.

W kontekście drzew prawdopodobieństw **reguła Bayesa** oznacza że prawdopodobieństwo warunkowe można wyznaczyć dzieląc sumę (znanych) prawdopodobieństw przecięcia obu zdarzeń przez sumę prawdopodobieństw zdarzenia. Podobną strukturę do drzew prawdopodobieństw posiadają **drzewa oczekiwania** (*expectation trees*) [211]. Różnica między nimi polega na tym że w węzłach terminalnych (liściach) zawierają one formułę pozwalającą na wyliczenie wartości (oczekiwanych w następnej iteracji) zależnych od wartości zmiennych losowych reprezentowanych w drzewie.

1.7.4 Aspekty zarządzania ryzykiem i ocena zagrożenia

W instytucjach finansowych istnieją są dwa sposoby pomiaru ekspozycji na ryzyko rynkowe: system wczesnego ostrzegania oraz polegający na dekompozycji i agregacji ryzyka [Rys. 1-1]. **Systemy wczesnego ostrzegania** opisane w pracy Altmana i innych [8] dotyczą ryzyka niewypłacalności (ściśle zdefiniowanego) a ich celem jest przewidywanie trudnej sytuacji dłużnika lub warunków prowadzących do takiej sytuacji na podstawie znajomości określonych zmiennych. System wczesnego ostrzegania wielokrotnie korzysta ze zbudowanego modelu, **monitorując wartości zmiennych i reagując** (uruchamiając sprzężenie zwrotne) na wykryte niepokojące zdarzenia. Należy zauważyć, że systemy takie jako wykorzystujące **statystyczną klasyfikację** (przydzielanie danego przypadku do klas zagrożenia bankructwem) nie wyjaśniają i nie identyfikują czynników odpowiedzialnych za pojawienie się kłopotów, a jedynie zajmują się określeniem objawów złej sytuacji.



Rys. 1-1 Klasyfikacja metod zarządzania ryzykiem. Źródło: [337]

Dekompozycja i agregacja ryzyka została zastosowana w modelach wyceny aktywowo-

kapitałowej (*capital asset pricing models*) opisanych przez Galindo i Tamayo [90] takich jak CAPM, ICAPM, CCAPM oraz teorię arbitrażu cen. Wspólne cechy powyższych modeli to **zależność czynników ryzyka od wartości rynkowej posiadanych aktywów** oraz **większe znaczenie tych czynników ryzyka niż ryzyka rynkowego**. Metoda dekompozycji i agregacji ryzyka polega na **dekompozycji poszczególnych aktywów i zobowiązań** na klasy poddane określonym czynnikom ryzyka (stopy procentowej, kursów walut, cen akcji i towarów itd.) a następnie **agregacji** na podstawie poszczególnych czynników ryzyka. **Dekompozycja** polega na właściwej identyfikacji tych czynników i dokładnym oszacowaniu stopnia narażenia na te czynniki. Porównując systemy wczesnego ostrzegania oraz metodę dekompozycji i agregacji ryzyka należy zauważyć że choć druga metoda pozwala na ilościowe uwzględnienie źródeł ryzyka i przez to łatwiejsze staje się zarządzanie ryzykiem (rozumiane jako kontrolowanie ryzyka i jego finansowanie) to jednak **pierwsza sprawdza się lepiej w warunkach mniejszej dostępności informacji** i wymaga mniej obliczeń.

1.7.5 Technika estymacji

Metodą oszacowania nieznanych wartości liczbowych zmiennych losowych jest **estymacja punktowa, usuwająca** całkowicie z modelu **niepewność**. Wyznaczone dzięki niej estymatory zmiennych losowych uwzględniane są podczas poszukiwania rozwiązań optymalnych. W miejsce zmiennych losowych uwzględnianych w formułach kryteriów decyzyjnych **podstawia się estymatory tych zmiennych**, co pozwala traktować problem jako deterministyczny, ale jednocześnie nie pozwala na symulowanie zmienności w modelu.

Wartość oczekiwana może być obliczona wg formuły $E(X) = \mu_x = \sum_x P(X = x) * x$. Należy wspomnieć, że zarówno dla ciągłych jak i dyskretnych zmiennych losowych wartość oczekiwana może w rzeczywistości znajdować się poza zbiorem wartości zmiennych (przeciwdziedziny), za to położony jest w pewnym oddaleniu od krańców zbioru wartości zmiennych (na skali uporządkowanej). W przypadku dyskretnych zmiennych losowych z postaci wzoru na wartość oczekiwaną wynika, że wartość oczekiwana może nie należeć do zbioru dyskretnego, co więcej jej obliczenie wymaga uporządkowania zbioru wartości zmiennych losowych a następnie przyjęcia skali liczbowej. Warto wspomnieć także o odpowiedniku wartości oczekiwanej dla prawdopodobieństwa warunkowego, nazywanym warunkową średnią \underline{y} , wyrażoną wzorem $P(Y < \underline{y} | X = x) = 0,5$. Podsumowując, można stwierdzić że wartość oczekiwana nie jest w ogólnym przypadku tożsama z wartością najczęściej przyjmowaną przez zmienną losową.

Kolejna powszechnie stosowana miara zmienności, **odchylenie standardowe** wyraża się wzorem $stdev(X) = \sigma_x = \sqrt{E(X - \mu_x)^2} = \sqrt{\sum_x P(X = x) * (x - \mu_x)^2}$, gdzie $\mu_x = E(X)$, a \sum_x oznacza sumowanie po wszystkich możliwych wartości zmiennej losowej. Odchylenie

standardowe jest zatem pierwiastkiem kwadratowym z wartości oczekiwanej kwadratu różnicy między wartością zmiennej losowej a średnią (wartością oczekiwaną zmiennej losowej). Kwadrat odchylenia standardowego nazywamy **wariancją zmiennej losowej**. Odchylenie standardowe (a tym samym wariancja) jest miarą rozrzutu wartości zmiennych losowych. Można stwierdzić, że im większa wariancja, tym większy rozrzut, tzn. tym bardziej prawdopodobne jest przyjęcie przez zmienną losową wartości położonych w dalszej odległości od wartości oczekiwanej. Jednocześnie wartość odchylenia standardowego może stanowić wskazówkę, na ile prawdopodobne jest przyjęcie przez zmienną losową wartości należącej do danego zakresu, licząc od wartości oczekiwanej w górę lub w dół. Empiryczne oszacowanie wartości oczekiwanych i odchylenia standardowego oparte jest na prawie wielkich liczb. Według prawa wielkich liczb wystarczająco duża próbka pochodząca z niezmiennającego się rozkładu prawdopodobieństwa (ale o nieznanym charakterystyce) może służyć do bardzo dokładnego oszacowania wartości oczekiwanej wyznaczenia wartości oczekiwanej na podstawie obliczonej wartości średniej (z próbki). Podobnie prawo wielkich liczb ma zastosowanie w wyznaczaniu odchylenia standardowego rozkładu rzeczywistego na podstawie odchylenia standardowego próbki. W tym celu należy wyznaczyć estymowaną (za pomocą średniej) wartość oczekiwaną próbki, następnie podzielić sumę podniesionych do kwadratu odchylen (różnicy) próbek od wartości oczekiwanej (estymowanej) przez pomniejszoną o jeden liczbę próbek (korekta w górę).

Kolejnym ważnym prawem rządzącym rachunkiem prawdopodobieństwa **jest centralne twierdzenie graniczne**, stwierdzające szczególny charakter rozkładu normalnego i stanowiące że rozkład ten można zastosować do określenia zachowania się wartości statystycznych próbki. Według centralnego twierdzenia granicznego [211], jeśli z rozkładu prawdopodobieństwa (nie znamy jego kształtu) charakteryzującego się wartością oczekiwaną μ i odchyleniu standardowym σ pobierzemy próbkę składającą się z n elementów (niezależnie wybranych) i obliczymy jej średnią, to jeśli potraktujemy ją jak zmienną losową to jej rozkład prawdopodobieństwa można aproksymować rozkładem normalnym oraz posiada ona wartość oczekiwaną także wynoszącą μ natomiast odchylenie standardowe jest równe σ / \sqrt{n} . Jeśli znamy odchylenie standardowe i znamy sposób wygenerowania rozważanej zmiennej losowej to centralne twierdzenie graniczne umożliwia określenie z określonym stopniem zaufania wyrażonym liczbami wynikającymi z własności rozkładu normalnego (tj. 0.683, 0.95, 0.997) wartości oczekiwanej tej zmiennej losowej (na podstawie obserwacji symulowanych realizacji zmiennej losowej).

1.8 Analiza decyzyjna jako przedmiot badań

Decyzje podejmujemy wtedy, gdy dokonujemy wyboru spośród alternatywnych rozwiązań dotyczących postępowania w przyszłości w celu osiągnięcia określonego celu lub zamierzenia. Cardenosa i inni [34] wyróżniają cztery aspekty podejmowania decyzji: wybór, ukierunkowanie na przyszłość, proces poznawczy, celowość działania. Poniżej omówiono trzy najistotniejsze wymiary w kontekście niniejszej pracy.

Wybór – każda decyzja związana jest z pewnym stopniem wolności wyboru dotyczącego teraźniejszości i przyszłości. Nie ma mowy o decyzji jeśli jest tylko jedna możliwa droga postępowania. Podejmowanie decyzji jest nieodłączną czynnością człowieka wykonywaną przez niego codziennie. Przyszłość – każda decyzja oprócz oddziaływania na teraźniejszość będzie miała skutki w przyszłości.

Proces poznawczy – decyzja musi być podejmowana jako akt rozumu. Osoba wykonująca intelektualną czynność wyboru, stosuje własną skalę preferencji, szeregując wg niej syntetyczne oceny poszczególnych wariantów decyzyjnych. Teoria decyzji znajduje zastosowanie w **warunkach pewności** (każdy z wybranych wariantów posiada jednoznacznie odpowiadające mu następstwa – wybór między wariantami jest wyborem między następstwami), **ryzyka** (każdy wariant może prowadzić do następstwa należącego do zbioru następstw i każdy element tego zbioru ma określone znane prawdopodobieństwo zależne od wybranego wariantu, wyboru zatem dokonujemy między rozkładami prawdopodobieństw) i **niepewności** (rozkłady prawdopodobieństw nie są znane).

Celowość – podmiot podejmujący decyzje musi mieć na względzie istnienie celu do którego prowadzić ma decyzja. Gdy tak nie było, decyzja mogłaby być za każdym razem losowana. Z celowością podejmowania decyzji związany jest aspekt przewidywalności podejmowanych decyzji. Wynika z niego konieczność posiadania doświadczenia, wiedzy a także umiejętności korzystania z modeli decyzyjnych oraz przeprowadzania eksperymentów decyzyjnych.

Teoria decyzji posiada wiele wspólnych cech z teorią gier, przydatną szczególnie w podejmowaniu decyzji w warunkach konkurencji czy konfliktu. Teorię decyzji można traktować [162] jako szczególny przypadek teorii gier w którym przeciwnik decydenta jest bezosobowy (bezrozumny) lub można go traktować jako przejaw czynników naturalnych (tzw. gry z naturą). Można przedstawić wiele kryteriów klasyfikacji gier, np. liczbę graczy (dwóch lub więcej), sumie wypłat (gry o sumie zero i o niezerowej), sekwencji ruchów (sekwencyjne lub równoczesne).

Rao [247] dokonał **podziału reguł decyzyjnych** na dwie grupy: **niezrandomizowane** (takie, które dokonują zaklasyfikowania wyniku pomiaru do jednego z wzajemnie rozłącznych obszarów (reprezentujących poszczególne populacje)) oraz **zrandomizowane** (takie, które zaklasyfikowania dokonują po wykonaniu eksperymentu losowego (wygenerowaniu wartości reprezentujących poszczególne populacje, ale zgodnie z pewnym zależnym od wyników pomiaru

prawdopodobieństwem). W takiej sytuacji za lepszą uznaje się regułę która skutkuje **mniejszą wartością oczekiwaną strat**, określanych jako **błędne zaklasyfikowanie pomiaru**. Teoria decyzji uwzględnia **preferencje i wymagania użytkownika** przez sporządzenie **rankingu wariantów** według obranego **kryterium**. Dzięki obranemu kryterium możliwe jest rozważanie nietrywialnych problemów decyzyjnych. Posiadanie modelu powiązań między przyczyną a skutkiem, czyli informacji o odwzorowaniu między zbiorem dostępnych alternatyw a zbiorem wyników umożliwia racjonalne (zgodne z przekonaniem/preferencjami decydenta) podejmowanie decyzji.

Jeśli jesteśmy w stanie z całkowitą pewnością orzec o następstwach (wynikach) wszystkich decyzji (alternatyw) to wtedy sprawa wyboru staje się prosta: należy zdefiniować funkcję jedno lub wieloargumentową, która **uporządkuje zbiór wyników**, porządkując jednocześnie zbiór wariantów. Podstawowe **pojęcia analizy decyzyjnej** [162] w warunkach dyskretnego zbioru następstw które można z całkowitą pewnością przyporządkować danej decyzji to **alternatywy decyzji**, **stany natury** (przyszłe zdarzenia które mogą być następstwem podjętych decyzji) i **wypłaty** (syntetyczny opis powiązań między decyzjami i ich skutkami – np. wartości prawdopodobieństw). Tabela **wypłat** składa się z wierszy reprezentujących poszczególne alternatywne decyzje i kolumn zawierających odpowiednie możliwe rezultaty (stany natury). Podczas wypełniania tablicy wypłat przestrzega się założeń że miejsce będzie miał jeden i tylko jeden ze stanów rzeczy, a decydent po podjęciu decyzji nie ma już wpływu na rezultat. Poszczególne komórki tablicy (macierzy) wypłat zawierać mogą **wartości liczbowe, pieniężne lub symbole**. Aby możliwe było dokonanie wyboru optymalnego, należy jednak poszczególne wypłaty uporządkować.

Teoria decyzji zakłada, że chociaż decydent może **preferować** wysokie ryzyko, średnie lub unikać ryzyka (preferencje te znajdują odzwierciedlenie w wartości funkcji użyteczności) to zawsze wybiera **decyzję o najwyższej oczekiwanej wartości funkcji użyteczności** (jedno lub wieloargumentowej). Wartość funkcji użyteczności określa względne położenie wyniku (wypłaty) decyzji na skali między najmniej korzystną i najbardziej korzystną spośród możliwych do wystąpienia alternatyw. Schemat podejmowania decyzji w warunkach ryzyka oparty na określeniu użyteczności [207] składa się z następujących etapów:

- (1) Określenia użyteczności poszczególnych możliwych wypłat
- (2) Oszacowania prawdopodobieństwa poszczególnych możliwych wypłat
- (3) Obliczenia wartości oczekiwanej użyteczności dla każdej decyzji
- (4) Wyboru decyzji o najwyższej wartości oczekiwanej użyteczności.

1.9 Podejmowanie decyzji jako proces fazowy

Dutta [67], Gibson [94] i Power [237] za Simonem [275] wyróżniają trzy fazy procesu przygotowania decyzji: wykorzystanie ludzkiej inteligencji (rozpoznanie), projektowanie i wybór (i decyzja o skierowaniu do wdrożenia). Mintzberg [218] rozszerzył proces na pięć faz [Rys. 1-2].

1. **Rozpoznanie problemu**, wywiad w który została zaangażowana ludzka inteligencja; w tym:

a) Identyfikacja i ewentualnie zdefiniowanie celów organizacji, pozyskiwanie i systematyzacja danych, zebranie informacji umożliwiających zaklasyfikowanie problemu

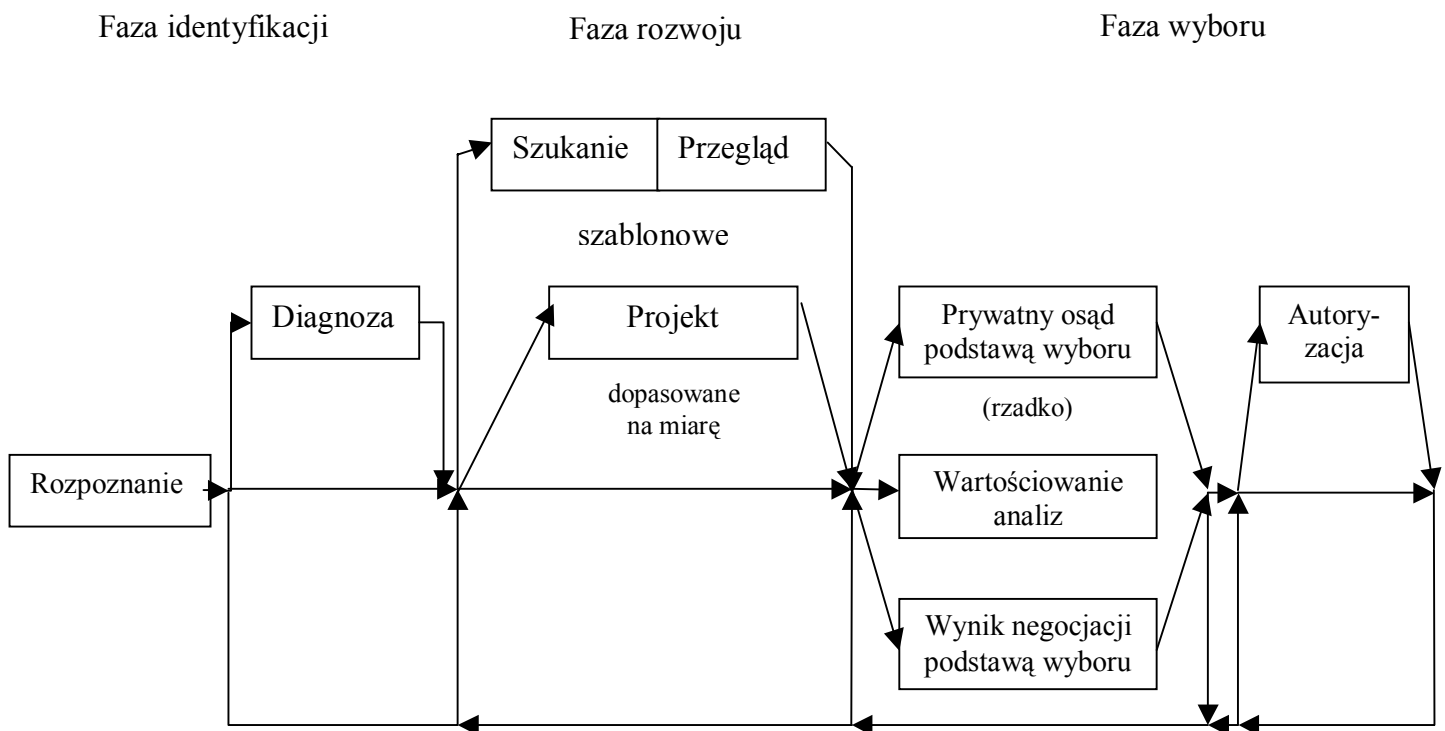
b) Wstęp do procesu podejmowania decyzji poprzez wykonanie działań o charakterze wywiadowczym. Obejmują one oględziny badanej sytuacji oraz nakreślenie celu badań, scharakteryzowanie problemu, poszukiwania metodologii rozwiązań,

2. **Projektowanie modelu i eksperymentów**. Polega ono na konstruowaniu modelu w tym ustaleniu kryteriów wyboru wariantów, budowie zbioru dopuszczalnych wariantów, sformułowaniu zasad oceny prognozowanych wyników przyporządkowanych poszczególnym wariantom,

3. **Dokonanie wyboru** (właściwe podejmowanie decyzji). Obejmuje ono:

a) Wykonanie **serii eksperymentów** tzn. zebranie **wyników symulacji** wykonywanej na podstawie modelu. Operacje te wykonywane są zgodnie z **planem eksperymentu** i przyjmują formę analizy wrażliwości, wyboru najlepszych wariantów na podstawie przyjętych kryteriów. Z dokonaniem wyboru wariantów (rozwiązań problemu) wiąże się także planowanie etapów implementacji

b) Właściwą decyzję czyli **skierowanie do wykonania** określonego zestawu wartości zmiennych decyzyjnych jako wyniku procedury wyboru wariantu.



Rys. 1-2 Inkrementacyjny model faz procesu podejmowania decyzji. Źródło: [218]

2 Modelowanie i symulacja

2.1 Symulacja i modelowanie symulacyjne - przedmiot badań

Modelowaniem możemy nazwać [37] wyszukiwanie w systemie cech i związków istotnych ze względu na dany cel (czyli budowanie modeli). **Modelowanie** wraz z **symulacją** (wzajemnie uzupełniająca się diada) należą wg Kleibera [150] do **trzech filarów współczesnej nauki** (stanowiąc dopełnienie **teorii i eksperymentu**) przede wszystkim o wykonywanie obliczeń, analizę i interpretację wyników, zmianę skali czasu i przestrzeni (to ostatecznie zadanie nazywane jest właśnie podstawą symulacji). **Symulacja** posługując się modelami pozwala na badanie dynamicznych zależności przyczynowo-skutkowych w szerokim zakresie zmian przestrzeni (odległości), materii (kapitał, majątek, przepływy finansowe) i czasu (inwestycje, dekapitalizacja majątku, zmiany cen, popytu itp.) za pomocą aparatu matematyczno-komputerowego. Symulacja komputerowa jest niezastąpiona w badaniu złożonych systemów, jeżeli brak (lub nie można ich zastosować z przesłanek czasowych, kosztowych itp.) modeli przyczynowo-skutkowych. Dzięki symulacji osiągnąć można [37] następujące korzyści:

- (1) Zmniejszenie kosztów oraz czasu projektowania i planowania przedsięwzięć (obserwacja oddziaływania na środowisko, oszacowanie skutków zastosowania poszczególnych wariantów, eliminacja działań szkodliwych i nieopłacalnych)
- (2) Możliwość przewidywania własności nowych wyrobów, technologii i przedsięwzięć na podstawie znanych (na podstawie ich modeli) własności zastosowanych elementów, technik i obserwowanych w przeszłości zachowań w danej organizacji (uogólnienie, akcent na synergię i podejście systemowe)
- (3) Korzystanie (wzorowanie się) z rozwiązań sprawdzonych w wirtualnej (możliwej do zaistnienia) rzeczywistości (symulacyjna weryfikacja i walidacja)
- (4) Integracja modułów komputerowych modeli symulacyjnych ze składnikami rzeczywistych systemów: zastępowanie (reprezentacja) fragmentów rzeczywistości modułami komputerowymi (zniesienie ograniczeń komunikacyjnych, czasochłonnych i kosztownych pomiarów parametrów systemu rzeczywistego, wirtualna diagnostyka)
- (5) Realne są próby budowy struktur hierarchicznych (modeli mikro i makroekonomicznych) zawierających moduły wirtualne integrowane z modelami uczestników gry rynkowej oraz obserwowanych trendów gospodarczych.

Przeprowadzenie **postępowania badawczego** nazywanego modelowaniem wymaga wykonania następujących kroków [37]:

- (1) Ustalenia obszaru rzeczywistości (granic systemu/procesu który będziemy modelować); działanie to obejmuje także wyodrębnienie przedmiotu badań z otoczenia (które także należy scharakteryzować, jednak jego model będzie posiadał prawdopodobnie mniejszą złożoność niż modelowany obiekt (lub układ))
- (2) Ustalenia stopnia szczegółowości docelowego modelu np. określenia rzędu stopni w hierarchii

elementów obiektu, liczby i złożoności zmiennych, dziedziny zmiennych, dokładności odwzorowań (dynamika – odstępów czasowe przeprowadzania pomiarów oraz dopuszczalne błędy) i sposobów badania zgodności wyników otrzymywanych z wykorzystaniem modelu z założeniami (metody weryfikacji) i systemem rzeczywistym (dostępnym do obserwacji, oryginałem) systemem (metody walidacji)

(3) Określenia (wyodrębnienia) zmiennych i ich charakteru (obserwowalne, sterowalne, niesterowalne, symptomy działania), wykonanie obserwacji systemu rzeczywistego (oryginału) lub dostępnego w postaci teorii (dostępnego opisu),

(4) Określenia struktury modelu i jego elementów (na podstawie obserwacji klasy zachowania się systemu (wzrastanie, zanikanie, stabilność, oscylacje, nakładanie się elementarnych przebiegów wzorcowych itd.) w odpowiedzi na wymuszenia – sparametryzowanie sprzężeń zwrotnych (dodatnie, ujemne itd.), określenie zmiennych: wejściowe, wyjściowych, zmiennych stanu, badanie zgodności zachowań modelu i oryginału (w rozważanych przedziałach wartości zmiennych, stopie błędów/zakłóceń, wiarygodności), konfigurowanie powiązań między modułami modelu, ustalenie parametrów (współczynników),

(5) Wykonania czynności mających na celu ocenę modelu (przyjęcie – odrzucenie względnie skierowanie do dalszych badań). Obejmują one weryfikację, walidację itd.

Symulacja tożsama jest z czynnością imitowania, czyli działania w podobny sposób jak system rzeczywisty. Choć nawet komputerowy edytor tekstu można traktować jako symulację długopisu i kartki papieru, to niniejsza praca dotyczy **modelowania systemów**, czyli więcej niż imitowania funkcji i odwzorowania struktury i relacji między elementami systemu rzeczywistego oraz wykorzystania modeli do komunikacji międzyludzkiej (np. w dydaktyce zarządzania i grupowym podejmowaniu decyzji).

Jensen i Bard [133] definiują symulację jako podejście odmienne od programowania matematycznego i modeli stochastycznych, polegające **na tworzeniu przez analityka modelu systemu**.

2.2 Cele zastosowania symulacji

Decyzje podejmowane z wykorzystaniem symulacji komputerowej wykazują przewagę nad opracowanymi wyłącznie za pomocą metod analitycznych szczególnie w warunkach systemów o dużej złożoności. Law i inni [161] podkreślają, że **model symulacyjny zastępuje system rzeczywisty** w tym sensie, że jego użytkownik może rozważać **wiele wariantów działania w systemie rzeczywistym**, (oceniać zjawiska z przeszłości i wyznaczać najbardziej obiecujące postępowanie w niepewnej przyszłości) bez potrzeby eksperymentowania z systemem rzeczywistym co zwykle jest nieopłacalne lub nawet niewykonalne (np. konieczność przerywania funkcjonowania systemu rzeczywistego, nieporównywalność warunków otoczenia w kolejnych eksperymentach z systemem rzeczywistym). **Kolejnym argumentem** za symulacją komputerową jest doskonale przystosowanie komputerowych modeli symulacyjnych do prowadzenia analizy wrażliwości. Analiza wrażliwości polega tutaj na zmienianiu wybranych parametrów modelu oraz zmiennych sterowalnych (wejściowych) i obserwacji reakcji modelu na te zmiany (takie postępowanie można wykonać nawet wykorzystując specjalizowane biblioteki zawarte w pakiecie MS Excel) oraz rejestracji wskaźników przedstawiających stopień ich optymalności (np. porównując z już rozpatrzonymi wariantami lub założoną wartością progową powyżej/powyżej której rozwiązanie uważane jest za godne dalszych badań). Dzięki przeprowadzeniu (na podstawie opracowanego planu) szeregu eksperymentów można wnioskować o zależnościach między zmiennymi modelu a przede wszystkim dokonać wstępnej weryfikacji wielu alternatywnych strategii (w stosunkowo krótkim czasie) oraz wybrać spośród nich najodpowiedniejszą. Ze względu jednak na możliwość przeprowadzenia w określonym czasie jedynie ograniczonej liczby eksperymentów (tożsamy z wyznaczeniem efektów konkretnego wariantu decyzyjnego) **nie można oczekiwać** (w przeciwieństwie do metod analitycznych) gwarancji wyznaczenia **rozwiązania optymalnego**. Powyższa wada staje się jeszcze bardziej dotkliwa gdy w modelu występuje czynnik niepewności; w takiej sytuacji każdy wariant decyzyjny musi być dodatkowo wiele razy wprowadzany podobnie do programu symulacyjnego a zatem będzie wygenerował wiele rezultatów, co skutkować będzie koniecznością statystycznych analiz wyników eksperymentów nawet pojedynczych zestawów zmiennych decyzyjnych. Jest to cecha wspólna wszystkich metod przeszukiwania abstrahujących od kontekstu problemu (podobnie jest w przypadku algorytmów genetycznych czy *tabu search*). Można jedynie oczekiwać wsparcia (tzn. zmniejszenia ryzyka rozpatrywania i wyboru rozwiązania generującego niezadowolające wyniki) w zadaniu wyznaczenia nieznacznego liczebnie zbioru rozwiązań (wariantów decyzyjnych) będących kandydatami na ostateczne rozwiązanie (wdrożone w praktyce).

Sporządzony na potrzeby eksperymentów symulacyjnych model **umożliwia ocenę funkcjonowania systemu rzeczywistego** za pomocą **obliczeń komputerowych**, numerycznych. Szapiro [293] podkreśla, że wynik modelowania (model) umożliwia przeprowadzanie eksperymentów

„bez zakłócania dynamiki rzeczywistości” tzn. bez umieszczania podmiotu (decydenta) w środku realnych procesów (eksperymenty na „żywym” organizmie firmy) co mogłoby go narazić na **niezbyt przyjemne następstwa** (np. finansowe) błędnych decyzji, które mógłby podjąć bez eksperymentalnego sprawdzenia ich przewidywanych skutków. Natomiast dzięki modelowaniu i symulacji może on prześledzić przewidywane efekty znacznej liczby wariantów decyzyjnych i przemyśleć zasady wyboru najbardziej korzystnego (wg przyjętego przez podmiot kryterium). Symulacja związana jest z modelowaniem systemu i wykonywaniem eksperymentów polegających na **obserwacji modelu** w czasie [125]. Symulacja za pomocą modelu (komputerowego, analogowego lub posiadającego postać gry przeznaczonej do rozgrywania przez ludzi) składającego się z **komponentów odwzorowujących** (na różnym stopniu dokładności) **poszczególne elementy i aspekty zachowań**, umożliwia **obserwację zachowań** (czyli zmian w funkcji czasu) analogicznych do zachowań systemu rzeczywistego (niekoniecznie materialnego czy posiadającego sztywne ramy – może to być np. firma wirtualna) poddawane go różnicowanym wartościom wejścia przy różnych wartościach stanu początkowego. Findeisen i inni [84] określają symulację jako „posługiwanie się modelem dynamicznym w celu otrzymania sekwencji wyników, które z jakimś prawdopodobieństwem mogą mieć miejsce w systemie rzeczywistym”. Co więcej, zadaniem symulacyjnych procesów modelowania jest **przewidywanie** (oszacowanie następstw) **wyników poszczególnych wariantów decyzyjnych** poprzez przeprowadzenie **serii wirtualnych** (służących doskonaleniu prognozy) **eksperymentów** (prowadzonych na podstawie reprezentacji opracowanego modelu a nie rzeczywistego systemu czy sytuacji). Według Swaina [290] zastosowanie symulacji w szerszym kontekście wsparcia analizy możliwe stanie się dzięki integracji z innymi modułami SWD. Integracja ta przejawia się coraz większą zdolnością do wymiany informacji między modułem skonstruowanym z wykorzystaniem pojęć modelowania symulacyjnego, technikami sztucznej inteligencji oraz zasobami informacyjnymi przedsiębiorstwa takimi jak dane z rachunkowości czy bazy danych.

2.3 Symulacja a optymalizacja

Symulację od wielu technik analitycznych odróżnia **skoncentrowanie na odwzorowaniu procesów i zdarzeń** mających miejsce w systemie rzeczywistym a **nie** na poszukiwaniu **optymalnego rozwiązania**, chociaż analiza decyzyjna, w szczególności analiza wrażliwości prowadzona z wykorzystaniem modelu symulacyjnego może prowadzić do wyboru rozwiązania **w pewnym sensie optymalnego**. Arsham [14] porównując modele matematyczne, analogowe (fizyczne) i symulacyjne dochodzi do wniosku, że modelowanie symulacyjne (szczególnie wsparte modułem wizualizacyjnym) **wykazuje przewagę nad pozostałymi podejściami**, polegającą na znacznie **większej realistyczności i przydatności** w procesie podejmowania decyzji szczególnie w problemach o cechujących się **dynamiką i probabilistyką**. Wadą modeli matematycznych (wg Arshama) są wymagania stawiane menedżerom polegające na konieczności czasochłonnego szkolenia i dokładnego zapoznania się z modelem, co może prowadzić do braku akceptacji dla takich modeli. Z drugiej strony, jak twierdzi Arsham [14], aby modele analogowe były wystarczająco elastyczne należy je **znacznie upraszczać**, co często prowadzi do wyznaczenia rozwiązania nieprzydatnego w rzeczywistych warunkach.

Jak wskazują Moore i Weatherford [207], chociaż zarówno **modele optymalizacyjne** (np. programowanie liniowe) jak **modele symulacyjne** są modelami ilościowymi, **to zasadniczo różnią się** jeśli chodzi o rozumienie zmiennych decyzyjnych. W modelach decyzyjnych wyróżnić można bowiem **pewien zbiór zmiennych**, które reprezentują podejmowane decyzje (parametrów, zmiennych sterowalnych i niesterowalnych), **zbiór ograniczeń** oraz **funkcję celu** (służącą do pomiaru stopnia osiągnięcia oczekiwań determinowanego przez podjętą decyzję). W modelach optymalizacyjnych **zmienne decyzyjne mają charakter wyjściowy**, czyli poszukuje się takich ich wartości, które **optymalizują wartość funkcji celu**. Natomiast w modelach symulacyjnych **badana się wpływ zmiennych decyzyjnych** - a zatem posiadają one **charakter wejściowy** - na wartość **funkcji celu**. W modelach symulacyjnych postępowanie badawcze dotyczy **ewaluacji** (tj. określenia ich jako akceptowane, do odrzucenia, do rozpatrzenia, bez związku z tematem) wielu decyzji opisanych za pomocą wartości zmiennych decyzyjnych poprzez obserwację wyników symulacji. Nie należy jednak rozpatrywać całkowicie rozdzielnie symulacji i optymalizacji. Arsham [14] wyraża pogląd że chociaż symulacja może być stosowana jako podstawowa technika projektowania złożonych systemów, to **techniki optymalizacyjne umożliwiają efektywne wykorzystanie jej zalet**. Nie jest to łatwe zadanie, gdyż wyniki uzyskiwane w toku przebiegów eksperymentów symulacyjnych charakteryzują się stochastycznością oraz **trudnymi do uchwycenia relacjami** między parametrami poszczególnych zdarzeń (momentami wystąpień) a wartościami zmiennych decyzyjnych.

Charakterystyczną cechą modeli symulacyjnych jest **dążenie do ograniczenia liczby cech** (atrybutów) obiektów systemu odwzorowanych w modelu (rozumianego jako opisu zależności przyczynowo-skutkowych, czyli interakcji między parametrami, zmiennymi sterowalnymi i

niesterowalnymi uzupełnionymi o ograniczenia nałożone na ich wartości) do rzeczywiście **niezbędnych i posiadających znaczenie dla rozwiązania** (oceny) problemu (zagadnienia). Przykładowo model kolejkowy (systemu zdarzeń dyskretnych) placówki bankowej w celu rozwiązania problemu zbyt długich kolejek będzie uwzględniał informacje o oczekiwanych przez danego klienta usługach i statystycznym rozkładzie czasu ich realizacji oraz odstępach czasu między przybyciami klientów, natomiast można postawić hipotezę, że nieistotna jest waga klienta, płeć czy kolor włosów. Zadaniem **projektanta modelu symulacyjnego** jest weryfikacja i walidacja opracowanego modelu i decyzje dotyczące odrzucenia lub wprowadzenia danego atrybutu (rozważanej cechy). Komputerowy model symulacyjny realizowany jest w postaci symulatora – oprogramowania symulującego badany system, umożliwiającego podejmowanie działań (np. wieloetapowych decyzji) i obserwacje ich wyników w postaci pewnych ważnych aspektów działania systemu. Symulator może mieć postać specjalnie opracowanego programu, odwzorowującego model pewnego konkretnego systemu lub opisu struktury, dynamiki i interfejsu w komputerowym języku (środowisku modelowania). Ważne jest jednak to, że symulator powinien pozwalać na wykonywanie eksperymentów symulacyjnych przy ponoszeniu stosunkowo niskich kosztów oraz otrzymywanie wyników w możliwie krótkim czasie [207]. Przegląd narzędzi symulacji zamieścił w swojej pracy Swain [292].

2.4 Definicje pojęć modelowania

Model jest uproszczeniem rzeczywistości, pozostawiającym jednak dość aspektów modelowanego (oryginalnego) systemu aby był przydatny w analizie systemu. Arsham [14] definiuje model jako **reprezentację rzeczywistości** wykonaną przez **twórcę modelu zgodnie z własną perspektywą**, akcentuje zatem potrzebę budowania modeli uwzględniających wiele podejść (perspektyw). Odum [223] określa model jako **logiczny opis zachowania się systemu**, a jednocześnie układ powiązanych ze sobą pojęć, procesów zachodzących między elementami systemu takimi jak osoby, artefakty (produkty ludzkiej działalności), informacje. Arsham [14] podkreśla **użyteczny charakter modeli** polegający na uzyskiwaniu z modeli informacji które mogą być zastosowane w rozwiązywaniu problemów występujących w świecie rzeczywistym. Tenże autor wyróżnia następujące **zadania stojące przed człowiekiem modelującym systemy ekonomiczne** dla potrzeb podejmowania decyzji:

- (1) Uzyskanie od decydenta **jasnego wyrażenia założeń** (celów, teorii, elementów, metod rozwiązywania) dotyczących **modelowanego problemu** oraz ich prawidłowe zapisanie (zdefiniowanie umożliwiające wielokrotne wykorzystanie)
- (2) **Sformułowanie zasad**, kierunków prowadzących do realizacji celów organizacji
- (3) Dążenie do **strukturalizacji informacji** o organizacji, samoczynnego nadzorowania jej funkcjonowania w celu optymalizacji mierników jakości jej działania oraz **identyfikacji sposobności rozwoju** działalności (nowe rynki, produkty, technologie, innowacje organizacyjne)
- (4) **Oddzielenie aspektów funkcjonalnych od transakcyjnych**. Chociaż aplikacja (tj. model) mogłaby działać w oderwaniu od transakcji (wymiany materii, energii i informacji) systemu rzeczywistego (działającego na platformie aplikacji typu transakcyjnego), to swoją użyteczność wykazuje, gdy powiązana jest (korzysta) z danych (informacji) pochodzących z systemu rzeczywistego, przetwarza je, a następnie przekazuje z powrotem do aplikacji transakcyjnej.

Pojęcie modelu powiązać można z następującymi cechami [293]:

- (a) Celem tworzenia modelu jest **opisanie rzeczywistości** lub **prezentacja teorii** (stany, zdarzenia, zjawiska, zależności, relacje, elementy itd.)
- (b) **Upraszczeniem rzeczywistości** ze względu na możliwość zastosowania posiadanego aparatu badawczo-eksperymentalnego – przedstawianiem tylko tego co jest istotne dla podmiotu prowadzącego badania, możliwe do opisanie i zrozumienia (przedmiot zainteresowania podczas weryfikacji zgodności modelu z założeniami)
- (c) **Dopuszczalnością** (w pewnym zakresie) **różnic we własnościach obiektów** wyodrębnionych w **świecie rzeczywistym** oraz **własności ich modeli**. Ustalenie i ocena stopnia niezgodności jest przedmiotem zainteresowania **walidacji** (poprawności odwzorowania ze względu na cel modelowania a nie struktury).

Krippendorff [156] definiuje model (symboliczny) jako **uporządkowany zbiór założeń** (twierdzeń) lub **równań** opisujących w uproszczonej formy niektóre aspekty wyniesione z naszych **doświadczeń**. Autor definicji argumentuje zatem, że teoria z której korzystamy niekoniecznie musi być wyrażona w apriorycznie określonej i ścisłej formie. Podobnie Moore i Weatherford [207] zwracają uwagę na fakt, że model „zawsze upraszcza rzeczywistość”. Podkreślają także, że model staje się przydatny w podejmowaniu decyzji po **uzupełnieniu go danymi**. Dostępność modelu umożliwia opisanie zachowania pewnego aspektu rzeczywistości (badanego systemu). Co więcej, dzięki modelom możemy przeprowadzać weryfikację hipotez dotyczących badanych systemów dużo mniejszym nakładem środków (nie tylko finansowych). Nawet prawidłowo zbudowany model sam nie rozwiąże problemu, pozwala natomiast na **uzyskanie** za jego pomocą **wyników** (prognozy zachowania się modelowanego systemu) oraz ich analizę. **Wypełnienie modelu wartościami danych**, pochodzących najczęściej z **oszacowań i przybliżeń** rzeczywistości [207] umożliwia osiągnięcie przez model następujących **własności**:

- (1) Spełnienia wymagań użytkownika dotyczących formy i treści wyników
- (2) Ukonkretnienia i zgodności modelu z badanym systemem w takim stopniu, na jaki pozwalają dostępne dane
- (3) Przydatność w analizie procesów w wybranym przez użytkownika czasie.

Bäck i inni [16] określają modelowanie jako postępowanie mające na celu sporządzenie **opisu systemów, procesów i danych**. Ważną cechą modelowania jest możliwość jego **prowadzenia na wiele sposobów**. Modelowanie posiada zatem **aspekty sztuki** a nie sztywnych zasad. Zgadzają się z tym twierdzeniem także Moore i Weatherford [207] podkreślając jednocześnie konieczność korzystania podczas modelowania z fundamentalnych zasad postępowania takich jak:

- (1) Wykonania badań (studium) otoczenia w celu przygotowania do identyfikacji i **wyboru istoty problemu** spośród wielu danych pochodzących z obserwacji funkcjonowania przedsiębiorstwa (np. rozpoznanie konfliktów wewnątrz organizacji)
- (2) Sformułowania ograniczonej (uwzględniającej aspekty istotne dla procesu podejmowania decyzji) reprezentacji problemu, przy respektowaniu założonej polityki (w razie konfliktu celów) – określającej misję, cele i wizję organizacji; **działania te obejmują określenie wejścia** (zmienne egzogeniczne, w tym zmienne decyzyjne – sterowalne i parametry – na które nie mamy wpływu) i wyjścia systemu (zmienne endogeniczne, w tym wyniki oraz funkcje celu) oraz określenie sposobu ich pomiaru, stopnia złożoności, horyzontu czasowego zastosowania.

Arsham [14] zwraca uwagę że dobry model cechuje się **inkluzywnością** (zawiera to co określa dany problem) oraz **ekskluzywnością** (usunięto z niego wszystko co jest tylko zbędnym balastem i na pewno nie wpływa na rozwiązanie problemu),

- (3) Zbudowanie modelu symbolicznego, uwzględniającego matematyczną postać zależności między zmiennymi.

Smith [279] wyróżnia następujące przesłanki zainteresowań badawczych przedmiotem modelowania:

(1) Modelowanie ukierunkowuje wysiłki intelektualne **na zrozumienie przyczyn i następstw bieżącej sytuacji** oraz na jej komunikatywny opis. Opis bieżącej sytuacji jest podstawą **uogólnienia** modelu na sytuacje przeszłe i przyszłe. Konieczność zrozumienia przyczyn i komunikatywnego opisu teraźniejszości wynika z **charakteru czynności** prowadzonych podczas modelowania i **badania symulacyjnego**: obserwacji zachowania systemu rzeczywistego, wyróżnienia elementów systemu, parametrów i zmiennych które należy badać, prowadzenia rozmów i konsultacji z ekspertami, kierownictwem, pracownikami i klientami organizacji, wartościowania i filtrowania poszczególnych informacji aby wykorzystać te najbardziej istotne dla opisu organizacji. Wykonanie **modelu opisującego badany** system motywuje do poszukiwania i weryfikacji teorii wyjaśniającej działanie systemu oraz formułowania przyczyn ewentualnych niezgodności, dążenia do ulepszenia funkcjonowania systemu, wzbogacania lub modyfikacji teorii

(2) Badanie za pomocą analiz „**co się stanie jeśli ...**” hipotetycznych następstw. Mogą być one także pozornie odległe - przykładem jest zależność poziomu zadowolenia klientów od projektowanych zmian w organizacji produkcji. Dzięki temu można wartościować wprowadzane decyzje w różnych wariantach bieżącej sytuacji (a właściwie decyzji podejmowanych w najbliższym czasie), bez konieczności angażowania środków na realizację poszczególnych wariantów i pozostawienie możliwości dokonania wyboru w najbardziej dogodnej chwili. Dzięki modelowaniu **dokonujemy oceny** przyszłych korzyści i ryzyka związanego z wykorzystaniem poszczególnych sposobności do prowadzenia zmian w systemie

(3) **Modelowanie posiada walory dydaktyczne** (wykazuje przydatność w szkoleniu), gdyż pozwala na prezentację zachowania systemu (organizacji) rzeczywistego na platformie wirtualnej, komputerowej. **Beneficjentami szkolenia** wykorzystującego modelowanie i symulację mogą być zarówno członkowie kierownictwa (omawianie prognozowanych skutków projektowanych zmian) jak i członkowie zespołów roboczych. Moore i Weatherford [207] dowodzą, że ze względu na zmniejszenie kosztów i czasu potrzebnego na zdobycie doświadczenia, **wykorzystanie symulacji w nauczaniu** jest motywowane potrzebą **szybkiego pozyskania wiedzy** potrzebnej do **osiągnięcia doświadczenia** niezbędnego w podejmowaniu decyzji

4. Modelowanie służyć może także jako **metoda dowodzenia i potwierdzenia wartości** proponowanych **rozwiązań**. Wnioskowanie o przydatności danego wariantu decyzyjnego może być przeprowadzone na podstawie zestawień i wykresów składających się z informacji o stanie bieżącym systemu oraz wyników symulacji (najlepiej także wyników pośrednich) dla poszczególnych wariantów. W ten sposób można przedstawić prognozowane efekty proponowanych zmian i dokonać właściwego wyboru.

2.5 Modelowanie – klasyfikacja

Savolainen i Salminen [261] opracowali następującą **klasyfikację sposobów modelowania** wyróżniając pięć (sześć jeśli traktować oddzielnie modelowanie finansowe i optymalizację procesów) kategorii:

1. **Modelowanie procesów biznesowych** (*business process modelling*). **Proces biznesowy** jest “zbiorem zadań znajdujących się ze sobą w logicznych relacjach, wykonywanych w celu osiągnięcia określonych skutków ekonomicznych (biznesowych)”. Modelowanie polega na **sporządzeniu opisu wszystkich łańcuchów zadań biznesowych** czyli przedsięwzięć, zdarzeń, etapów, obiektów uczestniczących w procesie wytwarzania produktów oraz rejestracji, przetwarzania i przesyłania informacji z nimi związanych. Modelowanie to przeprowadzane jest w etapie wstępnym i jego celem jest opracowanie i realizacja nowych lub zdefiniowanych procesów biznesowych (nowych zasad funkcjonowania przedsiębiorstwa). Modele powstałe dzięki modelowaniu procesów biznesowych stanowią bazę dla innych podejść i mogą zostać wykorzystane dla porównywania wyników (w wymiarze jakości systemu) organizacji po przeprowadzeniu symulowanego BPR. *Business process reengineering* to przeprojektowanie na poziomie procesów wytwarzania lub linii produkcyjnej, zgodnie z następującymi regułami [57]: (1) celem jest radykalna zmiana a nie stopniowe zmiany, (2) korzystanie z możliwości rozpoczęcia wszystkiego od nowa (ang. *clean slate*), (3) odpowiedzialność ponosić będą członkowie naczelnego kierownictwa. Przykładami narzędzi BPM są Wall Charts zgodne z modelem IDEF (metoda modelowania decyzji, działań i czynności przebiegających w organizacji, przechowująca zdobytą wiedzę w hierarchicznie zorganizowanych modelach za pomocą sieci bloków reprezentujących funkcje i strzałek reprezentujących interfejsy), CIM-OSA [136] (architektura systemów otwartych zawierająca zintegrowaną metodologię wspomagania wszystkich faz cyklu życia systemów CIM, czyli specyfikacji wymagań, projektowania systemowego, wdrożenia i utrzymania systemów).

2. **Modelowanie danych** (*data modeling*). Podejście to stosowane jest w **projektowaniu baz danych**, mające na celu ustalenie odpowiednich do informacji pochodzących ze środowiska rzeczywistego reguł przechowywania (kodowanie zgodne z opracowanym modelem) i dostępu do danych co stanowi jeden z warunków podniesienia jakości transakcyjnego przetwarzania danych (zajmuje się tym modelowanie programów). Narzędzia modelowania danych korzystają przede wszystkim z **obiektywnej analizy danych** i **projektowania obiektywego**, diagramów encji (*entity-relationship diagrams*), słowników danych (repozytoriów definicji).

3. **Modelowanie programów** (*program modelling*). Celem tego podejścia jest stworzenie systemów informatycznych **czasu rzeczywistego**, przetwarzających dane i informacje pozyskiwane z baz danych, otoczenia oraz maszyn i urządzeń wyposażonych w **elementy umożliwiające komputerowe przetwarzanie danych** (czujniki przesyłające parametry procesu produkcyjnego i urządzenia

sterowane komputerowo). Narzędzia wspierające modelowanie programów korzystają z analizy i projektowania obiektowego a przykładem jest MetaEdit+, zgodny z podejściem analizy i projektowania strukturalnego [333] a także Real-Time SA/SD; Business Systems Planning firmy IBM Corp., Value Chains & Value Systems Portera, Activity Analysis Portera.

4. **Modelowanie finansowe i symulacja** (*financial modelling and simulation*) oraz **modelowanie i optymalizacja procesów** (*modelling and optimization of processes*). Zasadniczym zadaniem modelowania finansowego jest **prognozowanie wielkości wyniku finansowego**, zyskowności, kosztów w zależności od projektowanych decyzji produkcyjnych, inwestycyjnych i kapitałowo-majątkowych. Sporządzenie zestawienia prognozowanych wyników poszczególnych projektów jest wstępem do **optymalizacji decyzji**. Optymalizacja prowadzona jest na podstawie założonych kryteriów, korzystających z **analiz dyskryminacyjnych**, **wartości wskaźników**, obserwowanych trendów, dopuszczalnego stopnia ryzyka. **Modelowanie finansowe** może być prowadzone z wykorzystaniem szeregu **narzędzi informatycznych**, np. arkuszy kalkulacyjnych wyposażonych w możliwość programowania makr, tworzenia **interfejsu użytkownika** i dostępu do baz danych, narzędzi rozwiązywania **zadań programowania liniowego** oraz **analiz scenariuszy symulacyjnych** typu „co się stanie jeśli?”. **Podstawowa trudność** w osiągnięciu **rozwiązań optymalnych** (dotyczących planowania zasobów i ich alokacji, systemu sterowania procesami [16]) dla problemów w rzeczywistych systemach, których modele cechuje wielopoziomowa złożoność i wynikająca z niej konieczność przeprowadzenia znacznej liczby obliczeń polega na **konieczności ujęcia wszystkich** (choć o różnym stopniu istotności) czynników wpływających na wyniki działania systemu. Dlatego **praktycznie niemożliwa jest bezpośrednia** (tzn. opisana za pomocą formuł matematycznych) **optymalizacja struktury i parametrów elementów systemu**. Wyjścia z tej sytuacji są następujące: **stworzenie symulacyjnego modelu** systemu, co umożliwia badanie następstw poszczególnych decyzji będących reprezentacją przestrzeni rozwiązań, **optymalizacja określonego wycinka** działalności lub pogodzenie się z **ograniczoną wiarygodnością przybliżonych** (wartości atrybutów elementów) czy uproszczonych (struktura) modeli optymalizacyjnych.

5. **Modelowanie produktu** (*product modelling*). Modelowanie produktu ma na celu **ustalenie ram i procedur projektowania** nowych i **unowocześniania** obecnych produktów, zgodnie z **podejściem QFD** (*Quality Function Deployment*) dążącym do **podniesienia jakości** projektowanych, wytwarzanych, modyfikowanych i sprzedawanych produktów w celu zaspokojenia potrzeb klientów. Zajmuje się wsparciem poszczególnych faz cyklu życia i cech (atrybutów) produktu, aby **zwiększyć konkurencyjność** produktu, dzięki, jak stwierdza Kuutti [157], zmniejszeniu kosztów materiałowych, czasu pracy ludzkiej niezbędnego do wytworzenie produktu (przeprojektowania procesu technologicznego i organizacji produkcji), bardziej intensywnemu korzystaniu z podwykonawców (specjalizacja i optymalizacja kosztów), zmniejszeniu czasu wprowadzenia produktu na rynek równoległe wykonywanie prac projektowych, inwestycyjnych i zmian technologicznych równoległe,

zmniejszeniu ilości braków wynikających z błędów na etapie projektowania produktu. Zmniejszenie stopy braków osiągnąć można (jest to postulat) przez zastosowanie narzędzi kontrolujących i weryfikujących prace wykonywane przez projektantów, udostępniające gotowe moduły, bloki, modele zasobów przedsiębiorstwa i linii produkcyjnych: maszyn, urządzeń, centrów obróbczych, materiałów, narzędzi itp., korzystanie z modelowania wizualnego i procedur obliczeniowych (analiz) wykonywanych na żądanie. Podstawowe **cele modelowania produktu tożsame są z następującymi ideami reengineeringu** [37] za [78]:

- a) Ukierunkowanie procesów uruchamianych w przedsiębiorstwie na cele (w opozycji do zadań, które przeważają w podejściu funkcjonalnym),
- b) Wiodąca rola klientów składających zamówienia do realizacji przez wykonawców (serwery) jako instancji oceniającej działanie oraz organizacja typu modułowego, charakteryzująca się przyporządkowaniem poszczególnych decydentów (kontrolerów) do komórek organizacji,
- c) Równoległość działań zapewniająca większy stopień wykorzystania zasobów organizacji,
- d) Działania mające na celu osiągnięcie przewagi konkurencyjnej poprzez „kompresję” przestrzeni i czasu (czyli wirtualne skoncentrowanie zasobów, eliminacja prac niepotrzebnych, wzrost znaczenie dostępnej informacji i posiadanej wiedzy, *concurrent engineering* – równoległość projektowania i wytwarzania).

Istotną cechą podejścia modelowania produktów jest **ściśła współpraca poprzez wymianę informacji i danych** między zespołem odpowiedzialnym za **projektowanie produktu** i zespołem **kierującym systemem wytwórczym** przedsiębiorstwa (istotne korzyści można uzyskać dzięki oprogramowaniu typu *groupware* [57]).

2.6 Znaczenie uwzględnienia losowości w modelach

Kryterium określające możliwość przewidzenia wyników pochodzących z modelu przy pełnej znajomości stanu obiektów reprezentowanych w modelu i wartości wejściowych (wymuszeń). Wyróżnić można modele probabilistyczne oraz deterministyczne. Przede wszystkim należy zauważyć [14] podstawową różnicę (z punktu widzenia metodologii podejmowania decyzji) że następstwem właściwie, prawidłowo podjętych decyzji w **warunkach deterministycznych** będą **wyniki optymalne** (lub najlepsze z możliwych do osiągnięcia spośród wyznaczonych alternatyw decyzyjnych) natomiast w **warunkach niedeterministycznych** nie można być pewnym, czy decyzje podejmowane zgodnie z uznaną metodologią zawsze będą skutkowały wynikami **najlepszymi z możliwych**. Należy jednak pamiętać, że **praktycznie stosowane modele** nie są ani całkowicie probabilistyczne ani całkowicie deterministyczne. Przykładem może być wpływ warunków pogodowych na zachowanie (deterministycznego z natury) modelu komunikacji kolejowej lub lotniczej a z drugiej strony zastosowanie reguły (nawet o charakterze rozmytym) w probabilistycznym modelu rynku papierów wartościowych. **Modele deterministyczne**, zakładające pełną i pewną znajomość wszystkich niezbędnych zmiennych wejściowych, pozwalają na dokładne przewidzenie (prognozowanie) wartości i zachowania się wyników pochodzących z modelu. Charakterystyczną cechą **modeli deterministycznych**, jak podkreślają Moore i Weatherford [207], jest wykorzystywanie ich w **dobrze rozpoznanym środowisku** (tzn. takim w którym może występować niepewność dotycząca mniej znaczących zmiennych wejściowych modelu), przede wszystkim wewnątrz przedsiębiorstwa. **Motywacje** towarzyszące wykorzystywaniu **modeli deterministycznych** (oprócz możliwości ich analitycznego rozwiązywania) są następujące [207]:

- a) umożliwiają sformułowanie zagadnień problemowych z obszaru zarządzania (poczynając od programowania liniowego kończąc na rozważaniu skonfliktowanych celów),
- b) modele deterministyczne przystosowane są do uwzględniania ograniczeń narzuconych na poszukiwane rozwiązania,
- c) dostępność efektywnego oprogramowania optymalizującego,
- d) bezpośrednia użyteczność rozwiązań uzyskiwanych za pomocą tych modeli (otrzymuje się jednoznaczny przepis dotyczący ustawienia wartości zmiennych decyzyjnych),
- e) w większości sytuacji sformułowanie ograniczeń i optymalizacja funkcji celu **są tożsame** ze zbudowaniem modelu,
- f) doświadczenie zdobyte podczas formułowania i rozwiązywania modeli deterministycznych jest przydatne także podczas prac nad konstrukcją także innego rodzaju modeli (probabilistycznych i symulacyjnych).

Modele probabilistyczne (stochastyczne) **nie pozwalają** na przewidywanie wyników z **całkowitą pewnością**. Dzieje się tak dlatego, że modele probabilistyczne zakładają nieznaną

dokładnej (i pewnej) wartości co najmniej jednej istotnej zmiennej wejściowej przed podjęciem decyzji. Stosując model probabilistyczny godzimy się, jak zauważają Moore i Weatherford [207], na włączenie do procesu podejmowania decyzji **ignorancji** (nieznajomości stanu faktycznego). Niepewność dotycząca faktycznych wartości zmiennych wejściowych ujmowana jest w modelu zwykle poprzez uwzględnienie **czynnika losowości**. Dlatego też modele probabilistyczne zawierają **zmienne losowe**. Czynniki losowe reprezentują w modelu wielkości występujące w systemie rzeczywistym, które nie mogą (lub jest to nieopłacalne) być uwzględnione w sposób pozwalający na sformułowanie deterministycznego wpływu na wartości wyjściowe systemu rzeczywistego. Losowość w modelach probabilistycznych przedstawiona jest za pomocą parametrów danego typu rozkładu losowego (nie zawsze teoretycznego!). Każde przedsiębiorstwo już w chwili powstawania działa w warunkach niepewności. Przykładowo, firma emitując na początku działalności akcje nie może z całą pewnością stwierdzić, czy rynek będzie znajdował się w fazie wzrostu czy spadku cen. Powinna zatem uwzględnić obie możliwości planując warianty działalności inwestycyjnej, produkcyjnej i handlowej którą chciałaby uruchomić (rozвивać) dzięki funduszom pozyskanym z emisji akcji.

Chociaż modele deterministyczne nie zawierają zmiennych i parametrów endogenicznych o charakterze probabilistycznym to **można badać wpływ na ich funkcjonowanie**, czyli zachowanie wartości wyjściowych (wyników) na pochodzące z rozkładu losowego wartości wejściowe. **Ekspertyzy symulacyjne** zgodne z techniką **Monte Carlo** polegają na obserwacjach wyników otrzymanych z modelu. Wyniki są następstwem wielokrotnego losowania wartości zmiennych wejściowych pochodzących z generatora liczb losowych o częstościach zgodnych z założonym (badanym) teoretycznym rozkładem zmiennych losowych (np. rozkładem dyskretnym, o ograniczonej liczbie możliwych stanów). W pracach nad techniką Monte Carlo miał swój udział Stanisław Ulam [194]. Monte Carlo jest techniką numerycznego rozwiązywania równań różniczkowych. Jej zastosowania w zakresie finansów dotyczą głównie wyceny instrumentów pochodnych oraz oszacowania wartości narażonej na ryzyko (*value-at-risk*) portfela inwestycyjnego [129]. Według Kuziakowej [158] *value-at-risk* jest to „strata wartości rynkowej taka, że prawdopodobieństwo osiągnięcia jej lub przekroczenia w zadanym (przewidywanym) przedziale czasowym jest równe zadanemu (przez analityka) poziomowi tolerancji”. Znając wartość *value-at-risk* można spodziewać się że wartość straty w danym okresie jej nie przekroczy (z danym prawdopodobieństwem). Jak pisze Myerson [211] **Value-at-Risk** jest **miarą ryzyka**, rozumianą jako taki poziom zyskowności (przedsięwzięcia) dla którego skumulowane prawdopodobieństwo zyskowności mniejszej niż ten poziom jest nie większe niż przyjęta stosunkowo mała liczba (np. 5%). Wyrażona w ten sposób została reguła, że osiągnięcie odpowiednio wysokiego poziomu zyskowności może być mało prawdopodobne lub że statystycznie rzecz ujmując łatwiej zapewnić niski stopień zyskowności. Monte Carlo jest zwykle traktowana jako *ostatnia deska ratunku* gdy nie istnieją inne (analityczne) metody znalezienia rozwiązania problemu, ze względu cechującą ją *brutalną siłę*. Niestety większość problemów z

zakresu finansów jest na tyle złożonych że technika ta znajduje często zastosowanie. Dzięki zastosowaniu techniki Monte Carlo otrzymuje się **zestaw wyników** (tablicę) zawierającą wylosowane wartości zmiennych wejściowych oraz odpowiadające im wyniki eksperymentu (w ogólnym przypadku jest to przekształcenie wartości skalarnej (losowanej) w wektor wyników). Analiza wyników eksperymentów może być przeprowadzana np. w celu statystycznej (obejmującej jedną lub więcej wartości statystyk) weryfikacji hipotezy o wartości średniej badanego wyniku. Ważną własność techniki symulacyjnej Monte Carlo polega na tym, że jej precyzja jest **proporcjonalna do pierwiastka kwadratowego liczby badanych wariantów** [59], stąd obiecujące efekty daje jej połączenie z technikami zmniejszania wariancji (*variance reduction techniques*). Powyższe postępowanie analityczne zgodne jest z ze sformułowanym np. przez Arshama [14] krokami **decyzyjnej analizy kosztów i zysków**:

- (1) Określenie zbioru dopuszczalnych wariantów decyzyjnych
- (2) Przypisanie wartości zysku/straty oraz prawdopodobieństwa każdemu wariantowi decyzyjnemu
- (3) Wyliczenie wartości oczekiwanej każdego wariantu decyzyjnego
- (4) Wdrożenie wariantu decyzyjnego o najwyższej wartości oczekiwanej.

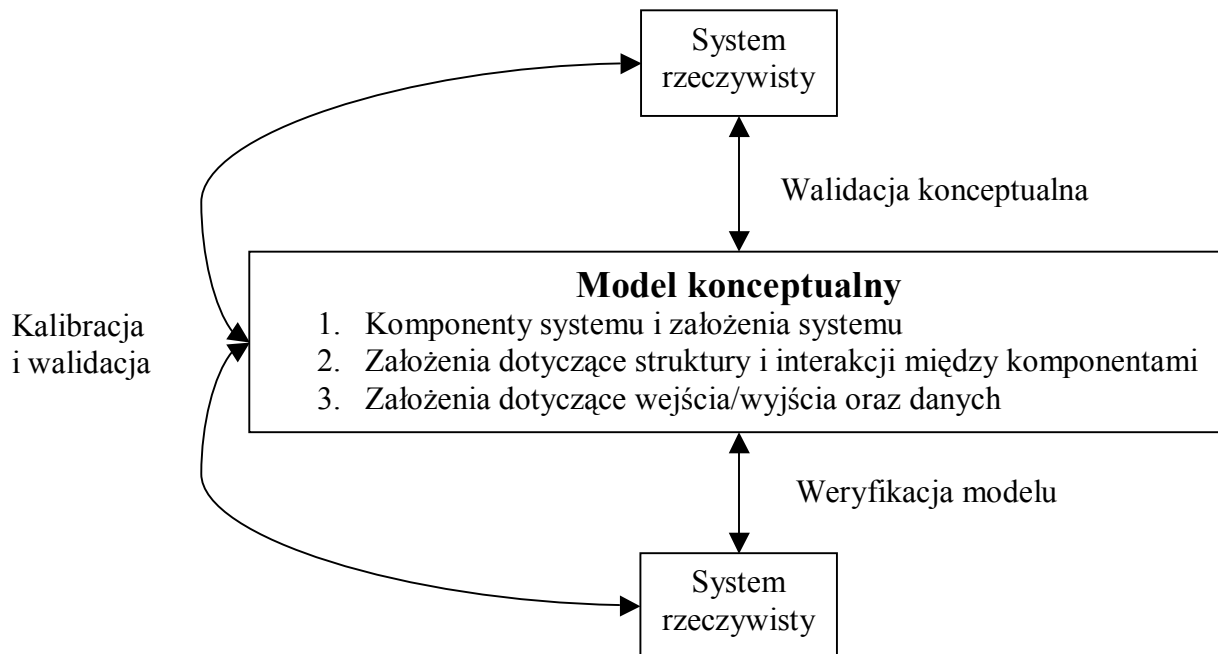
2.7 Budowa modelu symulacyjnego

2.7.1 Model konceptualny: weryfikacja i walidacja

Na proces powstawania modelu systemu rzeczywistego dla potrzeb eksperymentów symulacyjnych składają się zespoły czynności, wykonywane w odpowiedniej kolejności [325]:

1. **Obserwacja zewnętrznych przejawów** funkcjonowania systemu rzeczywistego (określenie jaka dziedzina wiedzy może wyjaśnić działanie systemu)
2. **Próby określenia części** z których składa się **system i ich oddziaływań** między sobą (zastosowanie wiedzy dziedzinowej (teoretycznej i praktycznej), wyodrębnienie podsystemów, elementów, relacji, określenie struktury systemu) – korzystanie z wiedzy ekspertów
3. **Zebranie danych w celu określenia zachowania systemu** (wybór sposobu opisu dynamiki systemu, zrozumienie działania systemu, wyznaczenie postaci opisu matematycznego, określenie wartości parametrów).

Powyższe czynności badawcze prowadzą do opracowania **modelu konceptualnego**. Model konceptualny wyraża za pośrednictwem wybranego medium **relacje między postrzegalnymi elementami systemu** oraz informacji przepływających między nimi i wartości danych opisujących (charakteryzujących) stan poszczególnych elementów. Rolę **medium reprezentacji** modelu konceptualnego może odgrywać język naturalny, język **matematyczny**, przedstawienie za pomocą **rysunków**, diagramów lub szerzej - **multimediów**, **programu komputerowego** w wybranym języku programowania, **modelu analogowego**, wreszcie **komputerowego środowiska modelowania** – wykorzystującym własną notację, symbolikę, format i semantykę). Odpowiednio **uszczegółowiony model konceptualny** jest podstawą zbudowania **modelu symulacyjnego**. Testowanie modelu ma na celu wykazanie czy wyniki (przewidywania) modelu **zgodne są z obserwacjami** systemu rzeczywistego pozyskanymi w przeszłości. Analiza wyników eksperymentów symulacyjnych i konfrontowanie ich z zachowaniem systemu rzeczywistego pozwala na **weryfikację i walidację** modelu symulacyjnego. Jeśli ocena istniejącego **modelu symulacyjnego** nie jest **wystarczająco pozytywna** (tzn. nie jest to decyzja o charakterze binarnym) to poszukuje się przyczyny a następnie przystępuje do **dalszych modyfikacji** (ulepszania) modelu konceptualnego: dokonuje się zmian w założeniach, opisie systemu, jego parametrach a nawet zmienia lub dodaje nowe części (elementy) przedstawiające strukturę systemu rzeczywistego tak aby wyniki uzyskiwane z modelu symulacyjnego **lepiej reprezentowały** funkcjonowanie systemu rzeczywistego. Z drugiej strony Shannon [266] podkreśla, że **nie ma możliwości stworzenia modelu idealnie oddającego zachowanie** i strukturę systemu rzeczywistego (może z wyjątkiem cyfrowych układów elektronicznych) i **eksperyment symulacyjny** znajduje zastosowanie tylko w **procesie postępującego zwiększania** adekwatności i dokładności odwzorowania.



Rys. 2-1 Diagram konstruowania modelu symulacyjnego z wyróżnionym etapem weryfikacji i walidacji. Źródło: [325]

Twórców i użytkowników modelu interesuje czy za pomocą modelu można **przewidzieć zachowanie systemu rzeczywistego w różnych warunkach**. Jeśli odpowiedź na powyższe pytanie będzie twierdząca, to model może być wykorzystywany w **rozwiązywaniu problemów** wymagających wielokrotnego, **sekwencyjnego podejmowania decyzji**. Zadaniem **weryfikacji modelu** jest ustalenie czy model konceptualny jest prawidłowo odwzorowany w modelu komputerowym. Zadaniem **walidacji** jest stwierdzenie czy modele: konceptualny i komputerowy są odzwierciedleniem systemu rzeczywistego, tzn. czy model poddany oddziaływaniom (podczas testowania), których charakter i wartość (opisane za pośrednictwem przebiegów mierzalnych wielkości) odpowiada (tzn. czy zachowuje się podobnie) jak poddany analogicznym oddziaływaniom system rzeczywisty (walidacja) lub konceptualny (weryfikacja) [Rys. 2-1].

Weryfikacja systemów informatycznych obejmuje wszelkie czynności podejmowane w **celu ustalenia czy zaimplementowany model funkcjonuje prawidłowo**, tzn. czy jego wyniki (wartości wyjściowe, zmienne stanu) są **takie jak oczekiwano** na podstawie modelu matematycznego. Czynności weryfikujące **nie obejmują oceny odwzorowania rzeczywistego systemu** (czyli istniejącego realnie zbioru elementów powiązanych za pomocą relacji [223]), gdyż zajmuje się tym walidacja. **Weryfikacja symulacyjna** różni się od zwykłej weryfikacji systemów informatycznych (oprogramowania) rozpatrywaniem oprócz zgodności przewidywalnych wyników pochodzących z systemu tzn. otrzymywania tych samych wyników po zastosowaniu takich samych wartości danych wejściowych (na etapie testowania oprogramowania) także spełnienia oczekiwań **dotyczących zachowania się wyjścia** (wyników) systemu na które **oddziałują czynniki losowe** (w rzeczywistości wyniki z generatorów liczb pseudolosowych) – w zależności od kontekstu może to być długość kolejki

do stanowiska obsługi, czas wykonania zadania czy prawdopodobieństwo zaklasyfikowania wyrobu jako braku.

Law i inni [161] **definiują walidację** jako “proces określania czy model symulacyjny jest właściwą reprezentacją systemu (obiektu badań) wypełniającą konkretne cele badań”. Wskazują także na następujące **aspekty procesu walidacji modeli symulacyjnych**:

- (1) Walidację modelu można uznać za udaną wtedy gdy **model można wykorzystać w procesie podejmowania decyzji** w podobny sposób jak **modelowany system** (przy zachowaniu zalet symulacji)
- (2) Przeprowadzanie walidacji modeli systemów o **dużym stopniu złożoności** oraz **wymagających dużo wiedzy** o warunkach obecnych w przyszłości wiąże się z **większą trudnością** niż modeli systemów prostszych i legitymujących się bogatą dokumentacją (opisem funkcjonowania) w przeszłości i teraźniejszości
- (3) Nie jest roztropnym dążenie do „stuprocentowej” walidacji ze względu na naturalną dla wszelkich modeli cechą przybliżonego odwzorowania systemu rzeczywistego, funkcjonującego w określonych warunkach środowiska
- (4) Status pomyślnie **zwalidowanego** model otrzymać może tylko ze **względu na konkretny cel** (zespół celów) badawczych
- (5) Walidacja powinna być **obowiązkowym etapem budowy modeli** systemów rzeczywistych.

Law i inni [161] podkreślają konieczność **przeprowadzenia analizy wrażliwości** na etapie **walidacji modelu** gdyż jej obserwacja jej efektów (rozważenie wyników uzyskiwanych z modelu po systematycznym dokonywaniu zmian) pozwala na zidentyfikowanie niedomagań modelu a także praktyczne określenie szczegółowości odwzorowania systemu rzeczywistego i reakcji na zmieniane definicje czynników losowych (rozkładu prawdopodobieństwa).

Oba procesy: **weryfikacja i walidacja** mogą **przebiegać równolegle** (a nawet przeplatać się). Jednak Moore i Weatherford [207] zwracają uwagę, że sterowanie eksperymentem na rzeczywistym systemie (w celu porównania z modelem symulacyjnym) nie jest zazwyczaj możliwe i jedynym wykonalnym sposobem walidacji modelu jest „przewidywanie przeszłości na podstawie danych z przeszłości jeszcze dawniejszej” (*predict history*). Podejście to polega na traktowaniu jako danych teraźniejszych - danych z przeszłości (decyzje, parametry i wyniki) dotyczących sytuacji podobnych do mogących wydarzyć się w teraźniejszości lub przyszłości. Następnie, po wprowadzeniu danych do modelu symulacyjnego i wykonaniu przebiegów symulacyjnych należy **porównać wyniki otrzymane z modelu z wynikami rzeczywistymi** (z przeszłości). Przyjmując założenie o niezmienności praw w rozważanych okresach, na podstawie zgodności (pozytywnej walidacji) modelu w przeszłości, **wnioskuje się o prawidłowym działaniu** (prognostycznym) także w **przyszłości i przydatności** (otrzymywania za jego pomocą lepszych rekomendacji decyzji, przynoszących odpowiedni zwrot z inwestycji w modelowanie i symulację) danego modelu.

Zalecenia dotyczące przeprowadzenia **weryfikacji modelu** można znaleźć u Wintona [325] a większość zaleceń dotyczy także **procesu walidacji**. **Poniższe zalecenia zostały także zrealizowane w hybrydowym systemie wspomaganie decyzji Ekanwin:**

1. Przeprowadzanie testowania aplikacji (programu komputerowego) przez osobę nie zaangażowaną bezpośrednio w projekt. Osoba taka (tester) powinna jednak **znać system rzeczywisty** a reprezentacja wiedzy zastosowana w modelu być zrozumiała dla testera. W takiej sytuacji będzie możliwe wykorzystanie w poprawionym modelu poszczególnych uwag testera oraz **wychwycenie** przez testera **niedomagań** (niezgodności wyników lub nieoptymalnej drogi otrzymania wyników).
2. Sprawdzenie **zgodności wszystkich możliwych wariantów działania** (wartości stanów i wyjść) modelu z opracowanym diagramem przepływów (*flow diagram*). Mniej rygorystyczny warunek dotyczy **odzworowania** wszystkich gałęzi diagramów przepływów w raporcie z testowania (użytkowania) aplikacji w której zaimplementowano model symulacyjny.
3. Sprawdzenie sensowności wyników dla **rozmaitych kombinacji danych wejściowych**, ich przebiegów i stanu początkowego modelu. Rozważa się tutaj miary i dane statystyczne, **zasady zachowania** (zapobieganie „gubienia” obiektów wchodzących do systemu, błędów przekształcania jednostek, badanie czy otrzymywane dzięki eksperymentom **wartości średnie, wariancja, czasy przebywania obiektów w systemie** mieszczą się w rozsądnych przedziałach).
4. **Rejestracja na bieżąco** i przechowywanie danych wyjściowych w celu sprawdzenia zgodności z oczekiwanymi przez **eksperymentatora wartościami**. Law i inni [161] określają je jako najważniejszą próbę (walidację przez rezultaty – *results validation*) także w procesie **walidacji modelu**, gdyż model z którego uzyskamy wyniki sprzeczne z systemem odzworowanym nie jest godny dalszego rozwoju.
5. Umożliwienie **dołączania objaśnień** w miejscach **komunikowania się z użytkownikiem** (okienka informacyjne i dialogowe).
6. Korzystanie z **animowanych obiektów ekranowych** reprezentujących **zmiany w stanie modelu** i zachodzące w nim procesy – wysyłanie, przemieszczanie i odbieranie informacji lub obiektów symbolizujących zdarzenia, przepływ materii, energii i pieniądza. **Animacja pozwala na szybką kontrolę i reakcję** na nieprzewidziane **odchylenia** od oczekiwanych sekwencji zdarzeń, parametrów czasowych, brak stabilizacji lub nieplanowane oscylacje. Wykazanie (niestety potwierdzenie jest dużo słabszym argumentem) niezgodności wyników modelu i systemu za pomocą wykresów wartości, przebiegów, korelacji oraz animacje graficzne (aspekty dynamiczne) stanowią mocną przesłankę do **negatywnej walidacji modeli** [161]. Z drugiej strony, **pozytywna prezentacja zgodności** może być sugestywnym powodem podwyższenia wiarygodności modelu.
7. Korzystanie z możliwości wykonywania i **obserwacji eksperymentu w trybie krokowym**, przyspieszonym lub zwolnionym, sprawdzanie wartości zmiennych, wizualizacja za pomocą wykresów. Praca krokowa oraz wstrzymanie wykonywania dalszych czynności w razie wystąpienia

określonego zdarzenia czy warunku nałożonego na zmienną (tzw. pułapka) umożliwi dokonywanie modyfikacji (poprawek) wartości zmiennych a nawet zmianę ścieżki wykonywania eksperymentu.

2.7.2 Etapy tworzenia modelu symulacyjnego

Law i inni [161] zaproponowali **siedem zasad budowy modeli symulacyjnych i ich wdrażania**, które (ich zdaniem) powinny zapewnić opracowanie modeli cechujących się starannym przeprowadzeniem **pozytywnej walidacji** oraz **cieszące się zaufaniem** i stosowanych w powodzeniem przez decydentów. Oczywiście, zauważyć można ich znaczne podobieństwo do **krytycznych czynników sukcesu** wdrożenia **systemów eksperckich** [334] jednak w przypadku modeli symulacyjnych **dziedzinowi eksperci** (nazywani także ekspertami w zakresie danego zagadnienia, *subject-matter experts*) **wydają ocenę poprawności**, wiarygodności oraz **stwierdzają stan walidacji modelu** (zamiast współdziałać przy jego opracowaniu jako obiekty akwizycji wiedzy).

Wyróżnić zatem można następujące etapy prowadzenia badań symulacyjnych (łącznie z budową modeli):

- (1) **Formułowanie problemu**; najistotniejsze jest w tym przypadku określenie problemów (pytań) w których oczekiwać będziemy wsparcia środowiska symulacyjnego (modelu, danych, oprogramowania) oraz jakie zasoby będą potrzebne
- (2) Wyartykułowanie wystarczająco szczegółowego (ze względu na cele modelu uwzględniając dostępne środki i ograniczenia) **modelu konceptualnego** (układu założeń, algorytmów i danych, diagramy, opis modułów), akwizycja danych oraz ich przetwarzanie (np. zastępowanie zbioru danych agregatami – wynikami obróbki statystycznej – rozkładami prawdopodobieństwa)
- (3) Ocena **poprawności** (walidacja) modelu na podstawie modelu konceptualnego z udziałem odpowiedzialnych w organizacji za innowacje (oraz ekspertów dziedzinowych). W razie **oceny negatywnej** należy powrócić do etapu 2)
- (4) Wybór narzędzia programowania, **wykonanie i zweryfikowanie** (ocena zgodności z modelem konceptualnym) **komputerowego modelu symulacyjnego**
- (5) **Walidacja działającego programu** (komputerowego modelu symulacyjnego) przez odpowiedzialnych za projekt (oraz ekspertów dziedzinowych) z **wykorzystaniem wyników analizy wrażliwości** (wg planu eksperymentu) jako weryfikatora zgodności poszczególnych zależności wynikających z modelu konceptualnego oraz systemu rzeczywistego oraz orzeczenie o „zewnętrznej poprawności” (*face validity*)
- (6) **Opracowanie** i wykonanie **serii eksperymentów symulacyjnych** (zwracając uwagę na długość czasu przebiegu, warunki zakończenia, okres rozgrzewania oraz liczbę powtórzeń przy stałych wartościach wejściowych i parametrach) i na tej podstawie wydanie opinii o ewentualnej konieczności

dokonania poprawek

(7) Krytyczne **wykonanie szczegółowej dokumentacji** i prezentacja (na forum osób odpowiedzialnych za projekt) w celu oceny wiarygodności i domniemanej poprawności (uprawomocnieniu, walidacji).

2.7.3 Wskazówki metodologiczne i techniki walidacji

Wynik walidacji modelu jest **zawsze kategorią względną** [14], gdyż dotyczyć może tylko **określonych wartości utylitarnych** (nie można w modelu zawrzeć wszystkich szczegółów odwzorowanego systemu). Jak podaje Arsham [14] przeprowadzanie walidacji składa się z dwu etapów: **stwierdzenia logicznej poprawności modelu** oraz **stwierdzenia zgodności wyników z systemem rzeczywistym**.

W trakcie **sprawdzania wewnętrznej logicznej poprawności** wykonywane są następujące czynności:

- (1) **Porównanie wyników** otrzymywanych za pomocą modelu komputerowego i obliczeń wykonywanych ręcznie (lub za pomocą narzędzia uznanego za wiarygodne)
- (2) **Obserwacja wyników** obliczeń otrzymywanych na wyjściu poszczególnych modułów modelu
- (3) Rozpatrywanie wyników modelu **w warunkach deterministycznych** lub o **uproszczonej losowości** (po właściwym ustawieniu opcji generowania zmiennych losowych w interfejsie modelu) w celu uproszczenia obliczeń sprawdzających
- (4) Sprawdzanie działania modelu przez wykonanie eksperymentów z wieloma zestawami testowych (próba określenia zakresu problemów, które model zdolny jest rozpatrywać poprawnie).

Drugi etap, **porównanie wyników modelu i systemu rzeczywistego**, jest wykonywany wtedy, gdy **dostępne są informacje** (historyczne) **zebrane w przeszłości**, dotyczące sygnalizowanych problemów, metod ich rozwiązywania i wyników ich zastosowania. Jak podaje Arsham [14] jeśli wyniki modelu mają postać szeregów czasowych tzn. uporządkowanych **ciągów obserwacji** pewnej **zmieniającej się w czasie wielkości**, to walidację przeprowadza się wykonując **testy statystyczne** (np. t-studenta). Testowane hipotezy dotyczą podobieństwa (niewielkich różnic) między wartościami średnich, wariancji, parametrów teoretycznych rozkładów losowych pochodzącymi z systemów rzeczywistych a próbkami wyników modelu. Walidacja modelu przez **konfrontację wyników eksperymentu** z wynikami systemu rzeczywistego działającego w porównywalnych warunkach może odbywać się w trybie iteracyjnym, polegającym na dostosowaniu struktury i parametrów modelu po zakończeniu każdej iteracji lub etapu eksperymentu. Nazywane jest to **kalibrowaniem modelu** i zwykle wykonywane jest w **zwykłych** (przeciętnych) **warunkach** funkcjonowania systemu rzeczywistego. Lyneis i Pugh [180] stwierdzają że **podczas kalibrowania modelu** projektant kieruje

się informacjami (statystycznymi) o **odchyłkach między wynikami modelu a systemem rzeczywistym** w celu ustalenia zbioru **hipotetycznych przyczyn** takiego stanu rzeczy. Kolejną czynnością wykonywaną przez projektanta, rozumiejącego zasady działania systemu i posiadającego wiedzę ekspercką jest **strojenie modelu**, czyli takie dopasowanie jego parametrów aby w **kolejnej iteracji eksperymentu zmniejszyć** obserwowane **odchyłki**. Należy zatem zauważyć, że model poddany **walidacji** za pomocą kalibrowania prawidłowe właściwości wykazuje dla tych **przeciętnych warunków** (jest pewnego rodzaju zadaniem estymacji parametrów, które zapewniają niewielkie różnice między wynikami modelu i systemu rzeczywistego). Istnieje zatem **niebezpieczeństwo**, że **nawet całkowicie nieadekwatny** do systemu rzeczywistego zestaw równań i parametrów (struktura) wykaże **prawidłowe wyniki** eksperymentów symulacyjnych dla **konkretnych wartości** parametrów problemu (szczególnie właśnie po kalibrowaniu modelu), działając bezkontekstowo. Dopiero bezstronne sprawdzenie **poprawności odwzorowania** na poziomie struktury modelu i kalibracja dla różnych przypadków wartości zmiennych wejściowych prowadzić może do prawidłowo zwalidowanych modeli. Drugim czynnikiem **ograniczającym zastosowanie kalibracji** modelu wg Oliva'y [225] jest właśnie **przywiązanie twórcy do swojego modelu** objawiające się niechęcią do jego zrewidowania prowadzące do kierowania **uwagi na błędy pomiarów** czy samego procesu walidacji. Jeśli przewidujemy potrzebę skorzystania z modelu w celu zapoznania się z odpowiedzią (wynikami) systemu rzeczywistego także w warunkach odbiegających od przeciętnych to należy uprzednio przeprowadzić kalibrację w takich właśnie warunkach.

Zgodnie z powyższym rozumowaniem, należy **opracować wiele odmian modelu** dostosowanych do poszczególnych scenariuszy warunków w których mają być przydatne lub skonstruować jeden konfigurowalny, ale być może bardziej złożony model ujmujący poszczególne kategorie (klasy) warunków w których może funkcjonować system rzeczywisty (modelowany). Analogicznie jak weryfikacja, także **proces walidacji** postępuje głównie dzięki **wykryciu niezgodności** między wynikami eksperymentów a danymi pochodzącymi z systemu rzeczywistego. Poszukiwania powyższe powierzyć można zespołowi testerów, znających system rzeczywisty, rozumiejących jego działanie i potrafiących przewidzieć jego reakcję. Zwykle jest to jednak czasochłonne, stąd niektóre przypadki (konfiguracje) można poddać **testowaniu zautomatyzowanemu** (np. analizie wrażliwości) a następnie wykonać analizy statystyczne i porównania z systemem rzeczywistym. W praktyce **walidacja symulacyjna** [147] korzysta, o ile pozwala na to wykorzystywane narzędzie programistyczne, z obserwacji **animowanego zachowania się systemu** i **porównywaniu z oczekiwaniami** (przekonaniami, teorią, doświadczeniem) lub z **zachowaniem systemu rzeczywistego** funkcjonującego w podobnych warunkach. **Animacje w systemach zdarzeń dyskretnych** pokazują przemieszczające się obiekty reprezentujące poszczególne żądania obsługi, wykresy przedstawiające długości kolejek itp. **Animacja w symulacji ciągłej** polega na automatycznym sporządzaniu wykresów wartości zmiennych oraz na możliwości

obserwacji zależności kształtów tych wykresów w zależności od manipulowanych (zmienianych za pomocą suwaków ekranowych) wartości parametrów lub wartości początkowych.

Bardziej formalne metody walidacji wykonywane są na podstawie odpowiednio obszernego materiału porównawczego i polegają na statystycznym testowaniu hipotez [250] przy **założonym poziomie ufności** (np. hipoteza o równości średnich z próby wyników systemu rzeczywistego i wyników symulacji). **Wykrycie niezgodności** (o nieakceptowalnym stopniu) między **wynikami otrzymanymi z modelu a systemem rzeczywistym** (a także niezgodności między przebiegiem zmiennych stanu odwzorowania i systemu rzeczywistego) na etapie walidacji oznacza **konieczność ponownego przemyślenia** założeń konstrukcji modelu – w **obszarze struktury** (a więc także równań matematycznych i parametrów) i danych (będących przejawem zachowania: systemu rzeczywistego i modelu). Struktura modelu zależy od przyjętych zasad upraszczania i uogólniania charakteru i liczby elementów modelu oraz relacje między nimi. Można zaproponować następujące techniki radzenia sobie ze **zbytnią złożonością modeli utrudniającą przeprowadzenie walidacji** [208]:

- (1) Traktowanie niektórych zmiennych jako stałych
- (2) Całkowite usunięcie z modelu niektórych zmiennych
- (3) Zastąpienie zależności nieliniowych liniowymi
- (4) Zastosowanie większej liczby i bardziej restrykcyjnych założeń i ograniczeń
- (5) Usunięcie z modelu losowości.

O prawidłowości struktury modelu wyrokuje się na podstawie konfrontacji z obserwowanym zachowaniem systemu rzeczywistego. **Walidacja struktury** modelu nazywana jest także walidacją konceptualną (nacisk na zbadanie odwzorowania teoretycznych założeń). **Celem walidacji konceptualnej** jest wybór właściwej (spośród wielu) teorii wyjaśniającej działanie systemu rzeczywistego oraz wybór konfiguracji elementów (konstrukcja właściwej (lub nawet optymalnej) logiki relacji między formułami matematycznymi (ich reprezentantami)). **Założenia modelu w obszarze danych** dotyczą wiarygodności i reprezentatywności danych pozyskanych z systemu rzeczywistego (danych wejściowych i wyjściowych) służących do konfiguracji (kalibrowania) modelu komputerowego posiadającego poddaną walidacji strukturę. **Zmiany założeń** w obszarze danych mogą polegać na pozyskaniu dodatkowych, uzupełniających zestawów danych służących bardziej adekwatnej kalibracji modelu komputerowego (ale związane jest to z ponoszeniem dodatkowych kosztów).

Walidacja danych nazywana jest także **walidacją operacyjną** (ocena niezgodności wyników modelu i systemu rzeczywistego). Zwykle czynności związane z walidacją operacyjną wykonywane są większą liczbą razy a walidacja konceptualna uruchamiana jest wtedy, gdy walidacja operacyjna nie pozwala na osiągnięcie założonego poziomu adekwatności odwzorowania (pozwala np. na zmianę formuł matematycznych). Rao, Goldsman i Owen [248] twierdzą nawet, że **faza walidacji danych** może być jedyną. Do **metod walidacji operacyjnej** zaliczają: analizę wrażliwości, analizę wariancji,

testy statystyczne takie jak chi-kwadrat, analizę regresji, analizę spektralną, technikę standaryzowanych serii czasowych oraz inne. Niewystarczająca walidacja skutkuje **błędami** dotyczącymi **odwzorowania wariantów** działania systemu rzeczywistego przez **diagramy przepływów** (przedstawiających kolejność czasową poszczególnych procesów), wynikającymi z **nieodpowiedzialnego powielenia w modelu konceptualnym** a następnie w implementacji założeń wpływających z obserwacji przeszłości i przywiązania do rozwiązań stosowanych przez twórców modelu. **Nieodpowiedzialne powielanie rozwiązań** polega na wprowadzaniu do modeli elementów i **powiązań pochodzących** z poprzednich projektów bez dogłębnego rozpatrzenia podobieństw i różnic (zasady działania, procesy, informacje, ograniczenia) między **aktualnie odwzorowywanym systemem rzeczywistym** a systemami nad którymi pracował zespół projektowy.

Według van Horna [311] wyróżnić można następujące techniki walidacji:

1. Korzystanie z **dziedzinowej wiedzy eksperckiej**. Jest ona przydatna podczas wyodrębniania w modelu poszczególnych części (równań matematycznych) i ich odpowiednią konfigurację (np. określenie relacji i kalibrowanie). Doświadczenie **eksperta dziedzinowego** umożliwia prawidłową interpretację informacji towarzyszących systemowi rzeczywistemu i dobór na jej podstawie danych wejściowych, wyjściowych i reprezentujących stan obiektów (elementów) składających się na model
2. Wykonywanie **testów statystycznych** dotyczących zgodności wykorzystywanych rozkładów prawdopodobieństw i **wiarygodnych danych** pochodzących z **obserwacji systemu rzeczywistego**
3. Przeprowadzenie **testu Turinga** na dwu zestawach danych: pierwszy pochodzi z obserwacji systemu rzeczywistego a drugi z **walidowanego modelu symulacyjnego**. Jeżeli eksperci **rozpoznają różnice** między zestawami danych oraz określą pochodzenie każdego z zestawów, to walidacja jest negatywna. Posiadając zestawy danych w tym pochodzący z **negatywnie walidowanego** modelu można wymagać od ekspertów **określenia stopnia niezgodności** wyników z systemem rzeczywistym. Umożliwi to zaplanowanie poprawek które powinny być zastosowane w modelu
4. Statystyczne porównanie **wyników modelu** i systemu rzeczywistego
5. Zastosowanie kolejno **wszystko powyższych technik** (nawet kilkakrotne) dla generowanych za każdym razem zestawów danych. Zalecane po każdej modyfikacji modelu
6. Opracowanie nowego (poprawionego) modelu po **przeprowadzeniu analizy wyników** modelu symulacyjnego i skorzystanie kolejno z powyżej wymienionych technik.

2.7.4 Badania symulacyjne – wskazówki metodologiczne, instrumentarium

Postępowanie w **badaniach symulacyjnych** jest zgodne z ogólnymi zasadami metod badania naukowego. **Badanie w naukach o zarządzaniu** wg Lawrence'a i Pasternaka [162] składa się z następujących kroków: zdefiniowanie problemu, modelowanie za pomocą języka matematyki,

poszukiwanie rozwiązania na podstawie zbudowanego modelu oraz przedstawienie i/lub wdrożenie wyników badań. Po wykonaniu każdego z kroków, nawet końcowego, możliwe jest cofnięcie do kroków poprzednich i ponowne postępowanie w celu zrewidowania modelu, rozwiązań lub wniosków. Badanie naukowe [239] mające na celu opracowanie modelu wyjaśniającego badaną kwestię (funkcjonowanie organizacji) oraz prognozowanie i diagnostykę na jego podstawie składa się z następujących etapów:

1. **Zdefiniowania problemu**, składającego się wg Lawrence'a & Pasternacka [162] z **obserwacji procesów i operacji** zachodzących w firmie (z różnych punktów widzenia)
2. **Sformułowania prostych pytań** na które można uzyskać odpowiedź u członków organizacji (określa się w ten sposób które problemy można rozwiązać już w tej chwili)
3. **Identyfikacji rzeczywistych ograniczeń** o charakterze politycznym (sprzeczne interesy, zasady zdobywania informacji i komunikacja wewnątrz organizacji), **określenie stopnia uświadomienia** potrzeby zmian (sformułowanie problemu jest rozmyte lub dokładnie określone)
4. Zebrania **informacji o ograniczeniach rozwiązań** o charakterze operacyjnym (budżetowe, zasobowe, czasowe, technologiczne), ustalenie zasad i ram komunikacji między zespołami zaangażowanymi w badania.

Między definiowaniem problemu a zebraniem potrzebnych danych Lawrence i Pasternak [162] wyróżniają jeszcze takie **etapy tworzenia modelu matematycznego** jak: zidentyfikowanie zmiennych decyzyjnych (spośród mierzalnych czynników w systemie wyróżniamy zmienne decyzyjne tzn. te którymi można sterować i parametry tzn. te którymi nie można sterować, określające własności obiektów systemu/modelu), **kwantyfikowanie funkcji celu i ograniczeń** (celem jest przystąpienie do optymalizacji – jedno lub wielokryterialnej, uwzględnienie ograniczonych zasobów i czynników zewnętrznych – np. zamówień), budowa szkieletu modelu (formuł matematycznych wiążących zmienne decyzyjne, parametry i relacje między nimi)

5. **Zebrania** potrzebnych danych (Lawrence & Pasternack traktują ten etap jako końcowy krok w tworzeniu modelu matematycznego). W tym etapie następuje zespolenie teoretycznej i doświadczałnej wiedzy z konkretnym przypadkiem obserwowanym w danym momencie. Efektywne zastosowanie modelu jest uwarunkowane zbieraniem danych posiadającym odpowiednie wartości kosztu zbierania, organizacji, porządkowania danych i co ważne, przeprowadzanym w odpowiednio krótkim czasie. Należy także uwzględnić koszt i czas potrzebny do konfigurowania modelu zgodnie z rozwiązywanym problemem. Koszt i czas potrzebny do opracowania rozwiązania (czas przetwarzania) należy już do uwarunkowań opracowania hipotezy
6. **Opracowania hipotezy** dotyczącej systemu wyjaśniającej jego zachowanie, tzn. opracowanie i wykonanie modelu, przynajmniej w takiej postaci która umożliwi wykonywanie eksperymentów
7. **Wykonania eksperymentów** (obliczeniowych, materialnych, socjologicznych lub rozumowych) w celu potwierdzenia lub odrzucenia postawionej hipotezy

8. **Opracowania i interpretacji** wyników eksperymentów
9. **Wyciągnięcie wniosków** i na ich podstawie odrzucenie, przyjęcie lub modyfikacja hipotezy

Winton wyróżnia następujące etapy badań symulacyjnych [325]:

1. **Formułowanie problemu.** Krok ten ma na celu transformację pojęć teorii wyjaśniającej działanie systemu na postać zrozumiałą dla projektanta czyli taką, którą jest w stanie zamodelować. Lawrence i Pasternack [162] wyróżniają **trzy zasadnicze typy problemów w zarządzaniu**: zagadnienia rozszerzonego cyklu projektowego (tzn. łącznie z wdrażaniem) nowych procedur i operacji, ocenę funkcjonowania obecnie wykorzystywanych metod postępowania i technologii (procedur i grup operacji) oraz opracowanie na podstawie wyników oceny działania umożliwiające poprawę funkcjonowania organizacji
2. **Ustalenie celów i ogólnego planu projektu.** Działania te obejmują ocenę w jaki sposób i w jakim stopniu symulacja może pomóc w rozwiązaniu problemów oraz czy istnieją metody konkurencyjne a jeżeli tak, to czy wybranie symulacji jest opłacalne. W tym celu należy opracować **wstępny projekt modelu** oraz **projekt zastosowania** narzędzi symulacyjnych, określić ich koszt, niezbędne zasoby oraz czas potrzebny lub maksymalny (harmonogram) na wykonanie poszczególnych zadań projektowych (tzw. kamienie milowe)
3. **Konceptualizacja modelu.** Na tym etapie dochodzi do wydobycia i włączenia do modelu istotnych (z punktu widzenia projektanta) cech systemu. Zwykle rozpoczyna się od niewielkiego, prostego modelu a następnie rozbudowuje się go iteracyjnie w celu prawidłowego odwzorowania wszystkich istotnych aspektów systemu rzeczywistego. Jak podaje Winton [325] **zasadniczą przyczyną błędów modelowania** na tym etapie jest **zbyt szczegółowe** traktowanie tych aspektów i elementów systemu, które są **zrozumiałe dla modelującego** a pomijanie tych, których teoria, pojęcia i funkcje są nieznane. Projektant modelu powinien zatem dążyć do zdobycia odpowiedniej wiedzy i doświadczenia także w zagadnieniach będących dla niego nowością oraz takich o których wiedza, rozumiana [14] jako wiarygodna informacja posiadająca potencjalną wartość dla organizacji, nie jest jeszcze usystematyzowana i ugruntowana
4. **Akwizycja danych.** Chociaż zwykle przed przystąpieniem do kalibracji modelu i ustalenia wartości parametrów i wartości początkowych zmiennych stanu dostępny jest pewny zasób danych, to w trakcie konstrukcji modelu niezbędne jest zbieranie dodatkowych danych w celu weryfikacji hipotez o zachowaniu wejścia i wyjścia systemu oraz strukturze i wymianie informacji między jego elementami. Zebrane dane służą podstawowym celom modelowania: **opisowi zachowania się systemu** i określeniu charakterystyk (czyli postaci zależności wartości wyjściowej elementu systemu od wartości wejściowej lub czasu). Na podstawie znajomości charakterystyki dynamiki danego elementu systemu (opisanej za pomocą równań różniczkowych, różnicowych, relacji itp.) przedstawionych w postaci matematycznej lub zgodnej z notacją danego systemu komputerowego

(narzędzia symulacyjnego) jesteśmy w stanie prognozować (przewidywać) wartości jego zmiennych wyjściowych w następstwie określonego przemyślanego sterowania wartościami wejściowymi i reakcjach zmiany zachodzące w środowisku. Dzięki akwizycji danych a następnie ich tabelarycznym zestawieniom zdobywa się informacje o natężeniu strumieni wejściowych, charakterze i parametrach rozkładów empirycznych wejścia i wyjścia, ich dynamice oraz wpływie wartości początkowych zmiennych stanu na reakcję systemu w postaci określonych wartości zmiennych wyjściowych. Dane pozbawione wiarygodności lub błędne skutkować będą wprowadzającymi w błąd decydenta wynikami oraz mogą powodować błędy już na etapie projektowania modelu. **Prawidłowa akwizycja danych** musi odbywać się według założonego uprzednio planu, określającego jakie dane należy zbierać (unikając nadmiarowej ilości interesującego rodzaju danych i eliminując dane nieistotne dla rozważanego problemu) i przewidywać odpowiednie działanie w przypadkach nieprzewidzianego zachowania obserwowanego systemu. Można ponadto sformułować następujące **zalecenia dotyczące przeprowadzania akwizycji danych**: grupowanie danych reprezentujące pewne regularności (gdy w pewnym okresie nie obserwuje się zmiany trendu to proces opisany może być przez początek i koniec okresu i charakterystykę trendu), przyjęcie odpowiedniego do charakteru procesu okresu obserwacji (poważnym błędem jest zbyt krótki czas obserwacji), zidentyfikowanie zależności między wartościami danymi (unikanie zbędnego obserwowania wartości zależnych od siebie) i grupowanie danych dotyczących podobnego rodzaju obiektów (np. wielu stanowisk obsługi wykonujących podobne zadania). Określanie trendu szeregów czasowych (tzw. wygładzanie, gdyż poszukuje się funkcji teoretycznych, przedstawiających prawidłowości pojawiające się w obserwowanym szeregu; funkcje takie pozbawione są zakłóceń -reprezentowanych przez losowość) dokonywane może być metodami analitycznymi i mechanicznymi [293]. **Metody analityczne** mają na celu skonstruowanie reprezentacji szeregu czasowego za pomocą funkcji ciągłej (generującej kolejne wartości z teoretycznego szeregu czasowego) o odpowiednim dopasowaniu względem szeregów rzeczywistych. Natomiast **metody mechaniczne** [293] polegają na wyznaczeniu (na bieżąco) pewnych syntetycznych wartości liczbowych (średnich ruchomych, wskaźników rozumianych jako parametry wahań okresowych (numeru okresu i fazy, długości cyklu i fazy) – przy założeniu addytywności lub multiplikatywności obserwowanego trendu, wygładzanie wykładnicze, w którym dawniejsze pomiary mają wykładniczo zanikające znaczenie w prognozowaniu: Browna - beztrendowe, Holta – trend liniowy z przypadkowością, Wintersa – trend liniowy z okresowością i przypadkowością)

5. **Przetłumaczenie modelu konceptualnego** na język komputerowy. Celem tego kroku jest zaimplementowanie modelu w środowisku informatycznym. Rozpoczyna się od doboru narzędzia. Podczas wyboru narzędzia uwzględniamy następujące czynniki: złożoność modelu, typ modelu (ciągły, dyskretny, hybrydowy), istniejącego środowiska informatycznego – dostępności interfejsów do istniejących baz danych i systemu informatycznego przedsiębiorstwa, koszt nabycia, szkolenia projektantów i przyszłych użytkowników, konserwacji narzędzia i inne. Następnie dokonuje się

właściwej translacji modelu konceptualnego do modelu symulacyjnego, dokonuje się doboru elementów spośród zestawu dostępnych modułów, nadaje właściwe i czytelne dla projektanta identyfikatory (nazwy), dostosowuje ich parametry, wprowadza relacje między elementami (modułami), uzupełnia model o interfejs użytkownika i integruje z posiadanym systemem informatycznym. Podczas tego procesu wyszukuje się oraz usuwa błędy składniowe i logiczne a także określa się dopuszczalne zakresy danych wejściowych

6. **Weryfikacja modelu**, czyli sprawdzenie zgodności wyników eksperymentów z modelem symulacyjnym i przewidywań na podstawie modelu konceptualnego

7. **Walidacja modelu**, czyli sprawdzenie wystarczającej zgodności wyników uzyskiwanych z modelu symulacyjnego z wynikami i zachowaniem się systemu rzeczywistego. Jeżeli walidacja nie przebiegnie pomyślnie, należy wykonać ponownie kroki 3-7, czyli opracować nową lub poprawioną konceptualizację modelu. Lawrence i Pasternack [162] twierdzą, że **ocena walidacyjna modelu** związana jest z odpowiedzią na następujące pytania: (1) **czy wyniki wydają się** zgodne z oczekiwaniami, intuicją, (2) **uwzględnienie wykonalności** proponowanych rozwiązań – czy konieczne będzie dokonanie zasadniczych zmian w organizacji aby wdrożyć rozwiązania, (3) **czy zaproponowane rozwiązania** mieszczą się w strategii przedsiębiorstwa i czy nie spowodują zaburzeń jej funkcjonowania

8. **Projektowanie (planowanie) eksperymentów symulacyjnych**. Celem jest przystosowanie walidowanego modelu do rozwiązywania konkretnego problemu decyzyjnego. Realizacja tego zadania polega na uwzględnieniu w modelu (poprzez edycję jego parametrów) odpowiednich do danego problemu ograniczeń (tzn. uwzględniających wykonalność wariantu decyzji, koszty itp.), sformułowaniu konkretnego kryterium oceny, zaprojektowaniu zestawu eksperymentów. Glover i inni twierdzą [96], że „prowadzenie serii symulacji w najbardziej efektywny sposób, zamiast przygotowanych na ślepo scenariuszy (z nadzieją że jeden z nich będzie najwłaściwszy) było od dawna stawianym zadaniem. Integracja symulacji i optymalizacji umożliwi praktyczne rozwiązanie tego problemu”. **Plany eksperymentu** (określenie zestawu danych wejściowych i określenie liczby eksperymentów) **można klasyfikować na podstawie kryterium czasu**, w którym przypisywane są wartości zmiennych wejściowych dla kolejnej iteracji eksperymentu: można je ustalić przed serią eksperymentów albo ustalać na bieżąco a więc na podstawie wyników poprzednich eksperymentów lub postępów w rozwiązywaniu problemów. Innym kryterium **klasyfikacji planów eksperymentu** jest warunek stopu: wykonanie założonej liczby eksperymentów lub osiągnięcie założonego postępu w rozwiązywaniu problemów (np. zbieżności rozwiązań w kolejnych iteracjach)

9. **Wykonanie eksperymentów zaprogramowanych wg planu**, wstępne porównanie badanych wariantów i ocena wyników. Podczas eksperymentów kontrolowane są ich wyniki, uruchamiane odpowiednie procedury w przypadku błędów wykonania programu, przekroczenia założonych zakresów a także postępowanie zgodnie z założonym planem eksperymentu (sprawdzenie warunku

stopu, krokowa zmiana parametrów w analizie wrażliwości, wyznaczenie wartości kroku na podstawie dotychczasowych wyników – realizacja dynamicznego planu eksperymentu, klasyfikacja poszczególnych wyników – odrzucanie i zapominanie lub przyjęcie i wykorzystanie w dalszych iteracjach). W omawianym kroku sporządzany jest **ranking wyników** (alternatyw decyzyjnych) i **przeprowadzane studium wykonalności wariantów** po uwzględnieniu założonych ograniczeń implementacyjnych

10. **Ocena wystarczalności zebranych danych.** Jeśli istnieją przesłanki że zebranie większej liczby danych lub zebranie ich w inny sposób **zwiększy pewność podejmowanych decyzji**, to następuje powrót do kroku 9 (gdy zawodzi warunek stopu, nie osiągnięto zakładanej zbieżności, utknięcie w lokalnym minimum) lub 8 (obserwowano przekraczanie dopuszczalnego zakresu wyników, nie uzyskano znaczącej poprawy ogólnej wartości kryterium, gdy poszczególne **ocena poszczególnych wariantów decyzyjnych** różni się nieznacznie)

11. **Dokumentacja systemu, modelu i wyników oraz opracowanie raportu z eksperymentów.** **Dokumentowanie rozwoju środowiska symulacyjnego** ma na celu przechowanie historii zmian dotyczących ewolucji systemu rzeczywistego, rozwoju modelu konceptualnego i modyfikacji programu komputerowego (modelu symulacyjnego) i ich wzajemnych relacji. Na **ewolucję systemu rzeczywistego** projektant nie ma zasadniczo wpływu, natomiast na jego zmiany mogą wpływać zarówno uruchomione przez decydenta eksperymenty symulacyjnego w trakcie podejmowania decyzji zmieniających warunki i zachowanie systemu jak też modyfikacje struktury, sposobów komunikowania się elementów systemu (aspekty informacyjno-społeczne) i procesów finansowo-materialnych inicjowane przez decydenta po zaznajomieniu się z modelem konceptualnym i uświadomieniu przez niego wąskich gardeł, zbędnych zasobów itp. Rozwój modelu konceptualnego postępuje zazwyczaj w pewnym opóźnieniu względem systemu rzeczywistego (z wyjątkiem etapu formułowania nowych strategii rozwoju systemu rzeczywistego), natomiast model symulacyjny (po jego udanym wdrożeniu w przedsiębiorstwie) wykorzystywany może być na bieżąco w podejmowaniu decyzji o charakterze operacyjno-taktycznym, także w wymiarze wytwarzania i produkcji

12. **Wdrożenie modelu.** Model wykorzystywany jest w procesie podejmowania decyzji jako narzędzie prognozowania, projektowania i eksperymentalnej oceny wariantów decyzyjnych, weryfikowania wykonalności i skuteczności propozycji działań oraz wartościowania poszczególnych strategii. Narzędzie symulacyjne wspiera system zarządzania produkcją poprzez sterowanie procesami wytwarzania wielu różnych wyrobów z materiałów wielu dostawców dla wielu odbiorców. Efektem korzystania z modelu powinno być dokonanie optymalnego wyboru spośród wielu opcji (zazwyczaj ze zróżnicowanym ryzykiem) dotyczących wyrobów i partnerów handlowych a na poziomie procesu produkcyjnego wyboru szczegółów technologii wytwarzania (łańcuchy produkcyjne, konfiguracja stanowisk roboczych, sieci produkcyjnych, identyfikacja wąskich gardeł) [16].

2.7.5 Wskazówki metodologiczne - eksperymenty i raport z badań symulacyjnych

Winton [325] definiuje **eksperyment** jako proces którego wyników nie można przewidzieć z całkowitą pewnością. Jeśli możemy natomiast przewidzieć z jakiego zbioru pochodzić mogą wyniki eksperymentów, to taki zbiór wszystkich możliwych wyników nazywamy przestrzenią próbek eksperymentu (*sample space*). **Zmienna losowa** natomiast reprezentuje wyniki poszczególnych eksperymentów i przyjmuje wartości należące do przestrzeni stanów. Jeśli jest on zbiorem skończonym to dla każdego z możliwych wyników można określić wartość prawdopodobieństwa, suma tych prawdopodobieństw równa jest 1, a funkcja przyporządkowująca wartościom wyników ich prawdopodobieństwa nazywana jest rozkładem prawdopodobieństwa.

Lawrence & Pasternack [162] podkreślają, że rolą zespołu konstruującego model jest **przedstawienie decydentowi istoty problemu**, przyjętych założeń i przybliżeń i ścieżki wnioskowania prowadzącej do uzyskanych i zatwierdzonych propozycji działań. Proponują także następującą strukturę raportów:

- (1) **Wstęp – opis problemu.** Przedstawiony on jest za pomocą pojęć i wyraża poglądy właściwe dla kadry zarządzającej. Zawiera także propozycje postępowania w celu uzyskania rozwiązań
- (2) **Przyjęte założenia i przybliżenia.** Powinny być przedstawione w sposób zrozumiały dla decydenta aby mógł przeprowadzić ich wartościowanie i ocenić czy się z nimi zgadza
- (3) **Opis wykorzystanej techniki modelowania** i zastosowanego programu komputerowego. Chodzi tu o uwiarygodnienie sposobu otrzymania rozwiązania
- (4) **Przedstawienie i analiza wyników.** Zawierać powinno zestawienia, wykresy, perspektywy zastosowania wraz z przewidywanymi trudnościami
- (5) **Analiza typu „co się stanie jeśli”.** Podstawowym celem jest prezentacja pewnej liczby wariantów rozwiązań wraz z dyskusją wpływu założeń i przyjętych ograniczeń
- (6) **Ogólne zalecenia.** Omówienie wyników oceny poszczególnych wariantów, przedstawienie charakterystyk wybranych wariantów za pomocą wykresów i zestawień
- (7) **Dodatki.** Obejmują **techniczne szczegóły rozwiązań**, przedstawiają (dla weryfikacji) sposób samodzielnego dojścia do ustalonych rozwiązań.

Można wyróżnić **trzy techniki dotyczące analiz symulacyjnych** prowadzonych w **warunkach niepewności**: wykorzystujące różne rozkłady zmiennych losowych, jednakowe rozkłady zmiennych losowych, jednakowe wartości zmiennych losowych.

Pierwszy sposób polega na wykonywaniu dla poszczególnych strategii (tzn. zestawu ustalonych wartości zmiennych decyzyjnych) tej samej liczby eksperymentów symulacyjnych, uwzględniających czynniki losowe. Obserwacja wyników (np. wartości funkcji kryterium oceny) może prowadzić do opracowania (wyznaczenia) w kolejnych iteracjach nowych strategii. Należy jednak zauważyć, że warunki prowadzonych eksperymentów za każdym razem są nieco inne, gdyż zmienne losowe

przyjmują inne wartości; wyniki eksperymentów dla tych samych zmiennych decyzyjnych mogą różnić się między sobą – nawet znacznie jeśli założono duży znaczący wpływ wartości zmiennych, oddziaływania zmiennych losowych kumulują się, spowodowały zmianę parametrów modelu itp. Co więcej, można zaobserwować sytuację w której **zwykle lepsza strategia** przegrywa z gorszą z powodu zbiegu okoliczności (wylosowanych wartości zmiennych losowych). Powyższej opisany problem można nieco ograniczyć wykonując odpowiednio dużą liczbę eksperymentów symulacyjnych dla każdej z rozważanych strategii. Zaletą tej metody jest możliwość modyfikowania strategii w kolejnych iteracjach (kolejność wykonywania eksperymentów ma znaczenie).

Drugi sposób polega na wykonywaniu dla poszczególnych strategii (tzn. zestawu ustalonych wartości zmiennych decyzyjnych) tej samej liczby eksperymentów symulacyjnych, uwzględniających czynnik losowe, z zastrzeżeniem, że **rozkłady zmiennych losowych są takie same** dla wszystkich strategii. Oznacza to, że eksperymenty wykonywane są w tych samych warunkach. Eksperymenty mogą być wielokrotnie powtarzane (w każdej iteracji wygenerowane są wartości zmiennych losowych, z zestawu wartości losowych wspólnego dla wszystkich strategii). Obserwacja wyników (np. wartości funkcji kryterium oceny) prowadzi do wyboru najlepszej strategii. Oceny są wydawane po statystycznym przetworzeniu wyników poszczególnych strategii (dla każdej strategii dostępnych jest wiele wyników). Statystyki mogą (ale nie muszą) uwzględniać uporządkowanie wyników poszczególnych strategii w poszczególnych iteracjach. Oznacza to, że można wykorzystać sumę miejsc w rankingu danej strategii (wynik dyskretny) albo średnią ocenę dla wszystkich iteracji (wynik ciągły). Wadą metody jest konieczność przygotowania i rozważania ograniczonej liczby strategii. Spowodowane jest to wykonywaniem eksperymentów symulacyjnych dla wszystkich strategii praktycznie jednocześnie, gdyż kolejność wykonywania eksperymentów w danej iteracji (zestawie zmiennych losowych) nie jest istotna.

Zastosowanie *trzeciego sposobu* ograniczone jest do **niezbyt wielkiej liczby zmiennych losowych**. Tworzymy zestawienie zawierające pewną liczbę wygenerowanych wartości poszczególnych zmiennych losowych. Zestawienie to może służyć jako baza dla obliczenia wartości wynikowej dowolnej liczby strategii. W podejściu powyższym wartości zmiennych losowych (pochodzące z puli reprezentowanej przez opisane zestawienie) podstawiane są bezpośrednio do formuł wyliczających wartość kryterium. Nie ma potrzeby generowania zmiennych losowych podczas wyliczania wartości kryterium dla danej strategii. Wadą opisanego powyżej podejścia jest nieprzydatność w warunkach większej liczby zmiennych losowych. Pojawiają się wtedy trudności związane z koniecznością opracowania wszystkich kombinacji zmiennych losowych.

2.8 Symulacja ciągła i dyskretna

2.8.1 Pojęcia symulacji ciągłej

Podstawowymi pojęciami symulacji ciągłej są **zasoby, strumienie i atrybuty**. Zachowanie poszczególnych elementów systemu opisywanego za pomocą pojęć symulacji ciągłej uzależnione jest od rodzaju relacji zachodzących między nimi, czyli przede wszystkim sprzężeń zwrotnych dodatnich i ujemnych. Dodatkowo na wartości obserwowanych zmiennych wpływ mają parametry opisujące wzmocnienie lub tłumienie systemu. Systemy które nie przejawiają **własności dynamicznych** (posiadają w określonym zakresie charakterystykę liniową) mogą wykazywać własności wzmacniania lub tłumienia. Złożone systemy wykazują obecność wielu **sprzężeń zwrotnych** wzajemnie ze sobą powiązanych. **Sprzężenia zwrotne dodatnie** objawiają się przebiegami wzrostowymi (w wartościach bezwzględnych danej zmiennej, np. zasobu), natomiast **sprzężenia zwrotne ujemne** przebiegami dążącymi do stanu ustalonego (stabilności). Przejawem działania **elementów dynamicznych** (opisywanych za pomocą równań różniczkowych) jest także zjawisko opóźnienia, polegające na podobieństwie przebiegów zmiennych wyjściowych i wejściowych (wymuszeń) przy czym trajektoria zmiennej wyjściowych jest opóźniona o pewien czas względem zmiennej wejściowej. Za przebieg zmiennych stanu i zmiennych wyjściowych (przy danym wymuszeniu, czyli przebiegu zmiennych wejściowych) odpowiada struktura układu, a więc także przewaga wpływu sprzężenia zwrotnego danego typu (dodatniego lub ujemnego). Na podstawie obserwowanych zachowań systemu można wnioskować o charakterze sprzężeń.

Wyróżniającą cechą modelowania ciągłego jest możliwość opracowania i prezentacji szeregu modeli (szablonów) odwzorowujących strukturę i działanie systemów mikroekonomicznych. Wśród nich Odum [222] i Ortega [227] wymieniają model sprzedaży przy cenie odwrotnie proporcjonalnej do podaży, model ekonomicznego wykorzystania surowców kopalnych, model depozytów i odsetek bankowych, model produkcji, konsumpcji i obiegu pieniądza, model ekonomicznego wykorzystania zasobów odnawialnych, model ekonomicznej roli zróżnicowania biologicznego obszarów chronionych.

2.8.2 Pojęcia symulacji dyskretniej

Symulacje zdarzeń dyskretnych (zmiany wartości zmiennych i obliczenia wykonywane są tylko w dyskretnych chwilach) znajdują zastosowanie w rozwiązywaniu problemów uwzględniających czynniki losowe i dotyczących redukcji średniego czasu obsługi zleceń, oczekiwania zlecenia w kolejce, doboru liczby stanowisk obsługi itp. Wyróżnić można następujące główne podejścia (są także inne, prezentowane przez Fishmana [87]) wykorzystywane w modelowaniu

zdarzeń dyskretnych: **stałego odstępu** (*fixed time*) **czasu** i **symulacji następnego zdarzenia** (*next-event simulation*). Cechami charakterystycznymi symulacji wykorzystującej **stałe odstępy czasowe** są **prostsze obliczenia** (z tego wynika także prostsze sterowanie) i możliwość wystąpienia braku jakichkolwiek zmian w modelu w poszczególnych okresach. W **metodzie stałego odstępu czasu** wszystkie obliczenia dokonywane są wyłącznie w ściśle, z góry określonych chwilach (poza nimi nic się nie dzieje). Symulacja **następnego zdarzenia** może być oparta na **metodzie interakcji procesów** polegającej na śledzeniu (ustawicznej obserwacji) i ewentualnej aktualizacji wszystkich procesów przebiegających w modelu. W metodzie interakcji procesów **pojedyncze zdarzenie** (np. zwolnienie stanowiska obsługi na skutek ukończenia obsługi danego obiektu) powoduje **aktualizację** (tam gdzie jest to możliwe) stanu wszystkich procesów i obiektów (np. przemieszczenie następnego obiektu z kolejki do stanowiska obsługi, zmniejszenie długości kolejki, wyjście obiektu z systemu itd.). Podejście następnego zdarzenia **podobnie jak** podejście stałego odstępu cechuje się możliwością wystąpienia zmian wartości zmiennych i stanu obiektów tylko w dyskretnych chwilach czasu, jednak czas wystąpienia tych zmian nie jest z góry określony, choć możliwy do przewidzenia jeśli w modelu brak jest czynnika losowości.

Cechą charakterystyczną systemów zdarzeń dyskretnych jest występowanie **stanowisk obsługi** których celem jest obsługiwanie (trwające przez pewien czas) zleceń (zgłoszeń) oraz **obowiązywanie regulaminu** (zasad organizacji) oczekiwania zleceń na obsługę i postępowania względem nich przez stanowiska obsługi. System kolejkowy opisywany jest w języku modelowania dyskretnego, w którym wyróżnia się [293] następujące pojęcia: system (wyodrębniając go z otoczenia w celu objęcia badaniem), rodzaj obsługi (definicja i interpretacja czynności wykonywanych na stanowisku obsługi, relacja między stanowiskiem obsługi a klientem-zgłoszeniem poprzez kolejki tworzone przez oczekujących klientów), stanowisko obsługi (definicja i charakterystyka czasowa stanowiska obsługi w zależności od rodzaju obsługiwanego zgłoszenia), regulamin obsługi, liczba kanałów obsługi, opis możliwości wstrzymania i wznowienia działania itd.), zgłoszenie (definicja obiektu-zlecenia które ma być obsłużone na stanowisku obsługi, liczba zleceń (obiektów) i natężenie ich strumienia w funkcji czasu i jakości obsługi).

2.9 Symulacja – podstawowe zalety i trudności

Według Bäckä i innych [16] symulacja znajduje zastosowanie w rozwiązywaniu problemów następujących obszarów zarządzania: planowanie i analiza procesów produkcji, wybór wariantów (polityk) postępowania, planowanie systemów logistycznych, planowanie wymaganej wydajności procesów lub usług, analiza systemów ekonomicznych.

Podstawowe przyczyny wykorzystywania modelowania symulacyjnego w badaniach polegających na eksperymentach wg Wintona [325] i Jensena [133] są następujące:

1. Eksperyment symulacyjny umożliwia **badanie zachowania się systemu rzeczywistego** odwzorowanego poprzez model w sytuacjach gdy eksperymenty bezpośrednio na systemie rzeczywistym są **niemożliwe** (np. człowieka jako organizmu, złożonych organizacji ludzkich), **kosztowne** (np. określenie strategii przedsiębiorstwa, inwestycje finansowe i produkcyjne) lub **długotrwałe** (np. cyklu życia przedsiębiorstwa, kampanii reklamowej)
2. Podczas konstruowania modelu symulacyjnego i przeprowadzania eksperymentów symulacyjnych twórca dokonuje **dogłębnej analizy procesów** przebiegających w systemie rzeczywistym, określa **strukturę systemu** i wartościuje jego elementy. Dzięki temu twórca rozwija w sobie **rozumienie działania i istoty systemu**
3. Dzięki **modelowi symulacyjnemu** można przewidzieć **następstwa planowanych zmian** w systemie rzeczywistym (jego struktury, parametrów poszczególnych elementów i relacji, strategii, komunikacji między podsystemami) i zmian zachodzących w jego środowisku. Osoba korzystająca z modelu symulacyjnego może dzięki niemu uzyskać odpowiedzi na pytanie „co się stanie jeśli ...”. Umożliwia to **sporządzenie zestawienia tabelarycznego** przedstawiającego **skutki działania** (np. liczbę wytworzonych produktów, obciążenie stanowisk roboczych i pracowników, długość kolejek, czas realizacji zamówienia) **poszczególnych reguł decyzyjnych**, np. dobierających wykorzystywane zasoby, czas pracy i liczbę pracowników, typy stanowisk roboczych czy parametrów maszyn i urządzeń.
4. **Wielokrotne eksperymenty symulacyjne** umożliwiają empiryczne zweryfikowanie metodologii analitycznych i metod podejmowania decyzji. Zweryfikowanie negatywne może pociągnąć za sobą odrzucenie a pozytywne – zwiększenie zaufania lub określenie słabych punktów i koniecznych modyfikacji.
5. Model symulacyjny szczególne usługi oddaje podczas eksperymentów w **warunkach** w których może mieć miejsce **wadliwe działanie systemu** oraz sytuacji szczególnie niesprzyjających czy niebezpiecznych. Na podstawie tych informacji można stwierdzić **co należy zmodyfikować** w systemie rzeczywistym
6. Obserwacja przebiegu wartości zmiennych reprezentujących poszczególne elementy modelu i wzajemne ich relacje umożliwia **uświadomienie** sobie przez użytkownika modelu **powiązań**

przyczynowo-skutkowych (np. między ceną oferowanej usługi a wynikami finansowymi) i znaczenia nawet niewielkich różnic w wartości parametrów podsystemów i zmian warunków początkowych w poszczególnych próbach (eksperymentach).

Z powyższych przesłanek wynika, że symulacja zatem posiada **wartości dydaktyczne**, gdyż dzięki wielokrotnym eksperymentom symulacyjnym, **w przeciwieństwie do metod analitycznych**, jest szansa na przeprowadzenie **badania dotyczących rzadko spotykanych warunków**. W ten sposób istnieje możliwość istotnego zmniejszenia niebezpieczeństw w realizacji celów systemu rzeczywistego poprzez **przygotowanie procedur wyjścia z nieprzewidzianych i niekorzystnych sytuacji**.

Moore i Weatherford [2001] porównując modele analityczne (optymalizacyjne) i symulacyjne wyróżnili ponadto następujące przyczyny stosowania modeli symulacyjnych:

1. Jeśli wśród czynników występujących w systemie rzeczywistym znajduje się czynnik charakteryzujący się znaczną złożonością lub trudnością w modelowaniu (np. czynnik losowy o nieznanym rozkładzie) to odpowiednie **modele analityczne stają się bardzo złożone** lub nawet **niemożliwe do skonstruowania i rozwiązania**
2. Za pomocą **modeli analitycznych** otrzymujemy zwykle **wartości średnie** lub charakteryzujące stan ustalony, gdy tymczasem bardzo istotne nawet dla przetrwania firmy są wartości zmiennych i parametrów **w ciągu całego okresu funkcjonowania** (eksperymentowania) oraz wszystkie stany natury opisane za pomocą wartości prawdopodobieństw
3. Dostępność coraz łatwiejszych w obsłudze (skrócenie czasu szkoleń dla twórców i użytkowników modele) **narzędzi modelowania symulacyjnego**, posiadających coraz większe możliwości i wsparcie ze strony producentów oraz niezależnych firm software'owych.

Korzyści i zalety płynące z zastosowania symulacji dotyczą **zastąpienia badań systemu rzeczywistego badaniami modelu** (prognozowanie skutków poszczególnych wariantów przedsięwzięcia, radykalne ograniczenie kosztów, identyfikacja zagrożeń i reakcji systemu w sytuacjach niecodziennych), **przyspieszenia procesów badawczych** (czas symulacji może być wielokrotnie krótszy niż czas rzeczywisty odwzorowanych procesów rzeczywistych; ograniczeniem kompresji czasowej jest pojemność pamięci i prędkość obliczeń komputera), **możliwości analizy wpływu poszczególnych zjawisk na zachowanie modelu**. Lawrence i Pasternack [162] wyróżniają następujące korzyści płynące z zastosowania analizy na podstawie symulacji:

1. Symulacja może stanowić **jedyny sposób rozważania, opisu** (prezentacji) i dążenia do **rozwiązania** problemu gdy inne metody nauk o zarządzaniu zawodzą (jako przysłowiowa ostatnia deska ratunku)
2. **Możliwość poprawy funkcjonowania** systemu, stopnia wypełniania oczekiwań i wymagań (podwyższenie jakości) dzięki symulacyjnemu przetestowaniu różnych rozwiązań i modyfikacji systemu bez zakłócania jego bieżącego funkcjonowania oraz możliwości kompresji czasu (jako metoda abstrakcji, ewaluacji zmian strukturalnych i prognozowania na dłuższy okres)

3. Narzędzie przeprowadzania laboratoryjnych eksperymentów typu „co się stanie jeśli” **bez angażowania zasobów** związanych z systemem rzeczywistym (aspekt badawczy, analiza wrażliwości).

Podsumowując, można wyszczególnić następujące **użytkowe zalety symulacji** [125]:

1. **Odkrycie relacji wiążących przyczyny i skutki** – przewidywanie efektów poszczególnych działań (wyników końcowych jak i przebiegu wartości zmieniających się po wystąpieniu jakiegoś zdarzenia (symulacja zdarzeń dyskretnych) lub w stałych odstępach czasu (symulacja ciągła)
2. **Identyfikacja przyczyn** prowadzących do aktualnego (lub badanego) stanu bądź zdarzeń
3. Możliwość ukierunkowania badań na obszary **w których mogą powstać problemy** jeszcze przed implementacją
4. **Przewidywanie i badanie następstw** proponowanych lub przeprowadzonych modyfikacji, ulepszeń, poprawionych i nowych projektów. Określenie niedomagań systemu a następnie testowanie i ocena proponowanych sposobów usunięcia źródeł nieefektywności
5. **Konieczność wykonania walidacji modelu** poprzez porównanie wyników eksperymentów symulacyjnych i obserwacji systemu rzeczywistego a następnie uzupełnienie lub modyfikację modelu o zmienne poprzednio nieuwzględnione lub zebranie pełnych czy dokładniejszych danych o systemie rzeczywistym
6. **Walory dydaktyczne i naukowe** dzięki inspirowaniu nowych przemyśleń, dróg rozwiązań problemów i dokładniejszego zrozumienia działania systemów. Dzięki symulacji możliwe jest przedstawianie na różnych forach wariantów projektów rozwoju systemów i grupowa weryfikacja ich wykonalności oraz zwiększenie obiektywizmu i zapewnienie całościowego podejścia podczas wartościowania projektów.

Zastosowanie symulacji napotyka jednak na pewne **utrudnienia i niedogodności**. Dotyczą one m.in. trudności projektowo-metodologicznych takich jak **brak ściśle określonych i uznanych reguł** prowadzących do opracowania modelu symulacyjnego, a zatem przedstawione przez projektantów – znawców modelowania modele i ich aplikacje (zwłaszcza, gdy odwzorowywany system jest złożony a nie przedstawiono modelu uznanego za obowiązujący) **mogą znacznie różnić się** między sobą strukturalnie, technicznie, narzędziowo, użytkowo, aksjologicznie i metodycznie. Przyczyną tego stanu są przede wszystkim trudności weryfikowania i walidacji modelu i jego poszczególnych modułów z systemem rzeczywistym, gdyż często nie ma możliwości wykonywania między nimi **testów porównawczych**. Dlatego twórca modeli musi opierać się na swoim doświadczeniu i korzystać z wnioskowania przez analogie, na podstawie dostępnych modeli i doniesień o wynikach ich stosowania w praktyce. Lawrence i Pasternack [162] **wyróżniają następujące trudności** związane z zastosowaniami symulacji:

1. Do opracowania wiarygodnych modeli symulacyjnych niezbędni są wybitni specjaliści a rozwój modeli (szczególnie od podstaw) i ich wdrożenie **wymaga czasochłonnych prac** i jest zasadniczo

kosztowne (aspekt ekonomiczno-czasowy)

2. Modele symulacyjne oferują **uzyskanie przybliżonych** rozwiązań gdyż korzystają z przybliżonych wartości parametrów modelu (aspekt stochastyczno-metodologiczny); Moore i Weatherford [207] podkreślają wrażliwość wyników symulacji na **założenia dotyczące parametrów wejściowych** (rozkłady losowe i ich parametry) i konieczność podejmowania decyzji dotyczących liczby przebiegów symulacyjnych. Należy także zwrócić uwagę na **popelniany często metodologiczny błąd** polegający na mechanicznym zastępowaniu nieznanymi parametrów (na które wpływają czynniki losowe) przez wartości średnie

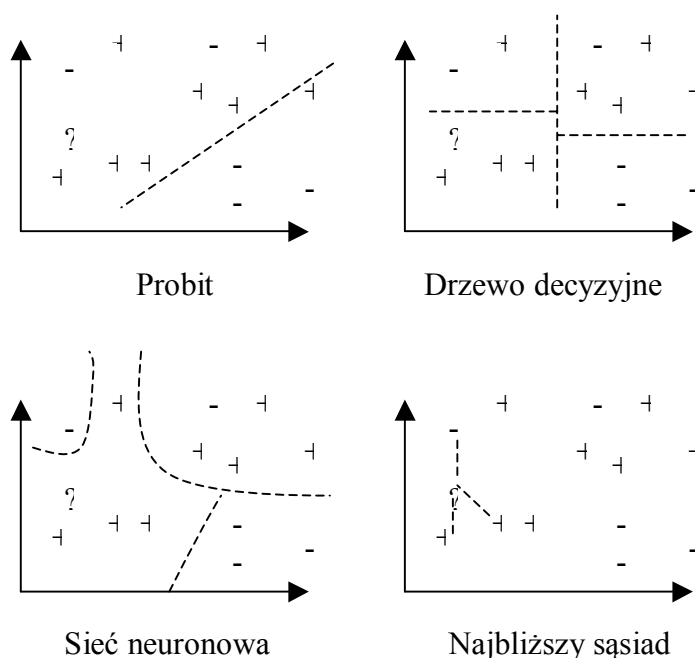
3. Praktycznie **nie jest możliwe zapewnienie bezwzględnej optymalności** rozwiązań uzyskanych za pomocą eksperymentów symulacyjnych (aspekt metodologiczny – wniosek wspólny dla technik numerycznych). Można mówić jedynie o wsparciu symulacji w procesie wyboru najlepszych rozwiązań z ograniczonej, dyskretnej puli wariantów. **To ograniczenie, jak zostanie wykazane w opisie hybrydowego SWD Ekanwin, zostało częściowo przewyżnione.**

Wynika z tego, że stworzenie modelu i przystosowanie go do wsparcia rozwiązania konkretnego problemu a następnie wykorzystania trudnych w interpretacji wyników **wymaga dogłębnych i długotrwałych analiz** wykonywanych przez odpowiednio przeszkolonych specjalistów i zapewne będzie kosztowne. Powyższe trudności można próbować przewyżnić w pewnym stopniu poprzez opracowanie i korzystanie odpowiednio z: wzorców – **szablonów programowych** reprezentujących pewne klasy problemów modelowania (tworzenie repozytoriów zawierających uprzednio tworzone modele i **gotowych konfigurowalnych modułów** – bloków dostępnych w oprogramowaniu komercyjnym) znacznie skracające czas projektowania modeli, biblioteki i narzędzia służące statystycznym interpretacjom znacznej liczby wyników. Nie można także określić a priori czy symulacja będzie na pewno lepszą metodą przygotowania do podejmowania decyzji niż np. metody analityczne. Chociaż, jak już wspomniano, eksperyment symulacyjny korzysta z możliwości **kompresji czasu** to jednak może się okazać, że procedury podejmowania decyzji wymagają znacznej dokładności i odpowiedniej liczby danych. Dlatego też należy **dobrać odpowiednio liczbę iteracji** lub symulowanego okresu badań – zbyt mały spowoduje niedokładności w odwzorowaniu funkcjonowania systemu rzeczywistego a zbyt długi lub drobiazgowy (choć korzystny ze względu na wartość analityczną) wymagać będzie dużych zasobów pamięci, szybkiego procesora lub uniemożliwi uzyskanie wyników w rozsądnym, założonym czasie. Zasadniczo model symulacyjny wykorzystywany jest do prowadzenia **wielokrotnych eksperymentów**, dlatego niezbędne jest zastosowanie metod statystycznej obróbki wyników i dlatego należy zastosować się do podobnych procedur i zaleceń jak we wnioskowaniu statystycznym.

3 Zarządzanie wiedzą, sztuczna inteligencja, systemy wspomagania decyzji

3.1 Metodologia reprezentacji wiedzy

Reprezentacja wiedzy (*knowledge representation*) jest owocem **modelowania** czyli „ustrukturalizowanych, nieprzerwanych, ukierunkowanych na cele strategiczne procesów myślowych dotyczących zrozumienia rzeczywistości” [14], wykonywanego z pobudek utylitarnych. Posługuje się w tym celu faktami, atrybutami i obiektami należących do określonej dziedziny **w celu wyciągnięcia wniosków** za pomocą programu komputerowego (wnioskowania”). Dziedzinowość oznacza ograniczenie się do fragmentu rzeczywistości i nauki wyodrębnionej na podstawie zastosowania oraz metod przetwarzania. Opracowanie reprezentacji wiedzy dziedzinowej wymaga sporządzenia opisu i zakodowania [Rys. 3-1] pojęć występujących w teorii wyjaśniającej funkcjonowanie obiektów należących do omawianej dziedziny (ujęcie statyczne i dynamiczne). **Stacyjny aspekt wiedzy** dotyczy **zdefiniowania pojęć i terminologii** składających się na opis dziedziny za pomocą obiektów i relacji dziedziczenia (określanych przez zestaw atrybutów oraz ich relacji – ograniczeń wyrażanych za pomocą matematycznych zależności z dokładnością do parametrów). **Dynamiczny aspekt wiedzy** dotyczy **parametryzacji zależności** między atrybutami i obiektami, przyjęciem wartości przez atrybuty obiektów oraz ich zmian w czasie. **Dynamika wiedzy** pozwala na nadanie wartości symbolicznym formułom (pojęć i terminologii) skutkując uformowaniem się wiedzy faktograficznej, grupującej rejestrowanych zdarzenia w postaci faktów i stanów obiektów. Wiedza przyczynowa ujmuje ilościowo i jakościowo (np. za pomocą reguł czy drzew decyzyjnych) związki występujące między powstającymi i dezaktualizującymi się faktami i obiektami, trajektoriami stanów obiektów.



Rys. 3-1 Metody odkrywania wiedzy (klasyfikacyjne). Probit oznacza liniową regresję. Źródło: [90]

3.2 Znaczenie psychologii poznania dla metodologicznych podstaw reprezentacji wiedzy i technik sztucznej inteligencji

Psychologia decyzji [18] określa dwa typy procesów podejmowania decyzji: **jednolitego** (pojedyncza osoba lub zespół osób, który można rozpatrywać jako pojedynczy podmiot, powiązany szczególnymi relacjami i posiadające zgodne poglądy) decydenta oraz **grupy decydentów** (grupa osób, posiadających własne, często odrębne zdanie, w której do podjęcia decyzji konieczny jest kompromis). Wyróżnia się także **dwa rodzaje podmiotów** podejmujących decyzje (decydentów): **posiadających doświadczenie** w danej dziedzinie (w tym przypadku zadaniem systemu wspomagania decyzji jest dostarczenie oprogramowania służącego do optymalizacji wyników podejmowanych decyzji, zwiększeniu wydajności (szybkość podejmowania decyzji) i powiększeniem zasobu wiedzy (doświadczenia)) oraz **bez doświadczenia** (w tym przypadku decydent potrzebuje wsparcia ze strony konsultantów oraz – w przypadku złożonych problemów – modeli i oprogramowania wspierającego wielokryterialne podejmowanie decyzji).

Psychologiczne aspekty teorii decyzji wyszczególnione przez Einhorna i Hogartha [73] pozwalają na dokonanie **dekompozycji** procesu oceny sytuacji i wyborów działań na następujące podprocesy charakteryzujące się mniejszym stopniem złożoności: pozyskiwanie informacji (*information acquisition*) tzn. poszukiwanie i przechowywanie informacji, ocenę alternatyw (*evaluation*), działanie i uczenie na podstawie obserwacji efektów działania (*feedback*). Według badań przedstawionych przez Eoma [75] można określić następujące **zasady kierujące działaniami decydentów**, którzy podejmują suboptymalne decyzje korzystając z wyników analizy ponoszonych kosztów w proporcji do spodziewanych zysków. Okazuje się, że decydenci działają w kierunku **maksymalizacji dokładności, jakości decyzji** ale także **minimalizacji wysiłku** (kosztów i czasu) poznawczego. Dlatego też nie zawsze większa liczba informacji jest przydatna dla podejmowania decyzji. Jak twierdzą Todd i Benbasat [297] decydenci korzystają z **systemów wspomagania decyzji** w celu zmniejszenia nakładu pracy związanego z wykonaniem zadania.

Ontologia według van Heijsta [310] **jest specyfikacją** (czyli zbiorem założeń o rzeczywistym systemie) **poziomu wiedzy** (w sensie modelu systemu) dotyczącej konceptualizacji dziedzinowej. Ontologia (hierarchia pojęć) zajmuje się strukturą i słownictwem wiedzy dziedzinowej. Szczególną rolę ontologia odgrywa w początkowym etapie konstruowania KBS ze względu na **przeprowadzanie formalizacji wiedzy** pozyskiwanej od eksperta. Jak pisze Xindong [330] ontologia **wymaga udziału doświadczonego eksperta** dziedzinowego, który uczestniczy w procesie formalizacji wiedzy i orzeka o poprawności reprezentacji wiedzy. Baza wiedzy (reguł i faktów) konstruowana jest za pomocą **pojęć wyszczególnionych w ontologii** oraz operacji logicznych.

Wyróżnić można następujące ontologie reprezentacji wiedzy stosowane w regułowych systemach eksperckich:

- Trójka: obiekt – Atrybut – Wartość (OAV) – stosowana w językach rodziny OPS5
- Reguły predykatów zastosowane w języku Prolog
- Uogólnienie reguł Prologu postaci: schemat reguły + ciało reguły opisane przez Xindonga [330] rolę atrybutu pełni w nim tzw. faktor, który może być predykatem lub zmienną przyjmującą wartości dyskretne lub ciągle. Ciało reguły może zawierać współczynnik pewności. Cechą szczególną tej metody jest traktowanie opisu schematu wiedzy dziedzinowej jako integralnej części wiedzy dziedzinowej za pomocą schematów reguł. Fakty w KE-shell powstają przez uzupełnienie faktora (odpowiednika atrybutu) o jego wartość. Faktowi towarzyszy współczynnik pewności (*certainty factor*) przedstawiający prawdopodobieństwo lub czynnik rozmycia (*fuzzy*).

3.3 Wiedza a informacja i zarządzanie wiedzą

Mechitov i Moshkovich [191] rozróżniają dwa rodzaje reprezentacji wiedzy: za pomocą **reguł wnioskowania** (*production rules*) oraz za pomocą **przykładów** (wyników eksperymentów) przedstawiających poszczególne obiekty dziedziny wiedzy. Przyjmują za Hayes-Rothem i innymi [111], że celem konstruowania reguł jest **uogólnianie i strukturalizowanie** wiedzy eksperta. Należy zgodzić się z nimi, że **wady reprezentacji regułowej** pojawiają się w przypadku gdy ekspert dziedzinowy nie jest w stanie wyrazić każdej możliwej sytuacji. Nawet jeśli ma jednocześnie doświadczenie inżyniera wiedzy, to zbudowana przez niego baza wiedzy może **zawierać luki** (sytuacje nieprzewidziane), **sprzeczności** (część z nich można usunąć podczas weryfikacji i walidacji, ale nie jest to łatwe) i **efekty osobistych poglądów** (nie będzie w pełni obiektywna). Zaletą jest natomiast **łatwość odwzorowania** przez inżynierów wiedzy **teorii** wyrażanych przez **ekspertów** jednakże, jak pisze Park [1995] milczące „założenie o znajomości i przestrzeganiu przez eksperta szczegółów reprezentacji wiedzy, takich jak *backward chaining* czy *constraint satisfaction*” może prowadzić do **trudności w pozyskiwaniu wiedzy**. Problemy spełnionych ograniczeń (więzów), *constraint satisfaction* [231] takie np. „problem n-królowych” charakteryzują się tym, że nie można ich zdekomponować na części składowe czyli podproblemy. Drugi typ reprezentacji wiedzy polega na **zestawieniu przykładów** wyników eksperymentów. Jak piszą Mechitov i Moshkovich [191] jest to **bardziej naturalna forma** gdyż łatwiej ekspertowi dziedzinowemu wypowiedzieć się na temat poszczególnych przykładów, każdego z osobna. Dzieje się tak dlatego, że **ekspert nabywa doświadczenie** przez kontakt i obchodzenie się z konkretnymi, praktycznymi sytuacjami i rozwiązywanie konkretnych problemów. Eksperci (jak wszyscy ludzie) mogą posługiwać się tutaj analogiami, porównując „nowy”, nieznaną przypadkiem z przechowywanymi w pamięci znanymi przykładami, z którymi sobie już „poradzili”. Oczywiście wydane w ten sposób oceny będą jednak zawierały jakiś stopień niepewności. Na podstawie odpowiedniej liczby ocenionych „zdiagnozowanych” przypadków formułuje się reguły decyzyjne (np. bierze się pod uwagę podobieństwo).

Według Arshama [14] wiedzę można zdefiniować jako **wiarygodną informację**, która ma **potencjalną wartość dla organizacji**. Celem zdobycia wiedzy jest zatem zdobycie przewagi konkurencyjnej przez podniesienie wydajności i skuteczności. Odbywa się to poprzez poprawę wyników na różnych poziomach zarządzania:

1. Poszczególnych procesów (np. czas potrzebny na wyprodukowanie określonej partii produkcyjnej, szybkość reakcji na zapotrzebowanie klienta)
2. Funkcji zarządzania (wyniki poszczególnych jednostek organizacyjnych)
3. Całościowej działalności przedsiębiorstwa (dynamika sprzedaży, zysku, udział w rynku).

Liebowitz [176] podkreśla (za twórcą pojęcia kapitału intelektualnego Hubertem St. Onge) **różnice**

między informacją a wiedzą. Informacją stają się **zestawy danych** po przybraniu przez nie **założonej formy**, struktury, czyli uporządkowane. Natomiast wiedzą **informacja** staje się wtedy, gdy umożliwia **wykonywanie działań i podejmowanie decyzji** na podstawie sformułowanych wniosków. Jeśli posiadana wiedza jest na tyle bogata i jednocześnie „operacyjna” że pozwala na **podejmowanie właściwych decyzji** to nazwać można ją **mądrością**, znajdującą się na najwyższym stopniu piramidy dane – informacja – wiedza - mądrość [115] przynależną systemom o najwyższym poziomie inteligencji (ludziom). Wiedzę wzbogaca **doświadczenie eksperckie** pozwalające na interpretację i wykorzystanie zbiorów pozornie odosobnionych faktów poprzez zastosowanie reguł wnioskowania (zdroworozsądkowych). Biorąc pod uwagę kryterium możliwości przekazywania można wyróżnić wiedzę o sposobach jej zastosowania (*tacit*) w podejmowania decyzji oraz wiedzę o faktach i zdarzeniach (*explicit*). **Wiedza ukryta**, cicha (*tacit knowledge*) [131] należy przynajmniej częściowo do sfery podświadomości, korzysta się z niej właściwie bez konieczności uruchamiania procesów myślowych. Niestety trudno ją wydobyć na światło dzienne w celu przechowywania i wykorzystania w systemie informatycznym. Według Jakubowskiego [131] „wiedza cicha jest przekazywana słownie i poprzez wspólne doświadczenia” a przykładem jej dwustronnej transmisji są „stosunki mentor-protegowany”. Kolejną wadą wiedzy ukrytej są trudności w posługiwaniu się nią przez większe zespoły ludzkie. Jak podaje Liebowitz [176] z wiedzą ukrytą związany jest (szkodliwy) paradoks inżynierii wiedzy (procesu wydobywania wiedzy od eksperta i jej modelowania w postaci reguł). Polega on na obserwacji, że im ekspert jest lepszy, im więcej posiada wiedzy o dużym stopniu „kompresji” np. zdroworozsądkowej, składającej się z reguł trudnych do wyjaśnienia, tym trudniej ją od eksperta wydobyć, sformalizować i wykorzystać. Tym większe trudności ma sam ekspert w jej wyrażeniu, sprecyzowaniu i wyjaśnieniu. A zatem **wiedza ukryta trudniej poddaje się przeniesieniu** z „naturalnego miejsca” (umysłu eksperta) do sztucznie przygotowanych systemów (*Knowledge-Based Systems*). Polanyi [235] wiedzę ukrytą określił w następujący sposób "wiemy więcej niż jesteśmy w stanie powiedzieć". **Chętniej widziany** przez inżynierów wiedzy jest drugi rodzaj wiedzy, wiedza formalna, uzewnętrzniona (*explicit*). **Łatwo ją dokumentować** i przekazywać natomiast jej **wadą jest podatność na różne interpretacje** przez użytkowników. Wymaga także (w przeciwieństwie do wiedzy ukrytej) informacji jak ją połączyć z wiedzą pochodzącą z innych źródeł, zdobytą w innym czasie, innym miejscu. Jak twierdzi Liebowitz [176] organizacje korzystają z dwu sposobów pozyskiwania i współdzielenia się wiedzą (obu rodzajów): **kodyfikacji i personalizacji**.

Kodyfikacja polega na formalizacji i dokumentowaniu wiedzy w przyjętym w organizacji systemie zarządzania wiedzą (docelowo mają to być reguły, przypadki, procedury optymalizacji, klasyfikacji). **Kodyfikacja jest ukierunkowana na dokumenty** [138]. Skodyfikowana wiedza ma postać raportów, korespondencji, formularzy (interpretowanych *off-line*). Jest to strategia przydatna gdy wiedza pochodzi z **dokumentacji przedsiębiorstwa** (sprawozdawczości).

Personalizacja jest ukierunkowana na komunikację. Posiadana wiedza (a także mniej złożone

formy) wymieniana jest za pośrednictwem rozmów prowadzonych bezpośrednio lub za pośrednictwem **różnorodnych mediów** (w czasie rzeczywistym) – dzięki temu może odwzorować aspekty **które są pomijane** ze względu na naturę sformalizowanej komunikacji za pomocą dokumentów. **Personalizacja** jest niezbędna gdy w grę wchodzi **wymiana poglądów**, dyskusja, prezentacja metod rozwiązywania danego problemu oraz czynniki jakościowe. Dobór strategii zależy przede wszystkim od charakteru pozyskiwanej wiedzy – zwykle któraś ze strategii pozwoli na szybsze wykonanie zadania, jednak w przypadku personalizacji dużo czasu może wymagać ustalenie terminu spotkania.

3.4 Rola uczenia maszynowego

Wydobywanie na światło dzienne **ukrytych praw**, określających nieznaną dynamikę zjawisk było zawsze w centrum zainteresowania nauki. Odkryta i uporządkowana dzięki nim wiedza umożliwia bowiem wyjaśnienie rzeczywistości i przewidywanie przyszłości. **Zdolność uczenia się** jest jednym z **przejawów inteligencji**. Pozostałe, mniej zaawansowane to według Gilberta i innych [95] umiejętność wyboru działań zgodnie z preferencjami, wnioskowanie i planowanie działań. Uczenie się polega na **uzupełnianiu, modyfikowaniu i wartościowaniu wiedzy** na podstawie **doświadczeń**. Systemy komputerowe (oprogramowanie komputerowe) korzystające z technik uczenia maszynowego zdolne są do przystosowania się do aktualnej sytuacji (tzn. odpowiedniego działania) i uczenia się na podstawie doświadczeń z przeszłości. Potwierdza to Moore [206] pisząc „jest to podejście oparte na rozważaniu danych, mające na celu wyekstrahowaniu wiedzy z poprzednich decyzji”. Uczenie maszynowe to druga, obok **wydobywania wiedzy** od eksperta dziedzinowego, **metoda pozyskiwania wiedzy**. Jej podstawą jest rozumowanie przez analogie i **odkrywanie pewnych prawidłowości**, wzorców (reprezentujących kategorie, wyodrębnione grupy do których należą poszczególne przypadki) na podstawie obserwowanych i analizowanych zjawisk (w szczególnym przypadku: rozwiązanych w przeszłości problemów) odwzorowanych poprzez zebrane dane. **Rozumowanie przez analogie** oznacza prowadzenie logicznego wnioskowania na podstawie wyodrębnionych podobnych aspektów dwu zjawisk, pomijając te które są odmienne. Podczas uczenia maszynowego przykłady (reprezentowane przez atrybuty wejściowe) wykorzystywane do uczenia systemu są przetwarzane (poprzez indukcję, np. kolejne wersje algorytmów Quinlana) na poszczególne klasy. Jak stwierdza Moore [206] dzięki systemom działającym na zasadzie „*learning-by-example*” możliwe jest „rozpoznawanie wzorców” użyteczne podczas podejmowania decyzji, np. szacowania zdolności kredytowej wnioskodawcy czy prognozowania wartości indeksów giełdowych. **Motywacją zastosowania uczenia maszynowego** (indukcyjnego formułowania reguł, *rule induction*) jest obserwacja że ekspertom często łatwiej przedstawić reprezentatywne przykłady na podstawie których dokonują klasyfikacji (wynik wnioskowania) niż od razu przedstawić reguły wnioskowania (sposób wnioskowania). Jak pisze Moore [206] po przedstawieniu bazy reprezentatywnych przypadków przystępuje się do **automatycznego generowania reguł** (*rule induction*), lub przeprowadza analizę MDA, Logit czy Probit. Zróżnicowane zastosowanie (ze względu na konstrukcję zbioru parametrów opisującego warianty decyzyjne) odpowiada zróżnicowanym technikom uczenia maszynowego. **Uczenie nadzorowane** (supervised learning) odpowiada dziedzinie podejmowania decyzji o charakterze dyskretnym (klasyfikacja) oraz ciągłym (predykcja). **Uczenie ze wzmocnieniem** (*reinforcement learning*) - polega na obserwacji przez komputerowego agenta stanów obiektów należących do otoczenia i wykonywania na ich podstawie działań określonych przez politykę decyzyjną; za działania zgodne z polityką decyzyjną agent jest nagradzany (znajduje zastosowanie w

problemach, których rozwiązanie polega na wyznaczeniu sekwencji elementarnych decyzji). **Uczenie bez nadzoru** (*unsupervised learning*) – ma na celu wydobycie informacji o wewnętrznej statystycznej strukturze zjawiska. W tym trybie sieć neuronowa, będąca nieliniowym, nieparametrycznym modelem wnioskuje na podstawie wewnętrznej struktury informacji wejściowych – wykrywa i samodzielnie tworzy opisy kategorii przydatnych podczas wykonywania zadania klasyfikacji; zadanie to polega na wygenerowaniu odpowiednich wartości (sygnałów) wyjściowych na podstawie danych wejściowych.

Uczenie maszynowe charakteryzuje się **zautomatyzowanym analizowaniem** (za pomocą metod rozumowania logicznego i obróbki statystycznej) dostarczonych przykładów, na podstawie których konstruowany jest **model dziedziny wiedzy** (zbiór elementarnych pojęć, twierdzeń-predykatów, operatorów wraz z przestrzenią rozwiązań problemów) oraz brakiem niezbędności dostępu do sformalizowanych pojęć podręcznikowych (abstrahując od kontekstu społecznego i technologicznego czyli wnioskowania bezkontekstowego). Porównując pozyskiwanie wiedzy wyrażanej przez ludzkiego eksperta i za pomocą uczenia maszynowego można zauważyć, że zalety uczenia maszynowego to [320]: obiektywizm (ludzki ekspert może pominąć niektóre fakty i reguły) oraz zdolność do odkrywania wiedzy (korzystanie z wiedzy ukrytej, potrzebnej w radzeniu sobie z problemami których żaden ekspert jeszcze nie rozwiązał). Wiedza pozyskiwana jest **metodami indukcyjnymi** lub znanymi ze **sztucznych sieci neuronowych**. Efektem zastosowania metod indukcyjnych są heurystyki (kojarzące wartości atrybutów z odpowiednimi klasami obiektów, reprezentujących poszczególne klasy np. diagnoz lub decyzji), pomocne w procesie klasyfikacji nowych obserwacji (przypadków) na podstawie wartości ich atrybutów. **Heurystyka** wg Kahnemana i innych [139] jest sformułowanym (za pomocą określonej terminologii) sposobem wyznaczenia rozwiązania spełniającego określone warunki, reprezentatywnego i wykonalnego; powinna także zostawiać pewien margines na autonomię decyzyjną decydenta. **Przykładem metod indukcyjnych** jest algorytm ID3 opisany przez Quinlana [244] działający na zasadzie poszukiwania tych atrybutów i wzorców (relacji między atrybutami), które umożliwiają klasyfikację obiektów. Wynikiem działania algorytmu są posiadające własność zwięzłości (minimalna liczba atrybutów czyli w konsekwencji najmniejsza liczba pytań) reguły wnioskowania (klasyfikacji). Pozyskane w powyższy sposób heurystyki przydatne są przede wszystkim w **klasyfikacji** (opracowania diagnoz lub dokonywania wyboru) przypadków (opisanych za pomocą zestawu atrybutów) które nie znalazły się jeszcze w bazie przypadków. Jak podaje Moore [206] proces indukcji jest **podejściem generalizującym**, wychodzącym od szczegółu (przykładów) do ogółu (reguł). Zaletą metod indukcyjnych w porównaniu do motywowania eksperta do samodzielnego (lub z udziałem inżyniera wiedzy) wyrażenia praw i związków rządzących dziedziną problemową jest łatwiejsze wydobycie od eksperta zestawu przykładów uczących wraz z diagnozą i na tej podstawie wnioskowanie o sposobie podejmowania decyzji przez eksperta (wyrażenie aspektu „co to jest” zamiast „jak to się stało”).

3.5 Sztuczna inteligencja + psychologia kognitywna – definicja

Definicja sztucznej inteligencji, podana przez Nowaka [220] określa sztuczną inteligencję jako „zbiór technik informacyjnych, które nadają komputerowi podobne ludziom zdolności widzenia, słyszenia, mówienia i rozumienia przy niepełnej i nieprecyzyjnej informacji oraz zdolność uczenia się”.

3.5.1 Systemy wspomaganie decyzji – sieci neuronowe

Sieci neuronowe (*SSN*) są **modelami odwzorowań**, które mają **wykorzystać ukrytą informację** zawartą w szeregach czasowych dzięki jej reprezentacji za pomocą **warstw** składających się z **węzłów** oraz **sparametryzowanych połączeń** między nimi. SNN służą przede wszystkim prognozowaniu na podstawie zestawu danych historycznych oraz ciągów uczących lub rozwiązywaniu **zadań klasyfikacji**. Użyteczność sieci neuronowych wynika ze **zdolności generalizacji**, czyli uzyskiwaniu dzięki nim **odpowiedzi na pobudzenie nieznanym wcześniej sygnałem**. Generalizacja (uogólnianie) odbywa się na podstawie odpowiednio dostosowanego do struktury danej sieci neuronowej zestawu trenującego czyli ciągu uczącego. **Wady generalizacji** z wykorzystaniem sieci neuronowej to zależność jakości rozwiązań od klasy problemu i konieczność doboru właściwej struktury – typu sieci. Jak stwierdza Watkins [318] systemy eksperckie czy logika rozmyta przechowują i przetwarzają stosunkowo **dobrze ustrukturalizowaną wiedzę** – posługując się pojęciami logiki, natomiast sieci neuronowe operują na wiedzy **całkowicie pozbawionej struktury** – całe przetwarzanie wiedzy odbywa się na podstawie danych numerycznych. Projektant aplikacji sieci neuronowych musi **dokonać wyboru** spośród szeregu typów architektury sieci (np. liczby i typu warstw sieci, sposobu połączeń, propagacji), technik trenowania oraz rozważyć zastosowanie i wartości parametrów sieci (postać funkcji błędu, standaryzacji wejść). Pozostałe **słabe strony sieci neuronowych** to według Dutty [67] konieczność operowania na ciągu uczącym znacznej wielkości oraz jedynie empiryczna możliwość oceny wiarygodności osiągniętych dzięki niej wyników.

Według Jo i Hana [134] sieci neuronowe stosowane są już od końca lat osiemdziesiątych XX wieku w **zadaniach klasyfikacji** (przewidywania bankructwa przedsiębiorstwa). Sieci neuronowe przydatne są także w prognozowaniu i zadaniach typu *pattern matching* (szczególnie w zadaniach oceny zagrożenia bankructwem). Ahn i inni [4] wymieniają następujące **przykłady zastosowań** sieci neuronowych: ocenę zabezpieczeń kredytów hipotecznych [288], wykrywanie prób oszustw związanych z transakcjami dokonywanymi za pomocą kart kredytowych, prognozowaniem ratingu papierów dłużnych emitowanych przez firmy [66], ocenę ryzyka związanego z pozytywnym rozpatrzeniem wniosków o wydanie karty kredytowej. Znane są także zastosowania sieci neuronowych w **przewidywaniu warunków prowadzenia biznesu**, prognozowania notowań rynkowych akcji, wydobywania ukrytej informacji ze sprawozdań finansowych, oceny stopnia zagrożenia

niepowodzeniem biznesowym na podstawie zestawu wskaźników finansowych firmy [257]. Doniesienia literaturowe dotyczące zastosowań sieci neuronowych można znaleźć także w artykułach [241],[243], [271], [299].

Sztuczne sieci neuronowe są techniką umożliwiającą **statystyczne odwzorowanie** [268] związków przyczynowo-skutkowych w rzeczywistych systemach. SSN oparte są na przykładach wykorzystywanych w procesie trenowania (uczenia). Z punktu widzenia użytkownika, jak stwierdza Najim i inni [212] sieci neuronowe wykonują **zadanie predykcji** (przewidywania stanu obiektów w przyszłości) w trybie wielokrokowym. Ich zaletą w porównaniu do innych technik statystycznych jest **zdolność do uogólniania zjawisk**, które są obserwowane po raz pierwszy. Możliwe jest to dzięki zdolności sieci neuronowych do **uczenia się na podstawie przykładów** oraz odwzorowania ukrytych i nieliniowych zależności w kolejno prezentowanych sieci neuronowej przykładach. Co więcej, SSN **nie jest potrzebna znajomość** tych zależności – zdolna jest do odwzorowania ich poprzez dostosowanie połączeń zawartych w jej strukturze. Jako przykład może służyć artykuł Hutchinsona i innych [122] opisujący w jaki sposób sztuczna sieć neuronowa odwzorowała (nieznana jej wcześniej) formułę Black-Scholesa [21] (stanowiącej że po spełnieniu pewnych warunków cena instrumentów pochodnych może być wyrażona jako nieliniowa kombinacja pewnych czynników i że wykonalne jest zbudowanie portfela dokładnie odwzorowującego strukturę wypłat (*payoff structure*) finansowych instrumentów pochodnych) na podstawie dwuletniego zestawu danych codziennych cen opcji.

Zasada działania podstawowych elementów sztucznej sieci neuronowej – neuronów została zapożyczona ze **zwierzęcego układu nerwowego**. Biologiczny **neuron** działa w następujący sposób: **dendryty** (jest ich wiele), czyli wypustki cytoplazmatyczne przekazują odebrane od innych neuronów informacje, następnie **neuron** przetwarza je wykorzystując w tym celu **pewną funkcję** i jeśli sygnał wejściowy jest powyżej pewnej wartości nazywanej **progiem pobudzenia** to przekazuje informację dalej, wysyłając ją poprzez wyjściową wypustkę cytoplazmatyczną, **akson** (pojedynczy). **Regulowanie wartości** odbieranych sygnałów możliwe jest dzięki działaniu **synaps** (łączników między dendrytami a aksonami), które posiadają zdolność zmiany tzw. **wag synaptycznych**. Neuron zdolny jest do modyfikowania swojego działania na podstawie przychodzącej informacji, czyli **posiada zdolność uczenia się**. Zazwyczaj SSN konstruowana jest zgodnie z topologią trzech [81] kolejno następujących po sobie warstw: **wejściowej**, **ukrytej** i **wyjściowej**. Każda warstwa zawiera proste elementy – neurony, których zadaniem jest wykonanie elementarnych obliczeń. Neuron przekształca dane (sygnały) ze swoich wejść (mogą być one tożsame z wyjściami poprzedniej warstwy neuronów) na tzw. poziom aktywacji (jest to wartość progowa, oblicza się ją na podstawie wejść i przypisanych im wag) i wyprowadza (wysyła) go na każde ze swoich wyjść. Najprostszy przypadek to uzyskiwanie na wyjściu neuronu znaku wyrażenia $\mathbf{w}^T \mathbf{x}$ gdzie \mathbf{w} jest wektorem wag a \mathbf{x} jest wektorem wejść. Działanie neuronu (posiadającego już ustalone wartości wektora wag) polega w tym przypadku na **zaklasyfikowaniu n-wymiarowego wektora** do jednego z dwóch rozłącznych zbiorów (klas) na

podstawie wartości wyjścia: wartość „1” oznacza przypisanie do klasy pierwszej natomiast wartość „-1” do klasy drugiej. **Poziom aktywacji** oblicza się w dwu etapach: pierwszy ma charakter przekształcenia liniowego, wynikiem jest wartość tzw. **potencjału komórki** – jest to ważona suma wartości podanych na wejście neuronu: $net_j = \sum w_{ij} * out_i$. Drugi etap polega na przekształceniu nieliniowym ważonej sumy (obliczonej w poprzednim etapie) zgodnie ze wzorem: $f(net_j) = 1 / (1 + \exp(-k * net_j))$ którego wartości należą do zakresu (0,1) oraz k jest liczbą dodatnią, odpowiedzialną za rozrzut wartości funkcji.

Przed zastosowaniem (tzn. użyciem w zadaniach klasyfikacji) sieci neuronowej przeprowadzany jest **proces uczenia** (trenowania) sieci. Jeśli jest on **odpowiednio dostosowany** do danego problemu klasyfikacji, pozwoli na uzyskiwanie prawidłowych odpowiedzi sieci nawet w sytuacji danych niekompletnych czy poddanych zakłóceniom. Algorytm trenowania (uczenia sieci) określany jako „*backpropagation rule*” (bez warunku stopu) może być przedstawiony w następujący sposób [260] (1) Inicjalizowanie wartości wag W, (2) Wykonywanie w pętli kroków 3-4, (3) Obliczanie gradientu funkcji uśrednienia błędu AQ(DL, W) na podstawie wszystkich danych DL zawartych w ciągu uczącym, (4) Przypisanie nowych wartości wag poprzez zmniejszenie ich o iloczyn gradientu i współczynnika szybkości uczenia się.

Pierwszy etap polega na **przedstawieniu sieci przykładów** reprezentowanych przez wartości wejściowe (przekazywane bezpośrednio na wejście pierwszej warstwy neuronów) i odpowiadające im wartości wyjściowe. Następnie neurony kolejnych warstw wykonują obliczenia (**etap liniowy**) aż do warstwy wyjściowej, w której następuje **porównanie wyliczonych wartości z wartościami ciągu uczącego**. Ich różnica, traktowana jako **wartość błędu** przekazywana jest **z powrotem do poprzednich warstw** w celu wyliczenia nowych wartości wag w taki sposób żeby prowadziły do **minimalizacji wyrażenia** (funkcji błędu): $Err = 1/2 * \sum (T_j - out_j)^2$. Im większa wartość błędu, **tylko silniejsza reakcja** w postaci zmiany wag towarzyszących neuronom. Kolejne iteracje algorytmu skutkują niewielkimi zmianami wag, coraz mniejszymi, ale prowadzącymi do poszukiwanej wartości. Można ogólnie zapisać że dostosowanie wag $w_{ij}(t+1) = w_{ij}(t) + \eta * \delta_j * out_j$, gdzie η jest szybkością uczenia się, a δ_j jest funkcją błędu związaną z neuronem j (jej postać zależy od warstwy neuronu). Algorytm *backpropagation* może prowadzić do **utknięcia w lokalnym minimum**, stosuje się zatem wiele **jego modyfikacji**, na przykład **stopniowe zmniejszanie szybkości uczenia się** zgodnie z wyrażeniem: $\eta = 1 - \ln(1 + t/t_{max})$. Uczenie sieci neuronowych kończy się wraz z osiągnięciem warunku stopu, który może być wyrażony jako odpowiednio mała suma zmian wszystkich wag lub wykonanie ustalonej liczby iteracji.

3.5.2 Systemy wspomaganie decyzji – metody wnioskowania symbolicznego

Zgodnie z definicją podaną przez Lucasa i Gaag [179] “**systemy eksperckie** są zdolne do oferowania rozwiązań określonych problemów w danej dziedzinie lub do udzielania porad na poziomie i w podobny sposób jak czynią to eksperci dziedzinowi”. **Regułowe systemy eksperckie** odróżniają się od **konwencjonalnych** (proceduralnych) języków programowania **opisem dziedziny problemowej**. Systemy eksperckie (*production systems, rule-based systems*) mają na celu gromadzenie wiedzy w postaci faktów i reguł wnioskowania modelujących pewną dziedzinę. Zamiast sekwencyjnego zestawu procedur regułowy system ekspercki zawiera **zestaw reguł**. Wywoływanie poszczególnych reguł odbywa się dzięki mechanizmowi wnioskowania w taki sposób, aby **aktywowane były reguły niezbędne w danej sytuacji** a nie jak w przypadku języków proceduralnych – sekwencyjnie. Mechanizm wnioskowania próbuje **dopasować odpowiednią gałąź** w strukturze reguł do poszczególnych badanych przypadków (wprowadzonych przez użytkownika w postaci odpowiedzi na szereg pytań lub sformatowanego opisu). Systemy eksperckie mają specyficzną strukturę, przystosowaną do przetwarzania wiedzy wydobytej (pozyskanej) od eksperta dziedziny. Dzięki wiedzy tej możliwe jest automatyczne uzyskiwanie ekspertyz pozwalających na wspomaganie procesów decyzyjnych. Jak pisze Jakubczyc i inni [130] **generowanie wiedzy** polega na opracowaniu rozwiązania problemu (weryfikacji hipotezy) zarówno na podstawie wiedzy pozyskanej od eksperta dziedziny jak i wyników aktywnego (automatycznego) przetwarzania oraz informacji zawartych w bazie danych i aktualizowanej bazie faktów. Systemy eksperckie jako metoda rozwiązywania problemów należą do metod **sztucznej inteligencji**. Tworzą wraz ze zbliżonymi metodologiczne metodami (takimi jak programowanie w logice Horna (język Prolog), rami i systemy zachowania spójności logicznej [281]) grupę **metod wnioskowania symbolicznego** [67].

Ramy wg Minsky’ego [201] są “strukturą danych stworzoną w celu reprezentacji pewnej stereotypowej sytuacji i wyposażoną w różnorodne informacje takie jak sposób wykorzystania ram, co się stanie z ramą w przyszłości, co się stanie jeśli te prognozy nie sprawdzą się”. **Poszczególne rami** zawierają gniazdko (*slot*) służące do połączeń z innymi ramami. **Kolekcja ram** tworzy **system ramowy** (*frame-system*), w którym sensowne są pojęcia takie jak np. odległość, podobieństwo, przeciwieństwo, poprzedzanie. Implementacje reprezentacji wiedzy za pomocą ram korzystają z **podejścia obiektowego**, dzieje się tak np. w systemie eksperckim CLIPS czy KAPPA wyposażonym w **mechanizm wnioskujący** oraz **reprezentację obiektową**. **Systemy zachowania spójności logicznej** korzystają z koncepcji **inteligentnych agentów** i według Paluszyńskiego [229] konstruowane są w celu zapewnienia następujących udogodnień: nieprzerwana aktualizacja i konserwacja faktów oraz wynikających z nich wniosków, sporządzanie uzasadnień faktów, obsługa niespójności założeń i wnioskowanie o charakterze niemonotonicznym (analogicznym do sposobu

rozumowania człowieka). Wśród **słabości metod wnioskowania symbolicznego** za najważniejsze można uznać **rozwlekłość zapisu** (przenośne i elastyczne bazy reguł są bardzo obszerne), **problemy** związane z **pozyskiwaniem wiedzy** od ekspertów oraz **walidacją** oraz przydatność raczej w **badaniach wykonalności** niż zagadnieniach optymalizacji (brak metodologii porównań poszczególnych rozwiązań między sobą).

Pozyskiwane wiedzy jest procesem w którym mają miejsce wielokrotne interakcje między inżynierem wiedzy (obserwatorem) a ekspertem dziedzinowym. Zanim inżynier wiedzy dokona transformacji wiedzy eksperckiej do formatu akceptowanego przez system ekspercki musi zmienić stan pewnej części (porcji) swojej wiedzy o dziedzinie aby zbliżyć ją do wiedzy eksperta. Jest to postępowanie analogiczne do uczenia się i obejmuje zapoznanie się z terminologią i problemami dziedzinowymi. Następnie inżynier wiedzy prowadzi studia literaturowe oraz często długotrwałe rozmowy z ekspertem (lub ekspertami – w przypadku architektury tablicowej i wielu źródeł wiedzy. Po wdrożeniu systemu eksperckiego zadania eksperta są przejmowane przez moduł sztucznej inteligencji a miejsce inżyniera wiedzy zajmuje użytkownik konsultujący się z systemem eksperckim.

Jak pisze Liebowitz [172] **pozyskiwanie wiedzy** i doświadczeń przez człowieka od systemu eksperckiego realizowane jest przez **trzy podstawowe komponenty**. Użytkownik spotyka się bezpośrednio ze **strukturą dialogową** (komponentem interfejsu użytkownika). We współczesnych SE interakcję z użytkownikiem ułatwia **środowisko graficzne GUI**. Komponent **mechanizmu wnioskowania** steruje strategiami poszukiwania rozwiązań (testowania hipotez) prowadząc do osiągnięcia konkluzji. Wnioskowanie przeprowadzane jest na podstawie bazy wiedzy natomiast użytkownik **kieruje nim** za pomocą interfejsu użytkownika. **Komponent bazy wiedzy** jest odpowiedzialny za zarządzanie faktami (pochodzącymi z baz danych, konsultacji z użytkownikiem oraz wyprowadzonych z innych faktów na podstawie reguł) i regułami (jest to skodyfikowana wiedza dziedzinowa oraz zdroworozsądkowa). Oczekuje się bazy wiedzy kompletnej, spójnej i wiernej.

Wszystkie podstawowe komponenty korzystają i komunikują się za pomocą **komponentów pomocniczych**. W ten sposób określa się **pamięć roboczą** (przechowuje ścieżkę wnioskowania, informacje o branych pod uwagę regułach i faktach), wspomniane już bazy danych, interfejsy już istniejących systemów informatycznych (umożliwiający także zastosowanie struktur hybrydowych) oraz komponenty zarządzające SE, obsługujące sytuacje wyjątkowe (błędy) i umożliwiające komunikację między komponentami.

3.6 Systemy eksperckie - zastosowanie w analizie finansowej

Shao [267] wyróżnia 5 grup **zastosowań systemów eksperckich w analizie bankowej**: ocenę zdolności kredytowej (w tym ocenę możliwości spłat oraz zabezpieczeń), obsługę wniosków kredytowych (w tym badanie historii kredytowej oraz kontrolowanie kredytobiorcy w trakcie spłat), wykrywanie nieprawidłowości (ostrzeżenie przed oszustwami pracowników oraz osób z zewnątrz: np. realizacji unieważnionych czeków, sfałszowane karty kredytowe, pranie pieniędzy), obsługę klienta (doradztwo finansowe i biznesowe, oferowanie produktów) oraz diagnostykę błędów w systemie informacyjnym (sprzęt i oprogramowanie). Przykładem środowiska badawczego dla zastosowań systemów eksperckich w obszarze analizy finansowej jest opracowany w Technical University of Crete [188] system *FINEVA*. Jest to narzędzie analizy wielokryterialnej przedsiębiorstwa, głównie badające **ryzyko zagrożenia bankructwem**. Zostało zrealizowane w postaci **hybrydy** modułu **systemu eksperckiego** (wykorzystującego regułową reprezentację wiedzy), modułu **analizy statystycznej** oraz modułu **analizy wielokryterialnej**. Analogicznie jak w pakiecie Ekanwin, w przypadku *FINEVY* zastosowano zewnętrzny moduł systemu eksperckiego (M4 firmy Cimflex Teknowledge). Bazę wiedzy modułu systemu eksperckiego wyposażono w dużą liczbę reguł wnioskowania (ilościowych i jakościowych) i przejrzysty interfejs wyjaśniający, **nie zastosowano** jednak **modułu wizualizacyjnego** oraz **środowiska eksperymentów symulacyjnych** (*FINEVA* korzysta ze sprawozdań finansowych dostarczanych z zewnątrz). Przykładem komercyjnego **hybrydowego oprogramowania** zawierającego moduły systemu eksperckiego, modelu symulacyjnego oraz modułu graficznej prezentacji danych jest Aitech DSS (wcześniej ISAF) [5], którego **zadaniem** jest monitoring (w tym wczesne ostrzeżenie o zagrożeniach) procesów gospodarczych w przedsiębiorstwie w ujęciu finansowym: zachowania płynności, rentowności, zachowania zadłużenia w bezpiecznych granicach, planowanie finansowe (z wykorzystaniem metod statystycznych), sporządzanie raportów i analiz (interpretacji) na podstawie wiedzy pozyskanej od ekspertów dziedzinowych, szacowanie efektów poszczególnych wariantów decyzyjnych z uwzględnieniem w modelu czynnika losowości (reprezentującego ryzyko) oraz prezentacja wyników eksperymentów symulacyjnych za pomocą zestawu konfigurowalnych wykresów wskaźników i wielkości ekonomicznych (odwzorowanych w sprawozdaniach finansowych).

Jak piszą Matsatsinis i inni [188], technologia **systemów eksperckich** jest **dobrze przystosowana** do wspomagania **typowych zadań analizy finansowej**, mających na celu identyfikację silnych i słabych stron firmy. Analiza finansowa polega przede wszystkim na **interpretacji wartości wskaźników finansowych** oraz wyrażeniu matematycznych zawierających informacje ze sprawozdań finansowych oraz ocenie **czynników jakościowych** [213]. Według Thomasa i innych [295] opracowanie **narzędzi eksperckich** wspomagających bankowy proces oceny wniosków kredytowych **pozwala utrzymać konkurencyjność** w warunkach kilkudziesięciokrotnego

zwiększenia wartości akcji kredytowej. Na podstawie badań Baldwin i Sangstera [17] można stwierdzić że **systemy eksperckie** wpływają **pozytywnie na szereg wymiarów** (ilościowych i jakościowych) działalności kredytowej w bankach, takich jak wydajność, skuteczność, ekspertyza, szkolenie, konkurencyjność na poziomie organizacji, poszczególnych pracowników jak i wykonywanych zadań. Obserwowano zatem **zmniejszenie kosztów** (zwiększenie wydajności organizacyjnej) w zakresie transakcji kartowych i transferów pieniężnych (system ekspercki w Security Pacific National Bank, *CitiExpert* w Citibanku, *ES FALCON* w Colonial National Bank), poprawę indywidualnej wydajności pracowników kredytowych (np. *LendingAdvisor* w Canadian Imperial Bank of Commerce) oraz wydajności dotyczącej konkretnych zadań (szybkich transferów telegraficznych w Security Pacific Bank). Do podobnych wniosków doszedł Radosiński [246] opisując przyczyny powodzenia systemu *CLUES* zastosowanego w banku Countrywide. Mimo że korzysta się z niego wyłącznie do oceny **wniosek osób fizycznych** starających się o niewielkie pożyczki, to ze względu na wielokrotne powtarzanie procedur (wybranych z dużej puli) przy każdym z wielu wniosków **opłaca się automatyzacja wydawania ocen**. Z drugiej strony pracownicy kredytowi mogą w tym czasie zajmować się problemami do rozwiązania których **konieczna jest wiedza specjalistyczna**, szczególna uwaga, doświadczenie i intuicja. Jeśli chodzi o **skuteczność działań** to obserwowano pozytywne efekty w zakresie organizacyjnym (wspomniany *CitiExpert* zastosowany w automatycznym systemie obsługi klienta), indywidualnym (*Sanwa Bank Personal Portfolio* który umożliwił delegowanie uprawnień rozpatrywania wniosków kredytowych do oddziałów banku) wreszcie konkretnych zadań (np. zmniejszenie kosztów identyfikacji oszustw w *Barclaycard FraudWatch*). **Pozytywny wpływ na ekspertyzę**, przede wszystkim organizacyjną polega na **udostępnieniu wiedzy wszystkim** pracownikom którzy powinni wykrywać przypadki oszustw dotyczących transakcji kartowych (SE wspomagający zarządzanie kartami debetowymi w Security Pacific National Bank). Znotowano także **pozytywne oddziaływanie w procesie szkolenia pracowników**; na poziomie organizacyjnym (*CUBUS* w Swiss Bank) SE wykorzystywany jest w szkoleniu całego personelu kredytowego, na poziomie indywidualnym oddziaływanie SE jest skutkiem wykorzystywania w edukacji każdego pracownika, natomiast na poziomie poszczególnych zadań ES przydatny jest w kształceniu pracowników jakie procedury powinni stosować w celu wykrycia oszustw transakcji kartowych (Security Pacific National Bank). **Pozytywne oddziaływania w zakresie konkurencji** i szerzej – środowiska – polegają na (1) **obniżeniu poziomu ryzyka** biznesowego lub kredytowego (lepsze wykorzystanie posiadanych środków i mniejsza stopa wadliwych ocen wniosków np. systemy eksperckie *MOCCA* i *CUBUS*), (2) **zmiany wzorców zatrudnienia** (zwolnienia, przesunięcia personelu, przyjęcia do pracy), (3) **zmiany procesów obsługi** poszczególnych zadań: decentralizacja decyzji, rozproszone kontrolowanie zadań. Jak słusznie zauważa Shao [267] decyzja o wcieleniu systemu eksperckiego do strategii zarządzania firmą jest zwykle wymuszona konkurencją liderów rynkowych takich jak American Airlines czy Bank of America.

MCDA – *multiple criteria decision making* – **wielokryterialne podejmowanie decyzji** zajmuje się według definicji podanej przez Keeneya i Raiffę [145] słabo ustrukturalizowanymi i nieustrukturalizowanymi **problemami decyzyjnymi** opisanymi za pomocą wielu atrybutów oraz wielu celów. Podstawowe trudności rozwiązywania problemów w których określono więcej niż jedno kryterium optymalizacyjne wg Szapiro [293] wynikają z braku możliwości jednoznacznego podziału za pomocą funkcji celu wszystkich dopuszczalnych rozwiązań na rozłączne klasy, który pozwala na liniowe uporządkowanie poszczególnych dopuszczalnych rozwiązań. Dlatego też proponuje się postępowanie zgodnie z **zasadą optymalności w sensie Pareto**, polegającą na uznaniu rozwiązania dopuszczalnego za decyzję optymalną (efektywną) gdy dowolne zmiany jednej ze zmiennych powodujące poprawę wartości jednego z kryteriów (funkcji celu) powodują jednocześnie pogorszenie co najmniej jednego innego kryterium. Istnieją także inne techniki znajdowania rozwiązań problemów w których występuje funkcja wielokryterialna: jedną z nich jest tzw. osłabianie warunków polegające na dopuszczeniu rozwiązań niebędących optymalnymi co najmniej ze względu na jedno z kryteriów (np. ustalając graniczną różnicę czy stosunek procentowy między dopuszczalną wartością kryterium a wartością wykazywaną w przypadku optymalizacji tylko tego jednego kryterium (funkcji celu)).

Kolejne podejście nazywane **hierarchią celów** [293] jest uogólnieniem opisanej uprzednio **techniki optymalizacji docelowej** i polega na modyfikacji kryterium optymalizującego w taki sposób aby utworzyć **uporządkowanie** (hierarchię) **znaczenia celów** postaci np. wartości procentowych (normalizowanie). Podejście powyższe zostało wykorzystane w systemie eksperckim FINEVA [188] służącym do klasyfikacji ryzyka bankructwa przedsiębiorstwa. Znane są także [293] kryteria min-max mające na celu znalezienie rozwiązania które minimalizować (min-) będzie największe (-max) odchylenie od tzw. poziomów optymalnych, tzn. wartości poszczególnych funkcji celu. Można także stosować **zasadę substytucji**, tzn. równoważnika (mnożnika) jednostek poszczególnych funkcji celu.

Wydaje się że **decyzyjna analiza wielokryterialna** jest tym, czego wsparcia można oczekiwać od systemów wspomaganie decyzji (SWD), gdyż filozofia metodologii MCDA doskonale uzupełnia zadania i cele narzędzi SWD. Zarówno MCDA jak i SWD zostały zaprojektowane w celu rozwiązywania problemów opisanych w **kategoriach jakościowych i ilościowych**, charakteryzujących się **słabym ustrukturalizowaniem** i obecnością obiektów wieloatrybutowych. Proces podejmowania decyzji jest nierozłącznie związany z obraniem racjonalnych kryteriów (zmiana celów formułowanych przez decydenta może wymuszać zmianę kryteriów) oceniających wyniki organizacji oraz dokonujących **serii wyborów spośród wielu alternatywnych rozwiązań**. Powstają przez to nowe metody działania zapewniających uzyskiwanie poprawy wyników organizacji w przyszłości. MCDA może korzystać z bazy wiedzy zapisanej w postaci wielowartościowego rachunku zdań, pozwalającego na zapis więcej niż dwu wartości logicznych poszczególnych atrybutów składających się na opis obiektu (np. wariantu decyzyjnego). Reguły wnioskowania operujące na wieloatrybutowych obiektach przystosowane są do wartościowania wariantowych hipotez (rozwiązań).

3.7 Systemy wspomaganie decyzji

3.7.1 Systemy wspomaganie decyzji - rys historyczny

Wyróżnić można dwa źródła [270] z których wyłoniły się systemy wspomaganie decyzji. Pierwszym są **teoretyczne studia nad podejmowaniem decyzji** w organizacjach, prowadzone przez Simona, Cyerta i Marcha w Carnegie Institute of Technology (lata pięćdziesiąte XX wieku) a drugim **badania nad technologią komputerową** prowadzone w Massachusetts Institute of Technology w latach 60-tych przez m.in. Gerrity i Ness. W tych czasach SWD składały się z następujących modułów: (1) systemu zarządzania bazą danych wzbogaconą o możliwość magazynowania i przetwarzania wiedzy, (2) modułu zarządzania modelami (dziedzinowymi), (3) rozbudowanym (interaktywność, raportowanie, grafika biznesowa) interfejsem użytkownika. Lata siedemdziesiąte przyniosły zainteresowanie [270] **stroną technologiczną** przedsięwzięć dotyczących SWD i podwyższaniem wydajności (w znaczeniu liczby podejmowanych decyzji w określonym czasie) podejmowania decyzji i efektywności decyzji (mierzona efektami decyzji podejmowanych przy wsparciu przez SWD). Wydaje się że do **głównych innowacji technologicznych** ostatnich lat (po okresie mikrokomputeryzacji i rozwijania koncepcji graficznego interfejsu użytkownika) które mają znaczenie dla projektanta i użytkownika SWD zaliczyć można upowszechnienie **technik sieciowych, Internetu, mobilnej komunikacji** i wykorzystania **rozproszonych zasobów informacji i wiedzy**. Pojęcie **systemu wspomaganie decyzji** wydaje się niektórym dość przestarzałe, jednak pozwala na łączne rozważanie takich współczesnych (różnorodnych) technik wspomaganie rozwiązywania problemów jak OLAP, EIS, ESS, systemy eksperckie oparte na regułach i wiele innych. Wspólną cechą narzędzi wspomaganie decyzji (*DSS Tools*) jest ich użytkowanie w celu usprawnienia procesu podejmowania decyzji, ze szczególnym uwzględnieniem złożonych systemów, opisywanych w kategoriach niepewnej i niepewnej informacji.

3.7.2 Systemy wspomaganie decyzji – pojęcia i klasyfikacja

Planowanie decyzji strategicznych (tzn. finansowych, inwestycyjnych, dotyczących rozwoju, innowacyjności) i planowanie strategiczne mogą wspomagać **systemy informatyczne z bazą wiedzy**. Wśród nich wyróżniamy [230]: systemy wspomaganie decyzji, inteligentne systemy wspomaganie kierownictwa, systemy eksperckie, grupowe systemy wspomaganie decyzji.

Pojęcie „system wspomaganie decyzji”, SWD jest **znaczeniowo dość obszerne**. Wydaje się, że określeniem tym można obdarzyć oprogramowanie, które jest przeznaczone do bezpośredniej pomocy osobie w podejmowaniu decyzji dotyczących gospodarowania (np. menedżerowi, inwestorowi). Według Griffina [103] „SWD jest systemem, który automatycznie wyszukuje, przetwarza i symuluje informacje potrzebne menedżerom do podejmowania konkretnych decyzji”. Jeśli do **klasy SWD**

zaliczymy systemy eksperckie, GIS, OLAP, inteligentnych agentów, systemy odkrywania wiedzy, wspomagania decyzji grupowych a nawet systemy informowania kierownictwa z podkreśleniem, że SWD korzystają z efektów działań a nawet sterują systemami transakcyjnymi to można stwierdzić że SWD jest systemem informatycznym o stosunkowo dużym stopniu złożoności. SWD powinny posiadać odpowiednie możliwości konfigurowania, automatycznego lub ręcznego wyszukiwania, przetwarzania i prezentacji informacji które menedżerowie mogą wykorzystać w procesie podejmowania decyzji.

Umiejscowienie SWD pośród narzędzi informatycznych znajdujących zastosowanie w zarządzaniu organizacjami przedstawiono w tabeli [Tab. 3-1]. Należy zauważyć, że złożoność narzędzi rośnie wraz z wydłużaniem horyzontu decyzyjnego i zwiększeniem wymaganego udziału ludzkiego doświadczenia (czynnika ludzkiego) w podejmowaniu decyzji czyli (w tabeli) ukośnie w dół z lewej strony do prawej. Jednocześnie pamiętajmy o warstwowej architekturze narzędzi informatycznych co przejawia się korzystaniem z funkcji narzędzi o większym stopniu zautomatyzowania przez narzędzia o większym stopniu złożoności.

Tab. 3-1 Wielowymiarowa macierz zastosowań informatycznych narzędzi wspomagających proces podejmowania decyzji w zależności od stopnia ustrukturalizowania decyzji i horyzontu zarządzania. Źródło: [303]

Typ decyzji	Typ kontroli			Potrzeba wsparcia
	Sterowanie operacjami	Zarządzanie (taktyczne)	Planowanie strategiczne	
Ustrukturalizowane	1. Należności, uruchomienie zamówienia	2. Analiza budżetowania, krótkoterminowe prognozy, zestawienia o personel, analiza zakupów do wykonania	3. Zarządzanie finansowe, lokalizacja magazynów, systemy dystrybucji	SIZ
Częściowo ustrukturalizowane	4. Szeregowanie operacji, sterowanie zapasami	5. Ocena kredytowania, przygotowanie budżetu, planowanie projektów, opracowanie systemu płac	6. Wznoszenie budynków, fuzje i przejęcia, planowanie nowych produktów, zapewnienie jakości	SWD
Nieustruktural.	7. Wybór opakowania, zakup oprogramowania, zatwierdzenie pożyczki	8. Negocjacje, przyjmowanie do pracy, zakup wyposażenia, lobbowanie	9. Badania i rozwój, rozwój technologiczny, planowanie i zarządzanie personelem	SWD, SE, sieci neuronowe
Potrzeba wsparcia	SIZ	SIZ, SWD, SIK, SE	SIK, SE, sieci neuronowe	

Keen i Scott Morton [144] określają **system wspomagania decyzji (SWD)** jako system komputerowy wspomagający proces podejmowania decyzji przez menedżerów w słabo ustrukturalizowanych zadaniach. Zwracają uwagę, że chodzi tu **wspomaganie** a nie zastąpienie decydenta oraz bardziej o **poprawę skuteczności** (mniejsza stopa błędnych decyzji) procesu decyzyjnego niż jego **wydajności** (prędkości podejmowania decyzji, liczby podejmowanych decyzji w jednostce czasu, śledzenie i wspomaganie jednocześnie wielu procesów decyzyjnych). Mówiąc dokładniej, zadaniem **systemu wspomagania decyzji** jest wsparcie działań prowadzących do przygotowania i przedstawienia alternatywnych scenariuszy i projekcji przyszłości (projektowanie, implementacja i ewaluacja), obejmujących przyszłą strukturę, stan i otoczenie obiektu gospodarczego. Shim i inni [270] definiują SWD jako „rozwiązania oparte na technologii komputerowej, które mogą być wykorzystane jako wsparcie dla podejmowania złożonych decyzji oraz rozwiązywania

problemów”. Wyróżniają trzy etapy ukierunkowanego na modele wspomaganie decyzji: formułowanie modelu, poszukiwanie rozwiązania, analizę rozwiązań. **Formułowanie modelu** (powinno odbywać się z wykorzystaniem narzędziowego środowiska modelowania) obejmuje **transformację** (często nieformalnego) **słownego** (lub mentalnego) opisu **problemu decyzyjnego** na język modeli (matematycznych, nawet jeśli modelowanie odbywa się za pomocą środowiska GUI zawierającego **szereg gotowych ikon** – klocków z których konstruuje się model i definiuje jego strukturę). Poszukiwanie rozwiązania, w zależności od **typu problemu decyzyjnego**, odbywać może się z wykorzystaniem **metod optymalizacji** (sympleks itp.), **drzew decyzyjnych** (modelowanie niepewności), **heurystyk**, **sztucznej inteligencji**: sieci neuronowych, symulowanego wyżarzania itd. Analiza rozwiązań obejmować może **analizę wrażliwości**, techniki dedukcyjne, **rozpatrywanie scenariuszy symulacyjnych** „co się stanie jeśli”, **wyszukiwanie związków** w zestawach wygenerowanych rozwiązań (metody indukcyjne).

Bardziej szczegółowy opis wyróżniający **4 etapy podejmowania decyzji** z wykorzystaniem metod **ekspertyzy, optymalizacji i konfrontacji** z systemem rzeczywistym przedstawił Simon [275]. Wyróżnił on **etap wstępny**, który związany jest przeprowadzeniem działań „wywiadowczych” czyli rozpoznaniem istoty problemu. Na tym etapie do głosu dochodzi **ludzka inteligencja** [215], na którą składają się procesy poznawcze, takie jak uwaga, pamięć operacyjna i kontrola poznawcza. Etap na którym korzysta się z ludzkiej inteligencji składa się z identyfikacji problemu, zbierania danych i sformułowania problemu. Zaznaczyć należy że etap ten realizowany jest **wyłącznie na podstawie ludzkiego doświadczenia** i polegający na wiedzy eksperckiej (ludzkiej). Na kolejnym (trzecim) etapie czyli projektowania, odbywa się **formułowanie modeli, wybór** modeli do dalszych rozważań oraz ich **walidacja**. Podczas **etapu projektowania** można już zastosować jako wspomagające odpowiednie techniki sztucznej inteligencji: wnioskowanie na podstawie przypadków (CBR) tzn. dokumentacji opracowanych dotychczas modeli, sieci neuronowe oraz struktury ram podczas wyboru modeli. W największym stopniu **informatyzacji** daje się poddać **etap wyboru rozwiązań** (określenia wartości zmiennych decyzyjnych) na którym ma miejsce poszukiwanie rozwiązania modelu (optymalizacja, techniki sztucznej inteligencji) testowanie i ocena rozwiązań (np. symulacyjne, przez analizę wrażliwości) oraz integrowanie (likwidowanie niespójności) częściowych rozwiązań z założeniami decydenta (korzystanie z systemów eksperckich, w szczególności o architekturze tablicowej). Jako ostatni w kolejności jest wykonywany **etap adaptacji rozwiązań** do zastosowań w systemie rzeczywistym. Oczywiście w przypadku braku powodzenia następuje nawrót do odpowiednio wcześniejszego etapu podejmowania decyzji. Etapy pracy **systemu wspomaganie decyzji** są zgodne z etapami pracy **systemu przetwarzania danych**:

- (1) Zebranie danych i ich zestawienie
- (2) Zapisywanie i dostęp danych oraz wyszukiwanie na podstawie treści
- (3) Analiza danych

(4) Sporządzanie raportów tworzonych na podstawie wyników analiz.

Ariav i Ginzberg [12] wyróżniają pięć następujących aspektów systemowych SWD: środowisko, rolę, komponenty, układ i zasoby. **Definiowanie środowiska** SWD dotyczy elementów leżących na granicy systemu, będących w bezpośredniej interakcji ze światem zewnętrznym – charakterystyki użytkownika i trybów dialogu z systemem, opisu powiązań (relacji) z innymi systemami (pakietami oprogramowania) ale także cech zadań których projektowanie ma wspierać SWD (strategiczne, taktyczne, operacyjne), wspieranego etapu procesu podejmowania decyzji. **Rola** SWD definiowana jest przez **poziom oferowanego wsparcia**: np. prezentacja zestawu rozwiązań lub pomoc w wyborze rozwiązania (wariantu decyzji) oraz wspierania poszczególnych etapów, zadań, czynności bądź kontrolowania zgodności z ogólnym planem. **Wyodrębnione komponenty** funkcjonalne SWD wg Laskey [159] to centralny moduł zarządzania i tworzenia wiedzy (*knowledge engine*), korzystający z usług modułu zarządzania danymi (pochodzącymi z pasywnych, zewnętrznych źródeł), modułu zarządzania dialogiem (z użytkownikami i aktywnym środowiskiem) oraz modułu zarządzania modelami (składowanymi w wewnętrznej bibliotece modeli).

Biorąc pod uwagę **etapy przetwarzania wiedzy** w SWD wyodrębnić można następujące moduły-komponenty:

- a) Realizujące rejestrację i przechowywanie wiedzy pozyskanej z różnych źródeł
- b) Obsługujące udzielanie odpowiedzi na podstawie zgromadzonej wiedzy na nieustannie składane zapytania
- c) Udostępniające intuicyjny interfejs użytkownika dzięki któremu wydobywane i prezentowane są wymagane informacje

Turban [303] wyróżnia następujące komponenty strukturalne standardowego SWD:

- a) Komponent zarządzania danymi – składa się z baz danych zawierających dane zebrane pod kątem ich zastosowania w rozwiązaniu problemu decyzyjnego
- b) Komponent interfejsu użytkownika – umożliwia komunikację i zarządzanie SWD przez użytkownika
- c) Komponent zarządzania modelem – składa się z wielu rodzajów oprogramowania służącego zastosowaniu modeli ilościowych: finansowych, statystycznych, z obszaru nauki o zarządzaniu
- d) Komponent zarządzania wiedzą – obsługuje wszystkie inne podsystemy lub działa jako niezależny komponent.

Istnieją dwa typy **układu** komponentów SWD: **dostosowany** do konkretnego problemu oraz zdolny do **adaptowania się** na podstawie informacji pochodzących ze środowiska. **Zasoby SWD** dzielą się na: **ludzkie** (osoby biorące udział w projekcie – inżynierowie wiedzy, eksperci, nadzorujący projekt oraz użytkownicy), sprzęt, **oprogramowanie** oraz **dane** (pochodzące ze środowiska i po przetworzeniu wysyłane na zewnątrz, będące pochodną pracy modułu zarządzania i tworzenia wiedzy).

3.7.3 Systemy wspomaganie decyzji – elementy teorii

Zazwyczaj system wspomaganie decyzji zawiera następujące moduły funkcjonalne: model problemowej sytuacji biznesowej, bazę danych, biblioteki metod obliczeniowych, analitycznych i graficznych i interfejs użytkownika. Właśnie modularna struktura SWD umożliwia rozszerzanie oferowanych przez niego usług o kolejno dołączane moduły metod i technologii sztucznej inteligencji.

Wśród referatów naukowych dotyczących SWD wg badań Eiermana i innych [72] wyróżnić można artykuły traktujące o systemach podejmowania decyzji grupowych (*GDSS*), systemów informacyjnych kierownictwa (*EIS*) oraz systemach eksperckich (eksperskich) a główne **elementy skupiające zainteresowanie badaczy** to: środowisko, zadania, strategie implementacji, możliwości SWD, konfiguracje SWD, użytkownik, zachowanie użytkownika, wydajność.

Środowisko definiowane jest przez **organizacje** zaangażowane w tworzenie SWD, mające wpływ na niego. **Na środowisko oddziałują następujące czynniki** (zmienne): kultura organizacyjna, cele organizacji, natężenie i charakter zmian z którymi musi liczyć się kierownictwo (spektrum od środowiska stabilnego do podlegającego szybko zmieniającym się trendom), dostęp do zasobów przedsiębiorstwa (wynikający z decyzji kierownictwa), poziom dotychczasowego wykorzystania technologii informacyjnych w przedsiębiorstwie, struktura organizacyjna, poziom wsparcia dla projektu udzielanego przez naczelne kierownictwo.

Zadanie – kolejny element teorii SWD – to działanie prowadzące do realizacji jednej z **złożonych funkcji organizacji** (ale może być ona osiągnięta np. po serii operacji; konfiguracja zadań zależy znacząco od decydenta). Każde zadanie jest określane przez następujące zmienne: złożoność, stopień ustrukturalizowania, fazę decyzyjną, stopniem ważności decyzji której dotyczy, etapem badań decyzyjnych. Etap badań decyzyjnych rozumiany jest w sensie podanym Simona [275] czyli jest to sformułowanie problemu, opracowanie wariantów decyzyjnych, ocena wariantów.

Strategia implementacji – to zestaw (niekoniecznie skodyfikowanych) **reguł i działań prowadzących do powstania SWD** i połączenia za pomocą interfejsu z istniejącą **infrastrukturą informatyczną** oraz oddziaływania na jej funkcjonowanie (korzystanie z zasobów, przewartościowanie metod pracy personelu przedsiębiorstwa). Zasadniczo pokrywa się z cyklem życia systemu informatycznego, jednak większy nacisk należy tu położyć na szkolenie użytkowników, lepsze wykorzystanie czasu sesji użytkownika z programem, sporządzenie i udostępnienie dokumentacji oraz starania o udzielenie wsparcia przez naczelne kierownictwo. Przyczyną większego znaczenia komunikacji jest wielość funkcji które może spełniać wdrożone oprogramowanie.

Wspomaganie użytkowników lub grup użytkowników SWD w wypełnianiu zadań zdeterminowane jest **możliwościami SWD**. Spektrum możliwości rozpoczyna się np. od wyszukiwania potrzebnych rekordów w bazie danych przez generowanie wariantów decyzyjnych do wsparcia zadania tworzenia matematycznego modelu zjawisk i procesów gospodarczych. **Użytkowe**

możliwości SWD określić można za pomocą **szeregu zmiennych** (własności): rodzaju interfejsu (interaktywny - graficzny bądź wsadowy), liczbie, zakresu i elastyczności wbudowanych funkcji, łatwości konstruowania własnych funkcji przez użytkownika (rozbudowa systemu), szybkości działania, trybu oddziaływania na procesy w przedsiębiorstwie (system alarmowy, monitorowanie stanu procesu, wykonywanie okresowych sprawozdań, uruchamianie na żądanie), postaci raportów (grafika, tabele, tekst), rodzaju wspieranych działań (analiza finansowa, zarządzanie zasobami, monitorowanie procesu produkcyjnego).

Kolejnym elementem teorii SWD jest jego **konfiguracja** – dzięki niej udostępniane są wymienione uprzednio możliwości SWD. **Konfiguracja SWD** opisuje jego **dwa wymiary technologiczne**: oprogramowanie i sprzęt. Programiści tworzący SWD mają do wyboru wszystkie standardowe **techniki software’owe** tworzenia systemów inteligentnych: bazę danych, bazę wiedzy, programy – demony, modele użytkownika i jego wymagań, procesor komend (realizowany np. w postaci menu) i języka naturalnego. **Wymiar sprzętowy** dotyczy wyposażenia w sprzęt komputerowy, wielodostępu, usieciowienia, **integracji** z istniejącym systemem informatycznym, finansowym i produkcyjnym.

Użytkownik jest elementem teorii SWD będącym w relacji z zadaniami. Wykorzystuje wyniki pracy SWD do realizacji zadań poprzez swoje **zachowania** (działania podejmowane przez użytkowników dla realizacji zadań). Każdy konkretny użytkownik (podmiot decyzyjny) ma charakterystyczne własności ale największe znaczenie posiada liczba podmiotów. Wyróżnić dlatego można osoby wykorzystujące SWD w celu podejmowania decyzji których są **wyłącznymi dysponentami** oraz **zespoły ludzkie** które podejmują decyzje wg opracowanych reguł uwzględniając wyniki analiz prowadzonych z wykorzystaniem SWD. W przypadku **grupowego podejmowania decyzji** istotne są następujące zmienne: wielkość grupy, bliskość poszczególnych członków, stopień formalnych powiązań, równouprawnienie, kolejność wyrażania opinii, niezależność między członkami. Inne **ważne własności** podmiotów decyzyjnych to: doświadczenie **dotyczące zadania**, rozwiązanie którego ma być wsparte przez SWD, doświadczenie **dotyczące technologii informatycznych**, nastawienie oraz oczekiwania **wyrażane względem korzystania** z SWD. Znaczenie także mają **zdolności analityczne** (znajomość modeli matematycznych prowadzących do rozwiązań problemów, umiejętność systemowego postrzegania zjawisk oraz budowania modeli mentalnych), **zdolności kierownicze** (wdrażanie wybranych wariantów decyzyjnych) oraz **fachowe** (dotyczące ekspertyzy dziedzinowej).

Zachowanie użytkownika (czyli sposób w jaki realizuje zadania; w jakim celu, w jaki sposób wykorzystuje SWD) stanowi pomost między możliwościami SWD (pojmowanymi w sposób obiektywny jako specyfikacja systemu) oraz wynikami działania. Zachowanie użytkownika określane jest przez: (1) **wykorzystanie umiejętności** postrzegania i rozważania poszczególnych aspektów mających wpływ na podejmowaną decyzję (czyli tworzenia i wykorzystania modelu sytuacji

problemowej), (2) rodzaj i ilość wykorzystywanych **informacji** i ich interpretację, (3) preferencje **dotyczące oprogramowania** np. marka, zgodność z dotychczas stosowanym oprogramowaniem (4) **intensywność korzystania** z SWD, zaangażowanie w rozwój, skłonność do działania zgodnie z zaleceniami SWD.

Wyniki działania są traktowane jako **ostateczny weryfikator skuteczności** wdrożonego SWD. Warto wymienić następujące **miary oceny wyników działania** SWD: (1) skuteczność/jakość – służą porównaniu skutków decyzji podejmowanych bez i z wykorzystaniem danego SWD, (2) miernik dotyczący aspektów psychologicznych zastosowania SWD: zaufanie do wyników analiz wydawanych przez SWD oraz satysfakcja z jego stosowania, (3) mierniki ekonomiczne (finansowe) tzn. wzrost zysku, wzrost przychodów, spadek wydatków po wdrożeniu SWD, (4) dynamikę wydajności (tzn. skrócenie czasu opracowania decyzji po wdrożeniu SWD).

3.7.4 Wsparcie SWD przez technologie przetwarzania informacji

Systemy wspomaganie decyzji korzystać mogą z szeregu nowych technik przetwarzania danych i informacji: hurtowni danych, OLAP, *data mining*.

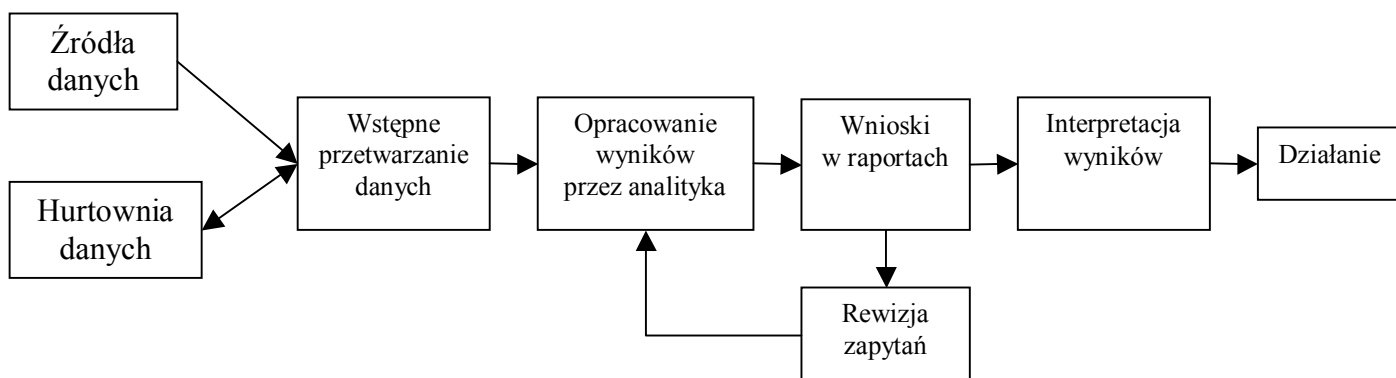
Hurtownie danych wg Inmona [126] są „tematycznie zorientowanymi, spójnymi, chronologicznymi i niezmiennymi kolekcjami danych tworzonymi dla wspomaganie procesów podejmowania decyzji kierowniczych”. Tematyczność związana jest z celem tworzenia hurtowni danych czyli zapoznania się z określonym aspektem zagadnienia takim jak sprzedaż w przekroju asortymentowym i terytorialnym czy identyfikacja jej trendu. Hurtownia danych tworzona w celu rozwiązania danego problemu decyzyjnego zawiera jedynie dane które będą przedmiotem analizy, korzysta zatem z **filtrowania danych**, dzięki któremu nie ma potrzeby przesyłania i przechowywania danych w miejscu z którego wymagany jest do nich dostęp), **zintegrowanymi** (pozwalają na korzystanie z rozproszonych **różnorodnych** zasobów danych i informacji w sposób ujednolicony (stosuje się przekodowanie na ustalony format zapisu) dostępnych w **konkretnym miejscu** i bez potrzeby korzystania z oddzielnych pakietów dostępu i przetwarzania danych), **podlegającymi zmianom** (rejestrującymi wyniki zachodzących procesów; hurtownia danych „rozsztala się” stopniowo w czasie, gdyż jej jądro (tabela faktów, zawierająca opis elementarnych danych, np. sprzedaży) jest stowarzyszone z tabelą wymiarów, rejestrującą wielkie ilości elementarnych danych, tworzących przekroje czasowe, przedmiotowe, terytorialne itd.) i cechującymi się **nieulotnością** (tzn. że można otrzymać informacje o zjawiskach zachodzących w przeszłości, nazywane jest to oznaczaniem czasowym i wykorzystywane do obliczeń parametrów trendów zmian) bazami danych.

Rynkiewicz [255] podkreśla względną łatwość tworzenia i zalety funkcjonalne tzw. *data marts*, czyli **tematycznych hurtowni danych**. W tematycznych hurtowniach danych możliwa jest

optymalizacja układu danych a następnie ich **agregacja** co powinno skutkować zwiększeniem szybkości uzyskiwania coraz bardziej szczegółowych odpowiedzi na pytania. Analizy controllingowe sprzedaży, przykładowo, dotyczyć będą sprzedaży ilościowej i wartościowej poszczególnych produktów i ich partii, przekrojów terytorialnych, porównania wyników pracy sprzedawców. Według Głowińskiego [97] wdrożenie hurtowni danych daje szansę na **realizację trzech postulatów** dotyczących źródeł danych wykorzystywanych przez decydentów podejmujących decyzje strategiczne: **rzetelność informacji**, **szybkość dostępu do wielkiej ilości danych** oraz **prezentacja przetworzonych danych** w postaci przekrojów (perspektyw). Wyróżnikiem technologii hurtowni danych jest zatem ograniczenie poleceń do dwóch procedur: **ładowania hurtowni** i **uzyskania dostępu do danych** (odczytu). Natomiast każda zmiana zawartości jest **rejestrwana** (tworzony jest znacznik czasu) i dzięki temu można określić czas i przyczynę każdego zjawiska które wpłynęło na „załadowanie” hurtowni danych nową zawartością.

Korzyści z utworzenia i eksploatacji hurtowni danych polegają na umożliwieniu użytkownikowi **dopasowania modeli i perspektyw** służących opanowaniu wewnętrznej złożoności wielkich ilości danych. Hurtownie danych stanowią rozwinięcie relacyjnego modelu baz danych poprzez połączenie wielu źródeł danych, aby można było dokonywać manipulacji nimi za pomocą interfejsu graficznego (operator) oraz odwoływać się do nich przez moduł bazodanowy (procedury inicjowane przez system wspomaganie decyzji).

Dostępność **hurtowni danych** jest warunkiem uruchomienia procesów typu *data mining* [113]. Pozostałe elementy procesu to źródła danych, wstępne przetwarzanie (przez nie hurtownia danych komunikuje się ze źródłami danych i procedurami wyszukiwania wzorców), wyszukiwanie wzorców, prezentacja wyników analitykom (ewentualnie rewizja ich zapytań), wnioski zawarte w raportach (przygotowanych przez człowieka), interpretacja wyników i rozpoczęcie działań w oparciu na wynikach.



Rys. 3-2 .Elementy procesu data mining. Źródło: [113]

OLAP (*On-line analytical processing*) jest technologią dostępu do hurtowni danych, która umożliwia analizę danych zapisywanych w trakcie jej działania (dane historyczne) polegającą na przemieszczaniu się między elementami poprzez [224] “szybki, spójny, interaktywny dostęp do

szerokiego zakresu perspektyw informacji”. Zestawienia, projekcje prezentowane użytkownikowi realizowane są dzięki przetwarzaniu „surowych” danych.

OLAP nazywany też *Fast Analysis of Shared Multidimensional Information - FASMI* oznacza technologię dostarczającą usług polegających na opracowaniu w **czasie rzeczywistym** (lub wystarczająco krótkim) **zapytań** i **zestawień** (tabelarycznych i graficznych - wykresów) na podstawie **wielowymiarowych** (najczęściej także wielodostępnych) **baz danych**. Wytyczne (w postaci 12 reguł) dotyczące technologii wielowymiarowej analizy danych (*OLAP*) opracował Codd a ich rozszerzenie (18 funkcji) przedstawił Thomsen [296]. Sierocki w pracy [272] przedstawił **postulaty** dotyczące systemów zrealizowanych w **technologii OLAP**: (1) przystosowanie do **analiz wielowymiarowych** (optymalizowane wykonywanie zapytań do baz wielowymiarowych), (2) **intuicyjność** modelu analitycznego (przystosowany do dynamicznych zmian poziomu analizy, np. drill down), (3) korzystanie z centralizowanego modelu przechowywania danych - **hurtowni danych** (*data warehouse*) i **tematycznych hurtowni danych** (*data mart*; zorientowanej na rozwiązywanie określonych problemów), (4) możliwość wykonywania analiz przez **wielu użytkowników równocześnie**, (5) **zautomatyzowanie** przetwarzania danych (dotycząca raportów, tabel typu *pivot*), (6) **graficzny interfejs** użytkownika, (7) możliwość podłączenia i **integracji z aplikacjami bazodanowymi** (interfejsy programowe, niezależność funkcji użytkowych od aplikacji).

Zwykle korzystanie z *OLAP* polega na **manipulowaniu, konfigurowaniu** z góry przygotowanymi złożonymi **procedurami przetwarzania danych** (w tym np. filtrowanie zakłóceń, oczyszczanie danych) zawartych w biznesowych bazach danych (źródłowych). Przetwarzanie danych w **technologii OLAP** można traktować jako **uzupełnienie** (ze względu na odmienne cele) bądź jako **wstępny** (pomocniczy) etap *data mining*. Systemy *OLAP* występują w dwu odmianach: **wielowymiarowych baz danych (MDDB)** oraz **relacyjnych OLAP**. **Pierwsze rozwiązanie** polega na konstruowaniu **hipersześcianu** przechowującego przetworzone, zagregowane informacje z baz danych z których użytkownik korzysta przez wyświetlanie poszczególnych warstw i wzajemne ich porównywanie. Simon [277] podaje dwa ograniczenia którym podlegają systemy *OLAP*: brak standardowych metod dostępu do *MDDB* – konieczność polegania na rozwiązaniach firmowych oraz występowanie etapu konwersji danych zawartych w relacyjnych bazach danych do hipersześcianu. Drugi typ *OLAP*, określanany jako **relacyjny OLAP** oparty jest na korzystaniu z dodatkowej warstwy oprogramowania, przetwarzającej polecenia wydawane poprzez interfejs *OLAP* bezpośrednio (tzn. dla użytkownika niewidoczne są operacje składowania w hurtowni danych, agregacji wielowymiarowej i składowania w hipersześcianach bazy danych) do postaci instrukcji dostępu do baz danych (*SQL*), problemem jest jednak niezawodne ich przetłumaczenie (wielowymiarowe bazy danych wykorzystywane są przez interfejs użytkownika bezpośrednio). Dla potrzeb *data mining* czyli konstruowania modeli predykcyjnych (wielowariantowej prognozy) można wykorzystać następujące techniki z zakresu sztucznej inteligencji: sztuczne sieci neuronowe, drzewa decyzyjne, reguły

decyzyjne, metody typu najbliższy sąsiad itd. [80]. **Obfitość dostępnych metod** wynika ze zróżnicowania charakterystyk dostępnych zbiorów danych (typu powiązań między danymi, stopnia złożoności, występowania zakłóceń). Każdy algorytm opracowuje się dla różnych warunków, które trudno jest zidentyfikować a priori. Nawet sieci neuronowe są bardzo wrażliwe na zestaw danych: zbyt obszerny zbiór danych skutkuje dłuższym i trudniejszym czasem uczenia się, gdyż algorytm musi uwzględniać informacje nieprzydatne w dyskryminacji wykrywanych pojęć.

3.7.5 Przykłady doniesień literaturowych o SWD w zakresie zarządzania

Wiele firm software'owych dostarczających narzędzia budowania hurtowni danych wzbogaca je o wsparcie technologii OLAP i *data mining*. Wśród nich wymienić należy firmę Oracle proponującą Warehouse Builder wraz z Data Mining Suite, IBM oraz jej Visual Warehouse z DB2 Warehouse Manager oraz DB2 OLAP Server (technologia OLAP) i IntelligentMiner (technologia *data mining*), Sybase z Warehouse Studio, firmę SAS - SAS/Warehouse uzupełnioną o SAS/Enterprise Miner. Wśród doniesień o zastosowaniu SWD w zakresie zarządzania wymienić należy:

WinQSB [39] jest pakietem wspomagającym szkolenie w zakresie rozwiązywanie problemów biznesowych a w szczególności zarządzaniu operacjami. Składa się z zestawu narzędzi przystosowanych do rozwiązywania różnorodnych problemów z którymi spotkać może się menedżer. Przykładowo, narzędzie służące zastosowaniu metod optymalizacji pozwala na definiowanie modelu matematycznego, określenie funkcji celu, ograniczeń oraz wykonanie analizy wrażliwości. Analiza decyzyjna odbywa się z wykorzystaniem **eksperymentów symulacyjnych**. Obejmuje ona analizę typu *what-if* dotyczącą macierzy wypłat oraz założeń o subiektywnych wartościach prawdopodobieństwa poszczególnych stanów natury [51]. Zastosowano reprezentację problemu w postaci drzewa decyzyjnego oraz udostępniono procedury analizy bayesowskiej (użytkownik wprowadza założenia o **prawdopodobieństwach a priori** oraz **prawdopodobieństwach warunkowych** i otrzymuje **prawdopodobieństwa a posteriori**).

Folio [45], klasyczny już przykład oprogramowania wspierającego zarządzanie portfelem papierów wartościowych. Działa on w trybie dialogu, tzn. system przeprowadza wywiad z użytkownikiem (klientem), określając jego preferencje. Wynikiem działania systemu są propozycje zaangażowania środków w określony typ funduszy w zależności od wysokości ryzyka akceptowanego przez klienta: mogą to być akcje niewielkiego ryzyka, wysokiego ryzyka lub ukierunkowane na zyski z wypłacanych dywidend.

System wspomagania decyzji używany w Security Pacific/Bank of America ma na celu pomoc podczas podejmowania decyzji dotyczących udzielania kredytów przedsiębiorstwom oraz zabezpieczenie przed oszustwami. Pakiet **@RISK** firmy Palisade [228] znajduje zastosowanie w

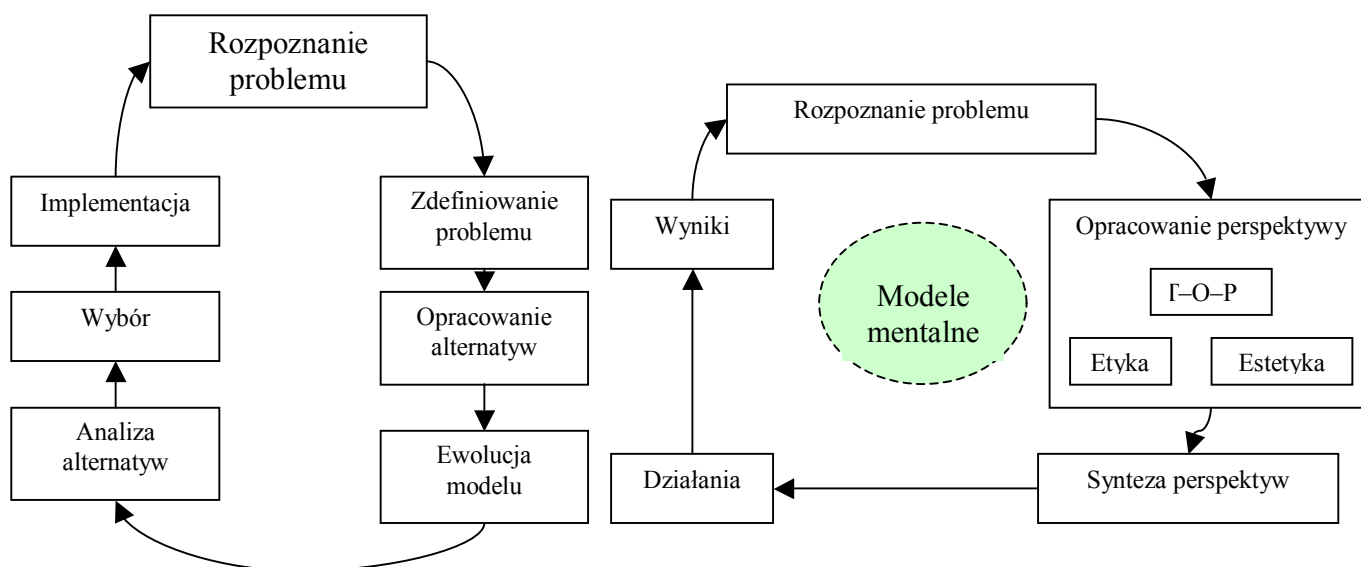
analizie ryzyka. Wykorzystuje w tym celu **symulacje typu Monte Carlo**, narzędzie dopasowywania rozkładów prawdopodobieństw BestFit oraz TopRank wspierające **analizę wrażliwości**. Oprogramowanie współdziała z arkuszami kalkulacyjnymi. Wśród użytkowników @RISK są firmy takie jak Merrill Lynch, Zimmerman Investment Co. oraz Global Emerging Market Advisors [64].

Davidson [53] donosi o wielu zastosowaniach komputerowych systemów zarządzania ryzykiem. Pakiet Enterprise Risk firmy Summit Systems (www.summithq.com/index.htm) w banku WestLB wspomaga zarządzanie ryzykiem związanym z instrumentami pochodnymi. Pakiet zarządzania ryzykiem rynkowym i kredytowym Front Arena firmy Front Capital Systems (www.front.com) znajduje zastosowanie w zarządzaniu ryzykiem dotyczącym depozytów składanych w tymże banku a także w Nordea Securities i TD Securities. Pakiet FleX firmy Murex (www.murex.com) wspiera analizy typu *value-at-risk* i monitoruje wielkość ryzyka w banku Bank One. Pakiety Firm Risk i Firmament Credit firmy Lombard Risk Systems (www.lombardrisk.com) wspomagają zarządzanie ryzykiem kredytowym i *value-at-risk* oraz operacji typu *swap* (poprzez moduł *ValuSpread*), poza tym umożliwiają integrację z wykorzystywanymi przez użytkownika (np. Erste Europäische Pfandbrief- und Kommunalkreditbank Aktiengesellschaft z Luksemburga oraz BoE Bank Group Treasury z RPA) danymi i algorytmami. Oprogramowanie NX CR Engine firmy NumericX (www.numericx.com) służy wycenie instrumentów pochodnych (w tym zarządzaniu koszykiem papierów dłużnych, obligacjami egzotycznymi) oraz ocenie ryzyka na podstawie analizy wrażliwości, analizy scenariuszowej i kontroli strategii zabezpieczających transakcje terminowe (*hedging*). Narzędzia firmy NumericX znalazły zastosowanie w Natexis Banques Populaires we Francji. Od roku 2003 jest dostępny na rynku oprogramowania pakiet MorganRisk, stworzony pierwotnie dla wewnętrznych potrzeb J.P.Morgan. W pakiecie [31] wykorzystano **dwa modele (moduły) zarządzania ryzykiem**: *value-at-risk* (analiza wariancji-kowariancji: umożliwiająca na przewidywanie wartości portfela na podstawie korelacji cen poszczególnych rodzajów walorów, symulacje oparte na dużej liczbie danych historycznych, z wielu lat oraz symulacje Monte Carlo) oraz tzw. *stress testing* (wspomagający zarządzanie w sytuacji znacznej niepewności i katastrof rynkowych, np. ataków terrorystycznych z 11 września 2001). CreditManager, jedno z narzędzi firmy RiskMetrics Group wspiera technikę *value-at-risk* stosując metodologię *CreditMetrics* [128], opracowaną w J.P. Morgan. Narzędzia RiskMetrics Group są wykorzystywane w The Bear Stearns Companies Inc. oraz J.P. Morgan Fleming Asset Management.

Wiele innych narzędzi **zarządzania ryzykiem finansowym** opisano w przewodniku Bobsguide [22].

3.7.6 Zadania modułu symulacyjnego SWD

System wspomagania decyzji korzystać może z **modelu symulacyjnego systemu finansowego** badanego obiektu (realizując w ten sposób analizę „co się stanie, jeśli”), co zgodne jest z ustalonym przez Sprague i Carlsona [280] wzorcem **DDM** (dialog, dane i model) dotyczącym **architektury SWD**. Symulacje są tym, co uruchamia **proces samodzielnego adaptowania** (aktualizacji i odnawiania) zawartości bazy wiedzy do środowiska, którego model zastosowano w symulatorze. Prognozowanie wykorzystujące symulację umożliwia **przesunięcie tradycyjnego paradygmatu** pętli podejmowania decyzji ze wsparciem ze strony SWD [270] obejmującego kolejne działania (rozpoznanie problemu -> definiowanie problemu -> formułowanie alternatyw -> opracowanie modelu -> analiza wariantów -> wybór -> wdrożenie -> nawrót) w kierunku **nowego, sformułowanego** przez Courtneya [Rys. 3-3] obejmującego na **nowo określone działania**: (rozpoznanie problemu -> tworzenie perspektyw (organizacyjnej, ludzkiej i technologicznej, z wymiarem etycznym i estetycznym) -> synteza perspektyw, operacje, wyniki) na które zasadniczy wpływ mają modele matematyczne wynikające z przekonań wyrażanych za pomocą modeli mentalnych.



Rys. 3-3 Wsparcie procesu podejmowania decyzji przez SWD: podejście konwencjonalne i nowoczesne. Źródło: [49]

3.7.7 Metodologia badań i klasyfikacja SWD

Eom [75] wśród obszarów badań dotyczących SWD wyróżnia: grupowe SWD, metody implementacji, zarządzanie modelami, projektowanie oraz trzy dyscypliny wpływające na rozwój SWD: kognitywistykę, naukę o organizacji oraz metodologię podejmowania wielokryterialnych decyzji.

Badania nad metodami implementacji dotyczą poznania czynników (*critical success factors*) wpływających na udane działanie SWD, konstrukcji mierników powodzenia SWD oraz identyfikacji

zależności między atrybutami metodologii SWD (*data-driven* czyli takie uczenie maszynowe dla którego atomem jest rekord danych, *process-driven* – istotą jest proces, *decision-driven* dla których atrybuty zmieniają się wraz z podejmowanymi decyzjami), ich twórców, użytkowników i środowiska a efektywnością SWD. Za najistotniejsze czynniki mające wpływ na powodzenie SWD Igbaria i Guimaraes [123] uznali **stopień przyjazności względem użytkownika** (łatwość obsługi, system pomocy i zdolność przystosowania do wymagań konkretnego użytkownika) jaki cechuje dany SWD, **stopień zaangażowania użytkownika** (psychologiczne nastawienie do projektu, zadań, wdrażania, szkolenia, użytkownika SWD w danej organizacji oraz chęć **udostępniania** własnych doświadczeń zawodowych w celu wzbogacania bazy wiedzy zawartej w SWD) w projekt SWD. Także Tyran i George [305] wymieniają jako kluczowe czynniki wpływające na powodzenie SWD (obok wkładu eksperta, współpracy przyszłego użytkownika oraz wsparcia ze strony naczelnego kierownictwa) właśnie prawidłowe **zidentyfikowanie potrzeb** przyszłego użytkownika. Zadanie o podobnej lub nawet większej złożoności – implementacja **narzędzi groupware** w przedsięwzięciach BPR [182] wymaga **zaangażowania wyższego kierownictwa** z trzech powodów [57]: (1) kierownictwo wysła komunikat, że projekt uważa za ważny i tym samym pomaga zwalczyć niechęć do uczestnictwa, (2) zapewnia zgodność wyników implementacji ze wszystkimi celami, z których część może być znana tylko naczelnemu kierownictwu, (3) zaangażowanie kierownictwa skutkuje większym zrozumieniem problemów implementacyjnych (szczególnie gdy kierownictwo otrzymuje codzienne informacje o postępach zespołu). Zasadniczą rolę naczelnego kierownictwa jest jednak katalizowanie procesu wdrażania, który nadzorować będą kierownicy średniego szczebla.

Inteligentne systemy informowania kierownictwa (ISIK) mają na celu realizację wczesnego ostrzeżenia przez niekorzystnymi sytuacjami, identyfikowanie nadchodzących problemów lub okazji (sposobność) rozpoczęcia działań. W tym przypadku inteligencja w SIK wyrażać się musi w **zdolności dostosowania działania** (reakcji) do okoliczności (wewnętrznych i zewnętrznych). ISIK monitorują poprzez **zestaw wskaźników ekonomicznych, stan podmiotu gospodarczego** i jego otoczenia, sygnalizując operatorowi **przekroczenie założonego zakresu wartości** ustalonych przez decydenta, wspierają procesy **oceny realizacji** poszczególnych **zadań**. Filozofia działania systemów informowania kierownictwa (*Executive Information Systems*) zwanych także systemami wspomaganie kierownictwa polega według Simona [277] na udostępnieniu naczelnemu kierownictwu firm **uproszczonej, bardzo skonsolidowanej** i przeważnie statycznej (w sensie ściśle ograniczonej liczby wybieralnych wariantów prezentacji) **wizji danych**. Postulowane cechy informacji prezentowanej przez SIK wynikają z obserwacji, że zwykle naczelnicy menedżerowie często nie mają czasu lub nawet zdolności wykonywania analizy dużej ilości danych (w rodzaju OLAP), **nie potrzebują manipulacji** wieloma wciąż rozwijającymi się poziomami zagnieżdżenia danych w celu odpowiedzi na pytania – chcą tylko rzucić okiem na ogólną sytuację, otrzymać odpowiedź na pytanie „o czym powinienem wiedzieć w tej chwili?”. Dlatego czyni się wysiłki, aby dostosować sposób obsługi SIK do środowiska

pracy menedżera. Zwykle SIK konfiguruje się za pomocą języków skryptowych. Można wyróżnić dwa rodzaje trybów obsługi SIK: *briefing book* oraz *command center*. Tryb *briefing book* (zwarty) emuluje na platformie elektronicznej (w czasie rzeczywistym) **zestaw dokumentów**, które wykorzystuje menedżer podczas kierownik podczas spotkań ze współpracownikami lub podwładnymi. *Briefing book* zawiera **szereg slajdów**, które prezentują w sposób zwarty i czytelny propozycje nowych przedsięwzięć. Jeżeli prezentacja dotyczy już uruchomionych projektów to dynamicznie powstające slajdy poparte mogą być tworzonymi na żądanie przekrojami danych i ich wykresami przedstawiającymi aktualny stan projektu. SIK obsługiwany w trybie *command center* (czyli centrum zarządzania) widziany jest od strony użytkownika jako **zestaw szuflad mieszczących raporty**, wiadomości ściągnięte z serwisów informacyjnych w internecie, komunikaty i komentarze dotyczące spółek. Odstępy czasu między sesjami korzystania przez menedżera z *command center* wyznaczać powinny **częstotliwość aktualizacji danych**. *Command center* może także przejmować inicjatywę i w razie nadejścia informacji wymagających uwagi i reakcji (np. przekroczenie granic bezpieczeństwa) menedżera generować alarm.

Zadania **systemów eksperckich** dotyczą przede wszystkim diagnozowania (osądzania związków przyczynowo-skutkowych), monitorowania, prognozowania i planowania finansowego. Inteligentne działanie systemów eksperckich dotyczy pewnego rodzaju obecnej w nich świadomości, objawiającej się zdolnością sporządzania wyjaśnień dotyczących sposobu (drogi) dojścia do rozwiązania problemu (ścieżka wnioskowania, osiągnięcie konkluzji).

Systemy **zarządzania modelami i danymi** są interesującym obszarem badań ze względu na strukturę, reprezentację i przetwarzanie bazy modeli decyzyjnych (drzewa decyzyjne, sieci decyzyjne, wykresy czasowe) oraz możliwości zastosowań sztucznej inteligencji, szczególnie integracji poszczególnych technik reprezentacji wiedzy. Jako rozszerzenie modelu danych w ujęciu *entity-relationship* (znanego jako technika modelowania systemów informatycznych) opracowano podejście strukturalnego modelowania, dotyczące struktury i reprezentacji, które omówione zostało przez Geoffriona [92].

Grupowe Systemy Wspomagania Decyzji (GDSS) mają na celu nadzorowanie i przeprowadzanie negocjacji lub głosowań a także doprowadzenie do porozumienia decydentów uczestniczących w poszczególnych etapach planowania strategicznego. Wspierają wymianę informacji między uczestnikami zespołu roboczego oraz umożliwiają komunikowanie się z innymi zespołami. DeSanctis i Gallupe [58] definiują GDSS jako **połączenie systemów komunikacji i technologii wspomagania decyzji** mające na celu ułatwienie grupom ludzi formułowanie i rozwiązywanie słabo ustrukturalizowanych problemów. Jednocześnie sytują metody wielokryterialnego podejmowania decyzji (MCDM) na drugim poziomie złożoności GDSS (narzędzi formalnych przydatnych w modelowaniu decyzji). Uzasadniają to możliwością interaktywnego konfigurowania tych metod oraz możliwością modelowania struktury i zachowania poszczególnych członków zespołu decyzyjnego jak

i zespołu jako całości. Opracowali ponadto trójpoziomą taksonomię funkcji GDSS (konkretny GDSS nie musi udostępniać wszystkich funkcji): (1) poziom techniczny, (2) poziom modelowania, (3) poziom doradztwa. **Poziom techniczny** dotyczy mediów i usług komunikacyjnych, dzięki czemu przestają istnieć bariery fizycznego oddalenia między decydentami i możliwe jest np. zdalne głosowanie czy ustalanie hierarchii celów, priorytetów i działań. **Poziom modelowania** dostarcza narzędzi modelowania problemów decyzyjnych oraz grupowego podejmowania decyzji których celem jest zmniejszenie niepewności warunków podejmowania decyzji i ich następstw, takich jak: wieloatrybutowe metody analizy użyteczności (ratingi, rankingi, metody punktowe), burze mózgów i metoda delficka czy technika grupy nominalnej [55]. **Poziom doradztwa** dotyczy wykorzystania wiedzy eksperckiej, sporządzania rankingu i sposobu wyboru reguł wnioskowania oraz podejmowania decyzji. Groupware jest wg Ellis, Gibbisa i Reina [74] „klasą aplikacji łączących przetwarzanie danych i komunikację, wspierających co najmniej dwu użytkowników zaangażowanych w wspólne zadanie oraz dostarczających interfejs do współdzielenia się środowiskiem”. Przesłanką wyłonienia się GDSS jako jednego z narzędzi *groupware* jest zatem konsolidacja i integracja rozległych systemów komputerowych i agentów decyzyjnych wynikająca z konieczności organizacyjnej polegającej na delegowaniu i rozproszeniu (dotyczącym funkcji, projektów, terytorium i czasu) odpowiedzialności. **Rozproszony proces podejmowania decyzji** zwiększa potrzebę konsultacji, narad i spotkań w celu szybkiego i prawidłowego reagowania na zmiany (tzn. podejmowanie dobrych decyzji) dokonujące się w rynku na którym działa przedsiębiorstwo (modelowanie oraz projektowanie BPR) oraz w środowisku. Co więcej, proces przystosowania spotkań do wykorzystania technologii wymusza jego formalizację w postaci ustrukturalizowanych procesów takich jak metoda delficka, technika grupy nominalnej, burza mózgów, głosowanie itd. Dyer i Forman [68] wyróżniają cztery **techniki określenia wspólnych preferencji grupy** decydentów: wyszukiwanie wspólnych opinii, głosowanie nad wykluczającymi się opiniami, normalizacja (uśrednianie) opinii, wyznaczanie reprezentatywnej (uśrednionej) opinii przy uwzględnieniu wag poszczególnych decydentów. Dennis i inni [57] wyróżniają dwa rodzaje technik przydatnych w realizacji BPR: radykalne i „drobnych kroczków” (*incremental*). Do **technik radykalnych** należą: (1) otwarte burze mózgów (mają na celu wygenerowanie idei bez oglądania się na realizm), (2) życzeniowe wyobrażenia przyszłości (ang. *future thinking*), (3) pozytywne artykułowanie przypowieści o zasadach BPR (*principles and parables*) mające na celu prezentacje udanych wdrożeń w innych firmach, (4) analogie międzybranżowe (próby „przeskoczenia” idei między branżami mającymi wspólne cechy), (5) stopniowe znoszenie ograniczeń (wyszukiwanie możliwości usunięcia ograniczeń i odkrywanie ukrytych, uważanych za oczywiste, założeń), (6) analizy „what-if” dotyczące ekstremalnych zmian parametrów procesów (np. katastrofalnych zdarzeń) i projektowanie decyzji w takich sytuacjach. Do technik „**drobnych kroczków**” zaliczyć można: (1) analizę problemową (uszeregowanie problemów i próby ich rozwiązania), (2) analizę kosztów opartą na technice rachunku kosztów działań, zgodną z metodologią

IDEF0 [190], (3) eliminację działań (dążenie do skreślenia z listy jak największej liczby działań bez ujemnego wpływu na funkcjonowanie organizacji), (4) próbę eliminacji poszczególnych elementów modelu IDEF wchodzących w relację z działaniami - ICOM (*input, control, output, mechanism*) bez ujemnego wpływu na funkcjonowanie organizacji – uzupełnienie punktu nr 3, (5) analizę czasu trwania działania (czasu obsługi) dla wszystkich działań tworzących sieć działań, (6) analizę wpływu wprowadzanych innowacji technologicznych na najistotniejsze działania. Prowadzą one do skonstruowania narzędzia, które zaspokoi potrzebę uzgodnienia różnych, subiektywnych celów a przede wszystkim pogodzenia różnych metod pomiaru i osądzania rzeczywistości, co związane jest z wydobyciem różnych aspektów wiedzy, tkwiących u podstaw poglądów poszczególnych osób uczestniczących w projektowaniu decyzji a nawet uzyskania efektu synergicznego. Zastosowanie w tym obszarze może znaleźć *cognitive science (nauka o poznaniu)*, badająca czynniki odpowiadające za zróżnicowanie metod poznania u różnych ludzi kształconych w różnych środowiskach i kulturach (wsparcie komunikacji wielojęzycznej). Biorąc pod uwagę kryteria osób zaangażowanych w przygotowanie i podejmowanie decyzji wyróżniamy [34] **cztery sposoby rozwiązywania problemu** związanego z **potencjalną niezgodnością** między decydentami: **autokratyczny** (pojedyncza osoba podejmuje decyzje bez porozumienia z członkami grupy), **konsultacyjny** (pojedyncza osoba podejmuje decyzje po zasięgnięciu opinii innych członków grupy wykonujących analizę problemu), **grupowy** (problem jest analizowany przez zainteresowanych członków grupy i decyzja podejmowana jest na drodze uzgodnienia – w tym głosowania), **delegowalny** (grupa wyłania osobę autonomicznie podejmującą decyzję). Zastosowanie w GDSS modułu *Knowledge-Based System* implikuje wykonanie wstępnej analizy dotyczącej przyporządkowania członkom grupy źródeł wiedzy i puli pytań odpowiednich do ich zakresu odpowiedzialności i kompetencji, na podstawie atrybutów problemu. Moduł systemu eksperckiego (KBS) zintegrowany z GSWD pełni rolę uprzywilejowanego uczestnika (wyroczni) lub zwykłego uczestnika dyskusji. Sage [256] określa GSWD jako wykorzystujący technologie informacyjną system wspomaganie decyzji udostępniający środki do **prowadzenia dyskusji grupowych** w czasie podejmowania decyzji w którym uczestniczy grupa ludzi. GSWD powinny wyposażać uczestników (często przebywających w różnych częściach świata) w narzędzia komunikacyjne i obliczeniowe oraz techniki grupowego podejmowania decyzji. Bui [30] wyróżnia ponadto **następujące cechy** GSWD: uczestnicy zachowują się wobec siebie „fair”, grupa debatuje nad znanymi wszystkim członkom grupy alternatywami (wariantami decyzji), każdy z uczestników posiada własne cele, które dają się określić a priori, a jeśli uczestnicy nie potrafią dojść do ugody, to poszukiwane są inne rozwiązania.

Inteligentne Systemy Wspomaganie Decyzji wyposażone są w **moduły zarządzania danymi** (bazy danych: zawartość, struktura i system zarządzający, zapewniający procedury dostępu), **zarządzania modelami** (modele ilościowe, interpretujące zmienne o charakterze strategicznym, korzystające z wiedzy z zakresu finansów, statystyki, teorii zarządzania), **dialogu** (udostępniający

użytkownikowi możliwość interaktywnej pracy z systemem a otoczeniu programowemu interfejs służyć wymianie danych i sterowaniu) i **zarządzania wiedzą** (wspomaga, nadzoruje wykonywanie zadań zleconych przez operatora systemu i umożliwia integrację pozostałych modułów) [33].

3.8 Regułowe systemy eksperckie

3.8.1 Charakterystyka i struktura systemów eksperckich

System ekspercki (ekspercki) [277] jest systemem informatycznym charakteryzującym się **zdolnością rozwiązywania problemów** o takiej skali trudności, że wymagają ekspertyzy specjalistów. Jak piszą Shepherd [269], Edwards i Connel [71] system ekspercki jest programem komputerowym od którego wymagamy uzyskania porady pozwalającej na podjęcie decyzji w taki sposób, **jak podjąłby ją ekspert** lub charakteryzującą się podobnymi rezultatami. Systemy eksperckie korzystają ze sformalizowanych ekspertyz przeprowadzonych przez ludzkich ekspertów. Ludzcy eksperci posługują się **wiedzą także heurystyczną**, nabytą w drodze **doświadczenia**. System ekspercki oprócz postulatu rozwiązywania problemów z danej dziedziny powinien rozwiązywać problemy postępując w **sposób podobny do ludzkiego eksperta** (imitując jego postępowanie). Cohen [46] pisze: „prowadząc badania z wykorzystaniem systemu eksperckiego nie oczekujemy że wnioskowanie heurystyczne jest drogą do poznania teorii ukrytej przed nami; przydatność systemu eksperckiego nie wynika z żadnego formalnego opisu dziedziny ale ze zgromadzenia dużej liczby sprytnych technik prowadzących na skróty do rozwiązania problemu rozwiniętych dzięki latom praktyki przez ekspertów”. Należy zgodzić się z Malhotrą [183], że prace nad tworzeniem systemów eksperckich i (szerzej) zastosowań sztucznej inteligencji mogą być przydatne **w zrozumieniu w jaki sposób człowiek przetwarza informacje w działania a następnie ocenia jego wyniki**. Najpierw jednak należy ustalić, w jaki sposób z **informacji wygenerować ich znaczenie** (interpretować informacje) na podstawie której podejmowane są decyzje (i działania). Co więcej, jak pisze Nguyen [216]: „większość ekspertów nie potrafi wyrazić w jaki właściwie sposób należy podejmować decyzje”. Bradley i Hauser [26] wymieniają szereg przesłanek przewagi systemów eksperckich nad tradycyjnymi systemami informatycznymi zarządzania (SIZ):

1. W przeciwieństwie do SIZ, praca z systemem eksperckim polega na **wspomaganiu przez niego heurystycznego wyszukiwania** (a nie algorytmicznego) takiego fragmentu zgromadzonej reprezentacji wiedzy (bazy wiedzy) która umożliwi rozwiązanie problemu (znalezienie odpowiedzi na zadane pytanie) w sposób **bardziej podobny do ludzkiego rozumowania**
2. Baza wiedzy systemu eksperckiego wypełniana jest wiedzą pochodzącą od **ekspertów dziedzinowych** (mniejsza rola użytkownika, który odpowiada przede wszystkim na pytania w trybie dialogowym, odbiera raporty i na ich podstawie podejmuje decyzje), sam **proces wnioskowania przebiega automatycznie**, natomiast tradycyjne SIZ korzystają z algorytmów opracowanych w celu rozwiązywania konkretnych problemów i są zarządzane (obsługiwane, konfigurowane) bezpośrednio przez użytkownika
3. Od twórcy systemu eksperckiego, **inżyniera wiedzy** wymaga się **znacznie większych**

umiejętności, gdyż stworzenie systemu zarządzającego wiedzą pozyskaną od eksperta jest dużo trudniejsze niż budowa SIZ w oparciu na standardowych specyfikacjach, systemów korzystających z ustrukturalizowanej wiedzy.

4. Dane wprowadzane do systemu eksperckiego mają charakter liczbowy, symboliczny lub są zdaniem języka naturalnego. Zawsze jednak zawierają **terminy należące do rozważanej dziedziny**. Dane po przetworzeniu definiują strukturę problemu i nazywane są faktami lub atrybutami. Źródłem danych mogą być **eksperymenty symulacyjne, parametry otoczenia** np. informacje o branży, bieżącej koniunkturze itd. System ekspercki oparty na regułach operuje na faktach i regułach wnioskowania, na podstawie których tworzone są nowe fakty. Reguły mogą być [62] heurystyczne, zawierać subiektywne informacje (wpływając na proces oceny i rozwiązywania problemu przez określonego specjalistę), respektować intuicyjne domysły, przypuszczenia, zdroworozsądkowe zasady postępowania.

Niederliński [217] podkreśla, że reguły są przykładem wiedzy dziedzinowej o **charakterze ogólnym** a fakty - wiedzy dziedzinowej o **charakterze szczegółowym**. **Faktem** może być zdanie „bieżąca płynność mniejsza niż dobra”. „bieżąca płynność” to nazwa faktu (atrybutu), „mniejsza niż” to operator relacji natomiast „dobra” to wartość atrybutu. **Reguła** może być przedstawiona jako wyrażenie “if <x> then <y> else <z>”. W części <x> zapisywane są przesłanki, w część <y> konkluzje natomiast część <z> jest opcjonalna. Po wyznaczeniu wartości logicznej przesłanek możliwe jest określenie wartości części <y> i <z>. Reguła wiążąca zdolność kredytową i decyzję o transakcji sprzedaży wyglądać może następująco:

If zdolność kredytowa jest dobra i wielkość transakcji sprzedaży jest mniejsza niż 10000 **Then** decyzja akceptacji propozycji transakcji sprzedaży **Else** decyzja odrzucenia propozycji transakcji sprzedaży.

Jeśli przesłanki reguły są prawdziwe to **efektem aktywacji reguły wnioskowania są nowe fakty**. Gdy reguła oprócz części *if-then* zawiera także część *else* to reguła owocuje nowymi faktami, także gdy wyrażenie w części przesłankowej nie jest prawdziwe. **Przesłanki** są wyrażeniami logicznymi zawierającymi miejsce na wstawienie konkretnych faktów (atrybutów). Uruchomiona (aktywowana) reguła przypisuje wartości poszczególnym atrybutom występującym w przesłance. Wang [316] zauważa dużą zaletę **regułowej (deklaratywnej) reprezentacji** wiedzy: **wiedza ta może być zmieniana dość łatwo** – zwykle wystarczy zmodyfikować kilka wyrażień. **Wiedza proceduralna** nie jest na tyle elastyczna: nawet małe doskonalenie wiedzy proceduralnej wymagać może **znaczących zmian** w oprogramowaniu komputerowym. Liebowitz [172] wyróżnia następujące główne komponenty systemu eksperckiego: interfejs użytkownika, mechanizm wnioskowania oraz bazę wiedzy. **Mechanizm wnioskowania** [Rys. 3-4] łączy metody wnioskowania oraz strategie sterowania wyborem metod wnioskowania np. forward (przewodnia rola danych) lub *backward chaining* (udowadnianie hipotez). Selig i Johannes [264] wskazują, że wnioskowanie zapisywane jest za pomocą

pojęć elementarnych (pierwotnych, *primitives*) języka. W większości regułowych systemów eksperckich **elementarnymi pojęciami są fakty i reguły**. Fakty mają postać zdań orzekających, co nie przesądza, że mogą w niektórych przypadkach być tylko hipotezami (błędnymi lub prawdziwymi). System ekspercki może służyć przygotowaniu zestawu zmiennych decyzyjnych – danych wejściowych dla modelu symulacyjnego lub ocenie wyników przedsiębiorstwa. W takiej sytuacji wiedza teoretyczna lub doświadczenie eksperta przyczyniają się do interpretacji wyników eksperymentów i sugerowania ewentualnych zmian w zestawie parametrów wejściowych. Należy jednak pamiętać że w realnie użytkowanych systemach eksperckich spełnione jest **założenie o zamkniętym świecie** (*closed-world assumption*) opisanym np. przez Geneseretha i Nilssona [91]. Przykładem tego problemu może być pytanie „Czy przedsiębiorstwo Omega w roku 1999 jest rentowne”, gdy aktywnym jest okres 2000-2002. Jeśli zachowane jest założenie o zamkniętym świecie, to odpowiedź będzie brzmiała: nie, chociaż nie wiemy nic o rentowności w roku 1999, gdyż informacja o sytuacji w roku 1999 znajduje się w węźle reprezentującym lata 1997-1999. Pełna informacja o roku 1999 będzie dostępna po uaktywnieniu węzła reprezentującego lata 1997-1999, będącego poprzednikiem węzła aktywnego w momencie zadania pytania. Sytuacja może być jeszcze bardziej złożona, gdy mamy do czynienia z drzewem posiadającym więcej niż jeden węzeł dotyczący tego samego okresu. Sytuacja taka ma miejsce, gdy opracowano wiele wariantów decyzyjnych dla tego samego okresu.



Rys. 3-4 Typowa struktura regułowego systemu eksperckiego. Źródło: [77]

System ekspercki jest także określany jako **system z bazą wiedzy** (KBS) i wg Durkina [65] jest programem komputerowym wyposażonym w **uznaną ludzką wiedzę i ekspertyzę** w celu modelowania możliwości rozwiązywania problemów przez ludzkiego eksperta. Należy jednak podkreślić [18] że rozwiązywanie problemów (wykonanie ekspertyzy) przez człowieka (eksperta) i system ekspercki są od siebie bardzo daleko. System ekspercki modeluje **zaledwie niektóre funkcje ludzkiego eksperta** (rozumiejącego i zdolnego do uogólnienia swoich koncepcji i strategii, adaptacji do zmian środowiska). Barthelemy stwierdza, że „samoloty nie naśladują ptaków”.

System ekspercki składa się z następujących modułów: **pozyskiwania wiedzy** (ważny dla środowiska rozwoju, opisanego dalej), **mechanizmu wnioskowania** (ważny dla środowiska konsultacji), **bazy wiedzy**, **interfejsu użytkownika** wraz z **modułem objaśniania**. W strukturze systemu eksperckiego wg Qiana [242] i innych wyróżnić można następujące zasadnicze **moduły**: interfejs użytkownika, monitor, mechanizm wnioskowania oraz objaśniający, pozyskiwania i zarządzania wiedzą, bazę wiedzy oraz bazę danych (przechowującą zarówno dane potencjalnie

wykorzystywane do przeprowadzenia wnioskowania – informacje pochodzące z systemu rachunkowości oraz pozostałe dane gospodarcze, jak również zmienne (pomocnicze) przechowujące wyniki pośrednie podczas wnioskowania w systemie eksperckim). Zadaniem **interfejsu użytkownika** jest komunikacja (wydawanie poleceń, kontrola ich wykonywania i prezentacja wyników) między użytkownikiem (operatorem lub systemem komputerowym) oraz systemem eksperckim. **Moduł monitora** znajduje zastosowanie głównie w systemach eksperckich czasu rzeczywistego (np. wspierających operacje giełdowe i bankowe) podczas śledzenia na bieżąco zachowania wszystkich obserwowanych parametrów i zmiennych systemu oraz otoczeniu w celu identyfikacji sytuacji diagnozy sytuacji niezwykłych i błędów. Identyfikacja powyższych sytuacji przez monitor umożliwia przedsięwzięcie reakcji przez operatora lub wykonanie wnioskowania z wykorzystaniem reguł przewidzianych w sytuacjach awaryjnych. **Mechanizm wnioskujący** wykonuje główne zadanie systemu eksperckiego czyli naśladowanie pracy ludzkiego eksperta dziedzinowego poprzez sterowanie procesem wnioskowania (rozwiązywania problemu). Dzięki mechanizmowi wnioskowania który znajduje się w module przetwarzania wiedzy, opracowywane są wnioski lub porady na podstawie wiedzy wydobytej od eksperta oraz dostępnej informacji o aktualnie badanym obiekcie. Mechanizm wnioskowania odpowiedzialny jest za rozszerzenie wiedzy pierwotnej (dotyczącej obserwowanych faktów i teorii dziedzinowej) o wiedzę wtórną, na którą składają się hipotezy potwierdzone przez proces wnioskowania, generalizacje przeprowadzone na podstawie zbioru poszczególnych wnioskowań, wartości formuł logicznych i obliczeń. **Moduł objaśniający** przedstawia w czytelnej dla operatora formie ścieżkę wnioskowania, objaśnia przesłanki wykorzystania poszczególnych reguł i faktów; może być zintegrowany z interfejsem użytkownika w celu prezentowania rozszerzonego raportu (wyników wnioskowania). **Pozyskiwanie i zarządzanie** wiedzą obejmuje przetwarzanie i przenoszenie wiedzy artykułowanej (ekspertyzy) przez ludzkiego eksperta dziedzinowego do wewnętrznego formatu reprezentacji wiedzy (dialektu komputerowego) w systemie eksperckim, badanie i utrzymanie integralności bazy wiedzy (spójności dotychczasowych reguł i faktów oraz nowych treści, wprowadzanych przez operatora, pochodzących z aktualizowanej bazy danych lub będących produktem wnioskowania czyli wynikających z obecnego stanu bazy wiedzy) oraz elementarne funkcje zarządzania bazą wiedzy jak wprowadzenie, kasowanie, wyświetlanie, edycję (modyfikację) poszczególnych zapisów (rekordów zawierających reguły wnioskowania) bazy wiedzy i bazy danych. Cechą charakterystyczną KBS jest obecność **bazy wiedzy** (wydobyta od ekspertów wiedza dziedzinowa zapisana w postaci powiązanych ze sobą faktów, reguł i pojęć), zapisana w notacji języka opisu wiedzy. Zadaniem **modułu bazy wiedzy** jest przechowywanie wiedzy dziedzinowej, w tym dotyczącej metod rozwiązywania problemów o bardziej ogólnym charakterze (tzw. wiedza zdroworozsądkowa). Zwykle baza wiedzy jest wielopoziomową strukturą wielu reguł i faktów, zintegrowaną poprzez interfejs z mechanizmem wnioskowania, modułem pozyskiwania wiedzy oraz modułem zarządzającym (czuwającym nad jej spójnością).

W **bazie wiedzy** (dziedzinowej) wyróżnić można [242]: **wiedzę podstawową**, reprezentującą reguły (podręcznikowe, ogólnie znane) służące przede wszystkim wykrywaniu sytuacji nadzwyczajnych (alarmowych) – wymagającej niekiedy szybkiej reakcji, **wiedzę odwzorowującą** **wiedzę eksperta** (doświadczalną, związaną z konkretnym ekspertem lub wynikającą z zespołu jego poglądów na związki przyczynowo-skutkowe), **metawiedzę** (dotyczącą opisu struktury i zasad utrzymania i korzystania z pozostałych klas wiedzy) – obecność jej umożliwia m.in. wykonywanie optymalizacji działania mechanizmu wnioskowania (jeśli baza wiedzy zawiera wiele reguł, to znalezienie odpowiedniej reguły wymaga zbyt wiele czasu; zaleca się zatem stosowanie mechanizmu buforowania oraz działania zmierzające do przewidywania na które reguły będzie zapotrzebowanie i przechowywanie w pamięci o najszybszym dostępie (pamięć RAM zamiast dyskowej)) oraz **wiedzę niezbędną do generowania wariantów decyzyjnych** – obejmującą także reguły wyjaśniające poszczególne propozycje (zalecenia dotyczące podejmowanych w przyszłości działań).

Wiedza przed zakodowaniem w bazie wiedzy jest zwykle pofragmentowana i ulotna. Po umieszczeniu w bazie wiedzy można ją zaklasyfikować do dwu grup: **wiedzy faktograficznej** (baza faktów korzystająca z baz danych) oraz **strategicznej**, korzystającej z zasad wnioskowania właściwych dla dziedziny problemowej (baza reguł i przypadków). Reguły (IF/THEN) mają postać **zdań składających się z przesłanek i konkluzji**. W łańcuchu wnioskowania **przesłanki** mogą **pełnić rolę pytań** bądź **hipotez**, których wartość logiczną mechanizm wnioskowania próbuje ustalić. Wiedza może posiadać cechy **deklaratywności** (zapisane są w ten sposób dane i fakty – tworzące bazę faktów oraz obiekty, relacje między nimi, definicje terminologiczne) oraz **deskryptywności** (zapisane są w ten sposób reguły, heurystyki i tzw. demony – określane wspólną nazwą bazy reguł) lub wspomnianej wcześniej **metawiedzy**. Jako **wiedzę faktograficzną** traktuje się fakty, koncepcje, teorie właściwe dla dziedziny problemowej a **jej uzupełnieniem** (bardziej abstrakcyjnym) są heurystyki, reguły i strategie (dotyczące zarówno obszaru problemowego jak i ogólnego zastosowania). **Bazę faktów** tworzy zbiór faktów wprowadzonych przez operatora oraz nowych faktów będących wynikiem procesu wnioskowania na podstawie reguł i faktów już obecnych w systemie.

W systemie eksperckim stosuje się koncepcję **programowania nieproceduralnego**, polegającą na reprezentacji dziedziny problemowej (wycinka rzeczywistości, dla którego obowiązują specyficzne prawa oraz zasoby informacyjne) za **pomocą reguł** (ogólnie: rachunku predykatów, relacji logicznych), uwzględniających lub nieuwzględniających **czynnik czasu** [242]. Zmiany stanu badanego systemu w czasie (czynnik czasu) mogą być ujmowane poprzez **rejestrwanie** (kodowanie) charakteru lub (bardziej kategorycznie) kierunku zmian oraz ich prędkości. Są to cechy odróżniające reprezentację wiedzy w systemach eksperckich i pakietach (językach) programowania proceduralnego. Jak wiadomo, **języki proceduralne** przetwarzają (generują rozwiązania) poprzez zapisane w nich programy **kolejno linia po linii** (wyjątkiem są instrukcje wywołania procedur lub instrukcje skoku), natomiast **kolejność czynności w regułowych systemach eksperckich nie jest zdeterminowana**

przez programistę i zależy od tzw. **mechanizmu wnioskującego**, dobierającego właściwe reguły odwołujące się do faktów obecnych w bazie wiedzy w celu przeprowadzenia wnioskowania, tzn. ustalenia prawdziwości hipotez (wyrażających np. warianty decyzyjne) postawionych przez użytkownika.

Mechanizm wnioskowania udostępnia następujące usługi [109]: **wnioskowanie w tył/wprzód**, ujednoczenie zapisu i porównywalność, propagację ograniczeń, zarządzanie priorytetami, **agendę czyli strukturę porządkującą i sterującą zadaniami** (regułami) do sprawdzenia (wywołania), **architekturę tablicową** (korzystanie poszczególnych ekspertów z tej samej bazy wiedzy oraz wzajemne udostępnianie wyników wnioskowania przez poszczególne moduły, nazywane źródłami wiedzy), **wnioskowanie po przypisaniu wartości hipotezom**, operacje logiczne, **wnioskowanie indukcyjne**, uruchamianie demonów, **metasterowanie**, zarządzanie czynnikami niepewnymi, wykonywanie obliczeń matematycznych itd.

Wnioskowanie wprzód (*forward chaining*) i **wnioskowanie w tył** (*backward chaining*) są głównymi strategiami wnioskowania stosowanymi w regułowej reprezentacji wiedzy. Obie strategie można **mieszać ze sobą** (*mixed chaining/reasoning*). Wnioskowanie wprzód, nazywane także *forward reasoning* rozpoczyna się od przygotowania dostępu do znanych faktów charakteryzujących aktualny (ale także przeszły czy prognozowany stan) a kończy się sporządzeniem wykazu wszystkich nowych faktów wygenerowanych na podstawie warunków początkowych (faktów) oraz zbioru dostępnych i zastosowanych reguł wnioskowania. *Forward reasoning* cechuje się złożonym, wielowymiarowym charakterem wyniku wnioskowania, na który może składać się wiele faktów i uruchomionych reguł. Wnioskowanie w tył, nazywane także *backward reasoning* rozpoczyna się od postawienia pytania (twierdzenia) systemowi, który ma wygenerować odpowiedź, przesądzając o jego prawdziwości. Operator przywołuje hipotezy dotyczące faktów lub wyrażenie zawierające hipotetyczne fakty (w znanym przez system formacie, należący do dziedziny problemowej) i przypisuje mu (hipotetyczną, wymagającą rozstrzygnięcia) wartość logiczną (prawda, fałsz, a w przypadku wnioskowania rozmytego/logiki wielowartościowej czy uwzględnienia czynnika niepewności także wartości pośrednie). Odpowiedź systemu zawierać powinna potwierdzenie lub zaprzeczenie postawionej tezy (ewentualnie w kategoriach logiki rozmytej itd.) bądź komunikat o braku możliwości rozpatrzenia logicznej wartości hipotezy (np. brak odpowiednich reguł, sprzeczności w bazie wiedzy, nieprawidłowości w zadanym pytaniu, wyjście poza dziedzinę problemową). Powyższy tryb znajduje zastosowanie przede wszystkim w systemach typu diagnostycznego oraz ostrzegających przed sytuacjami nadzwyczajnymi (niekoniecznie niekorzystnymi).

Jak twierdzą Leonard-Barton i Sviokla [169] w przypadku systemów eksperckich można mówić o **przesunięciu ośrodka projektowania**, budowy i rozwoju z działów informatyki na użytkowników (posiadających znaczną wiedzę techniczną i jednocześnie dotyczącą danego problemu). Systemy eksperckie charakteryzują się **separacją pamięci krótkoterminowej** (przechowującej

ścieżkę wnioskowania wraz z wykorzystywanymi faktami i aktywowanymi regułami, wynikami pośrednimi i wynikami końcowymi; jest to odpowiednik ludzkiej pamięci krótkotrwałej zawierającej informacje do interpretacji) i **długoterminowej** (baza wiedzy: fakty i reguły które nie muszą rezydować w pamięci roboczej, aktywowane w razie potrzeby; odpowiednik ludzkiej pamięci długotrwałej przechowującej doświadczenie, wiedzę i mądrość) oraz **sterowania przetwarzaniem wiedzy**. W tradycyjnych SIZ wiedza i sterowanie są w większym stopniu zintegrowane w postaci algorytmów. Niestety problemy spotykane w SI a w szczególności w SE polegają na ich (praktycznej) bezsilności w obliczu rzeczywistych procesów gospodarczych, charakteryzujących się szybkozmiennymi a nawet "skaczącymi" trajektoriami. Malhotra [183] stawia tezę że SI i SE operują na finalnych, końcowych wartościach opisujących proces, nie biorąc pod uwagę na etapów pośrednich. Podobnie jak człowiek będący ekspertem, system ekspercki prowadzi **wnioskowanie na podstawie dwu typów informacji**. Pierwszy typ to **dane ilościowe** natomiast drugi – **jakościowe**. Shepherd [269] wymienia następujące atrybuty pozwalające na **odróżnienie** systemu eksperckiego od innych systemów wspomagania decyzji:

1. Preferowanie raczej **wnioskowania heurystycznego** niż wykonywania obliczeń
 2. Poszukiwanie raczej **możliwości zastosowania rozwiązań** (wykonalności) niż ich optymalności. Jest to także jeden z nierozwiązanych problemów występujących w badaniach symulacyjnych
 3. **Oddzielanie** procedur **sterowania** od **reprezentacji wiedzy** (co doprowadziło do powstania komercyjnych *expert system shells* z pustą bazą wiedzy)
 4. Przeznaczenie do rozwiązywania **źle ustrukturalizowanych problemów**
 5. Przystosowanie do **problemów dziedzinowych**
 6. Potrzeba stosunkowo **intensywnej interakcji** między systemem a użytkownikami oraz ekspertami.
- Należy jednak pamiętać o poglądzie Malhotry [183] wg którego SI i SE przydatne są jednak przede wszystkim w **dobrze ustrukturalizowanych problemach**. Malhotra za Churchmanem [42] dzieli je na *Locke'owskie* – oparte na **obserwacjach** (z silnym konsensusem dotyczącym natury sytuacji problemowej - co jest **trudnoosiągalne**) i *Leibnitz'owskie* – oparte na **wnioskowaniu** (systemy zamknięte, posiadające teorię aksjomatyczną, **łatwo je falsyfikować**).

3.8.2 Klasyfikacja regułowych systemów eksperckich

Bradley i Hauser [26] za Meyerem i Curleyem [196] przytaczają **podział systemów eksperckich** na cztery kategorie [Tab. 3-2]. Podstawą podziału było **wyróżnienie dwóch wymiarów: złożoności przetwarzanej wiedzy** (głębokość, specjalistyczność przechowywanej wiedzy, zakres wiedzy odwzorowany w bazie wiedzy, poziom umiejętności eksperckich niezbędnych aby rozwiązać problem, czas potrzebny do przeprowadzenia konsultacji, oczekiwana dokładność i pewność wyników

konsultacji) i **złożoności technologicznej** (zróznicowanie oprogramowania i sprzętu składającego się na SE, złożoność sterowania, interfejsu i bazy wiedzy, intensywność i skala korzystania z baz danych, sieci komputerowych, stopień integracji i korzystanie z rozwiązań hybrydowych). Podział jest przydatny przy projektowaniu architektury odpowiednio do planowanych zastosowań i wymagań i przedstawia się następująco:

Tab. 3-2 Macierz klasyfikacji systemów eksperckich. Źródło: [26]

^ ----- Złożoność wiedzy ----- <	Przeznaczone dla decydentów intensywnie korzystających z wiedzy Zawiera wiedzę doświadczonych ekspertów Zachowana prostota interfejsu i struktury programowo-sprzętowej Przydatne i opracowane w celu zastosowania przez konkretnych decydentów (pojedynczych lub pracujących w grupie) Dzięki nim decydent nie musi być znawcą wszystkich niuansów dziedziny Kluczowa rola inżyniera wiedzy i pozyskiwania wiedzy.	Przeznaczone do zadań strategicznych Złożone zarówno dziedzinowo (wiedza doświadczonych decydentów) jak i technologicznie (usieciowienie, hybrydowość, architektura modułowa) Konstrukcja integrowana (wewnętrznie i zewnętrznie – zewnętrzne bazy danych, rachunkowość, produkcja, zarządzanie wiedzą, interaktywność, usieciowienie) Przystosowanie do formułowania i weryfikacji alternatyw (testowanie hipotez, modele organizacji i środowiska, symulacja)
	Wspomagające osobistą wydajność Zawiera stosunkowo niewiele wiedzy, korzysta z prostych rozwiązań technologicznych Poprawia strukturę procesu podejmowania decyzji, przyspiesza proces podejmowania decyzji a zatem i wydajność Decyzje mają charakter rutynowy, powtarzalny	Ukierunkowane na technologie i ich rozwój Założenie o prostocie dziedziny problemowej Przystosowanie do implementacji złożonych metod rozwiązywania problemów przy odpowiedniej szybkości obliczeń Cel zastosowania obejmuje utrzymanie lub wzrost wydajności produkcyjnej, zatrzymanie wzrostu kosztów itd.
	< ----- Złożoność technologiczna ----- >	

3.8.3 Regułowe systemy eksperckie – zalety i wady

Podstawowe przesłanki przydatności systemów eksperckich w rozwiązywaniu danego problemu są wg Mulawki [209] następujące:

- a) **Wiedza** o rozważanej dziedzinie lub obiekcie badań jest **niepewna lub niekompletna** (chodzi tu o możliwość wykorzystania systemu eksperckiego do generowania nowej wiedzy na podstawie dostępnych informacji, wyprowadzania nowych faktów nawet w sytuacji niekompletnej lub niepewnej informacji o przesłankach, badania wrażliwości wyników analiz na zmiany parametrów poszczególnych reguł wnioskowania)
- b) Zachodzi potrzeba podjęcia decyzji w warunkach **braku** (lub niemożności jej pozyskania

w konkretnej sytuacji) **sformalizowanej wiedzy** o technikach rozwiązywania problemów w danej dziedzinie. Jest to sprecyzowanie przesłanki (a) jeśli chodzi o rozważaną dziedzinę. Od **systemu eksperckiego** oczekuje się **wsparcia w sytuacji niedostępności decyzyjnych modeli statystycznych i matematycznych, rozwiązań algorytmicznych** koniecznych do rozwiązania problemu, czyli właśnie wymagających wiedzy eksperckiej (heurystycznej). Jednocześnie problem powinien cechować się **dużą złożonością, słabą strukturalizacją** a rozwiązanie powinno wiązać się z poruszaniem się w „dużej” przestrzeni rozwiązań - chodzi tu o nie wykorzystywanie SE dla rozwiązywania trywialnych choć niealgorytmicznych problemów ze względu na znaczne koszty opracowania, chociaż nie jest to niemożliwe czy „zakazane”

c) Istnieje uświadomiona **potrzeba koordynacji** lub prowadzenia **jednolitej i stabilnej polityki** w poszczególnych oddziałach firmy. System ekspercki (a właściwie baza reguł) umożliwia zorganizowanie **jednolitych zasad postępowania w warunkach kryzysu**, rozproszenie procesu podejmowania decyzji i **delegowanie uprawnień**. Na podstawie **bazy faktów generowanych w poszczególnych ośrodkach** powstawania sytuacji problemowych (alarmowych) można formułować zasady postępowania w miejscach dotychczas niezagrażonych. Z drugiej strony **sparametryzowanie bazy reguł i architektura tablicowa** umożliwia interpretowanie wniosków generowanych przez system ekspercki odpowiednio do lokalnych uwarunkowań

d) **Kodyfikacja wiedzy i doświadczenia ekspertów dziedzinowych** (dobro trudnodostępne i kosztowne) mają na celu częste wykonywanie konsultacji i uwzględnienie różnych stanowisk i obiektywizmu (architektura tablicowa, analityczne modele decyzyjne). Nie zawsze wnioskowanie zgodne z **subiektywnym osądem eksperta** (jego wiedza jest także ograniczona), stosujące indywidualne heurystyki i strategie rozwiązań problemu musi być zgodne z osiągniętym dzięki konsultacji z wieloma źródłami wiedzy (ekspertami) wynikiem procesu wnioskowania (decyzją o dalszym postępowaniu, kondycji finansowej przedsiębiorstwa, zdolności kredytowej itd.). Jednak jak twierdzi Zopounidis i inni [342] użytkownicy klasycznych SWD są **w większym stopniu zaangażowani** w rozwój systemu

e) Istnieje **potrzeba wykorzystania informacji o charakterze jakościowym** a wykonanie symulacji (eksperymentu z odwzorowaniem funkcjonowania systemu w rzeczywistych i prognozowanych warunkach) na podstawie modelu badanego systemu może być **metodą weryfikacji proponowanych rozwiązań**. Widać tu sposobność zastosowania podejścia **hybrydowego i struktury system ekspercki – symulacja**

f) Istotną wartością jest **wyjaśnienie wyprowadzanych konkluzji** (w klasycznym SWD nie akcentuje się potrzeby posiadania takiej możliwości).

Zaletą diagnoz i zaleceń pochodzących z systemu eksperckiego w porównaniu z osądami wydanymi przez pojedynczego ludzkiego eksperta może być **jednoczesne wykorzystywanie wiedzy** wydobytej od wielu ekspertów. W tym kontekście trafnym wydaje się spostrzeżenie Aguirre'a i innych

[3] „współdzielenie wiedzy to potęga” nawiązującego do powiedzenia Sun K’uanga czy Francisca Bacona „wiedza to potęga”. **Cykl życia systemu eksperckiego** wg Liebowitza i Letsky’ego [172] składa się z następujących faz: **wyboru problemu**, który ma być rozwiązywany z zastosowaniem systemu eksperckiego, **pozyskiwania wiedzy, opracowania i zakodowania reprezentacji wiedzy** (opracowanie modelu i wybór narzędzia), **testowania i oceny** (porównania wyników uzyskiwanych przez system oraz ludzkiego eksperta) oraz **zastosowania i utrzymywania** (pielęgnacji).

Główne korzyści z zastosowania systemów eksperckich są następujące: system ekspercki **przedstawia projekcje** tego co może zdarzyć się w przyszłości – wspomaga ocenę czynników ryzyka, **prezentuje zebrane informacje** - fakty – w formie ułatwiającej diagnozę bieżącej sytuacji, pozwala na **przechowywanie obserwacji**, w jaki sposób eksperci rozwiązują problemy w celu późniejszego wykorzystania. SE przedstawia istotne aspekty tego co się stało i dlaczego się stało, **umożliwia szkolenie użytkowników** na podstawie dostępnej bazy wiedzy i procedur rozwiązywania problemów, pozwala na wykorzystanie doświadczenia eksperta **jednocześnie w wielu miejscach** i w większych projektach niż pojedynczy ekspert byłby w stanie ogarnąć, wreszcie zmniejsza następstwa związane z różnymi przyczynami zaprzestania współpracy z organizacją. **Deklaratywna reprezentacja** wiedzy posiada przewagę nad **reprezentacją proceduralną** polegającą na większej łatwości wprowadzania zmian w bazie wiedzy, gdyż nawet małe zmiany w zawartości wiedzy proceduralnej mogą skutkować znacznymi zmianami w programie źródłowym z drugiej strony przetwarzanie wiedzy deklaratywnej jest **dużo wolniejsze** niż wiedzy proceduralnej. **Znaczne problemy** sprawia także przetwarzanie wiedzy z uwzględnieniem jej **semantyki** gdyż kategoryzacja wykonywana przez eksperta [252] posługuje się raczej **rozumowaniem przez analogie**, konstruowaniem prototypowych rozwiązań i poszukiwaniu zgodności z typami kategorii niż **taksonomią** (którą da się łatwiej zaimplementować w systemie komputerowym); większość wynika jednak z **ograniczonej elastyczności** języka reprezentacji wiedzy.

3.8.4 Modelowanie wiedzy w systemach eksperckich

W rozważaniu **procesu pozyskiwania wiedzy** i jej późniejszego **wykorzystania** (szerzej: modelowania działań wchodzących w skład zarządzania wiedzą) przydatny jest dwuwymiarowy *knowledge chain model (KCM)*. Został on szerzej zaprezentowany przez Holsapple’a i Singh’a [119]. Ma on na celu przedstawienie etapów rozwoju wiedzy (uczenia się organizacji) oraz jej „emanacji” w postaci zaangażowania zasobów organizacji w eksplorację i rozwój środowiska. Podstawowe **grupy działań** (operacji) przedstawione w modelu *KCM* zostały **uporządkowane** w kierunku **coraz większego zasobu wiedzy**, jej przydatności i **interakcji ze środowiskiem** (od pasywnego pochłaniania informacji przez jej interpretację aż do wykorzystania w organizacji będącej aktywnym

podmiotem gry rynkowej). Wyróżnia się zatem: **pozyskiwanie** wiedzy, **selekcję** wiedzy, **generowanie** wiedzy, **internalizację** (rozsyłanie wewnątrz organizacji) i **eksternalizację** (wykorzystanie wiedzy podczas interakcji ze środowiskiem).

Pozyskiwanie wiedzy polega na zbieraniu (odpłatnemu nabywaniu, wywiadom, monitorowaniu), identyfikowaniu, interpretowaniu, organizowaniu informacji (aktualnych!) pochodzących ze źródeł zewnętrznych (i niekiedy wewnętrznych, ale całkowicie nieuświadomionych) oraz ich przetwarzaniu na postać (reprezentację), która może być wykorzystana w kolejnych działaniach. **Selekcja** (filtrowanie) wiedzy pozwala na powiązanie pozyskanej wiedzy (a zatem obecnych wewnątrz organizacji) z istniejącymi potrzebami organizacji. Zadaniem grupy operacji realizujących selekcję wiedzy może być identyfikacja przyczyn niskiej sprzedaży czy monitorowanie działalności konkurencji w celu identyfikacji przyczyn powodzenia konkretnych przedsięwzięć i elementarnych operacji i na jej podstawie podjęcie odpowiedniej reakcji. **Generowanie** wiedzy polega na odkrywaniu nowej wiedzy z wiedzy już posiadanej. Generowanie wiedzy odbywa się poprzez monitorowanie (sprawdzanie czy wartość parametru mieści się w założonym zakresie i podjęcie na podstawie tej wiedzy odpowiedniej decyzji), tworzenie (syntezę, analizę, konstruowanie), przesyłanie (w celu internalizacji). Podczas generowania wiedzy wykorzystuje się procedury, reguły wnioskowania. Wiedza powstała dzięki generowaniu posiada prostszą strukturę i pozwala na jej dalsze wykorzystanie. **Internalizacja** wiedzy polega na rozpowszechnianiu posiadanej wiedzy wśród właściwych pracowników. Internalizacja związana jest z identyfikacją potrzeb informacyjnych i decyzyjnych poszczególnych komórek organizacji a następnie filtrowaniu, opisywaniu, integrowaniu dostępnej wiedzy i przekazywaniu do (wewnętrznych) adresatów. Działania internalizujące oznaczają zmianę kultury organizacyjnej firmy, obejmują korzystanie z e-mail i hurtowni danych, udostępnienie danych i informacji, dzielenie się wiedzą, rozsyłanie wiadomości, dyskusje grupowe. Ważnym następstwem internalizacji jest wg Leonarda-Bartona [170] zwiększenie **skłonności do eksperymentowania i działań ryzykownych** podejmowanych przez pracowników posiadających bogatszą wiedzę i delegowane uprawnienia decyzyjne. Na etapie działań **eksternalizacyjnych** zaangażowana i udostępniana środowisku jest nie tylko wiedza, ale także zasoby materialne przedsiębiorstwa. Eksternalizacja polega na aktywnej komunikacji ze środowiskiem: obsłudze klienta, kontroli jakości, nadzorowaniu procesów wytwarzania i dystrybucji, prezentacjach, działalności reklamowej, marketingu. Każdy produkt wytwarzany i sprzedawany przez firmę (jego *know-how*, parametry techniczne, ilość, cena, miejsca sprzedaży itd.) jest efektem posiadanej wiedzy technologicznej, marketingowej, organizacyjnej itd. nabytej, uświadomionej, przetworzonej i wykorzystanej dzięki działaniom ujętym w pięciu grupach w *KCM*. **Dopelnieniem KCM** jest wyróżnienie czterech dodatkowych grup działań, niezależnych od procesu „wzbogacania wiedzy” i wspomagających wszystkie podstawowe opisane powyżej grupy działań: **pomiar wiedzy** (za pomocą np. efektów działalności), **kontrolowanie wiedzy** (sprawdzenie w jakim stanie są źródła wiedzy i jej

jakość: poprawność i przydatność), **koordynowanie wiedzy** (zarządzanie relacjami między źródłami wiedzy, czynnikami ludzkimi i fizycznymi zasobami firmy), **dowodzenie wiedzą** (ogólny nadzór nad wszystkimi grupami działań, wsparcie udzielane przez naczelne kierownictwo, tworzenie wizji organizacji, generowanie zmian, dobór personelu, wykorzystywanie kapitału intelektualnego).

3.8.5 Pozyskiwanie wiedzy w systemach eksperckich

Korzystanie z systemu eksperckiego odbywać się może na za pośrednictwem tzw. **środowiska rozwoju** (*development environment*), obsługiwane przez twórcę systemu eksperckiego (tzw. inżyniera wiedzy) w celu konstrukcji i konfiguracji oraz stosowania poszczególnych modułów systemu w procesie wprowadzania wiedzy eksperckiej do bazy wiedzy w procesie tzw. pozyskiwania wiedzy. Pozyskiwanie wiedzy (niezbędne, gdy mamy do czynienia z złożonymi problemami, dla których trudno – przynajmniej niespecjaliście - **zidentyfikować** zależności przyczynowo-skutkowe i **konstruować** modele matematyczne) może odbywać się poprzez wydobywanie wiedzy w drodze interakcji z ekspertem dziedzinowym (metoda wymagająca bezpośredniego, aktywnego udziału człowieka) oraz przez uczenie maszynowe (zautomatyzowane narzędzia, wśród których znane jest maszynowe uczenie za pomocą indukcji - np. tworzenie drzew decyzyjnych na podstawie dostępnych przykładów, metody dedukcyjne, konekjonistyczne, genetyczne i inne). Jak pisze Nguyen [216] w przypadku systemów eksperckich **pozyskiwanie wiedzy** odbywa się na trzy sposoby: **wywiadu**, **analizy protokołu** oraz **obserwacji**.

Analiza protokołu, wg Wrighta i Aytona [328] polega na wsłuchiwanie się w “swobodną wypowiedź” eksperta (oczywiście można ją także nagrywać). Wyróżnić można **dwa podejścia analizy protokołu**: równoczesna (*concurrent*), tożsama z obserwacją i retrospektywna (*retrospective*). Podejście równoczesne (z rozwiązywaniem problemu) polega na komentowaniu i tłumaczeniu przez eksperta wszystkich wykonywanych czynności a następnie wyodrębnianiu reguł i ich kodowaniu. Podejście retrospektywne polega na udziale eksperta w przesłuchiwanie (oglądaniu) nagranych czynności i „wymuszaniu” wyjaśnień dotyczących czynności i myśli (rozumowania) eksperta. Wadą tego podejścia jest możliwość pominięcia rzadziej wykonywanych czynności (procedur).

Prowadzenie wywiadu z ekspertem dziedzinowym może odbywać się na dwa sposoby: nieustrukturalizowany i ustrukturalizowany. Nieustrukturalizowanie wywiadu polega na pozyskiwaniu wszystkich informacji, na udzielenie których wyrazi zgodę ekspert. Wadą tego podejścia są **pokaźne rozmiary pozyskiwanej wiedzy** oraz konieczność jej gruntownej analizy i strukturalizacji w celu sformułowania reguł wnioskowania. Wywiad ustrukturalizowany polega na zadawaniu pytań wg opracowanego planu, aby bezpośrednio pozyskać reguły wnioskowania. Czasami wywiad ustrukturalizowany jest poprzedzany wywiadem nieustrukturalizowanym. Wadą wiedzy pozyskanej w trakcie prowadzenia wywiadów jest jej **spekulatywny charakter** (ekspert musi przypominać sobie

swoje postępowanie z przeszłości a przede wszystkim prowadzić eksperymenty myślowe). Do metod pozyskiwania poprzez wywiad zaliczyć można także **analizę kwestionariuszy**. Według Bryanta [29] kwestionariusze są metodą przydatną do pozyskiwania wiedzy subiektywnej: ekspert wyraża swój stosunek do ważności danej informacji na skali ważności czy za pomocą wag, sporządzając rankingi. Kwestionariusz jest skuteczny w pozyskiwaniu i strukturalizowaniu wiedzy o wskaźnikach jakościowych i ilościowych rozpatrywanych podczas analizy wniosków kredytowych. Wright i Ayton [328] podkreślają także dużą przydatność **techniki wielowymiarowego skalowania** (*multidimensional scaling technique*). Polega ona na **poszukiwaniu relacji** (podobieństw) między obiektami poprzez wykonywanie form graficznych, rysunków a przede wszystkim tzw. *scattergraphs*, na których przedstawiane są poszczególne obiekty występujące w teorii lub wyrażane w trakcie wydobywania doświadczenia eksperckiego. Jeśli między obiektami na skali zauważamy małą odległość, możemy podejrzewać podobieństwa i relacje między nimi, co pozwala na uogólnianie. Znaną techniką wielowymiarowego skalowania jest tzw. *card sorting* (sortowanie kart), polegające na wyszukiwaniu kandydatów na uznane reguły na podstawie bliskości poszczególnych nazw obiektów, doświadczeń, reguł (zapisanych na oddzielnych kartkach). Wśród **reguł sortowania** Turban [301] wymienia: wydzielanie grup (*group separation task*), tworzenie grup (*group creation task*), porównanie trójkami (*triadic comparisons*). Postępowanie zgodne z **wzorcem wydzielania grup** polega na klasyfikacji przez eksperta karteczek z pojęciami na dwie grupy, wymieszania ich wielokrotnym powtarzaniu tej procedury klasyfikacji (już przy innym kryterium). **Tworzenie grup** polega na wyodrębnieniu par kartek a następnie dobieraniu „para do pary” (podobnych) semantycznie par karteczek. **Porównywanie trójkami** polega na wyróżnieniu w każdej losowo powstałej trójki dwu kartek o bardziej zbliżonym znaczeniu i jednej o znaczeniu mniej zbliżonym, tworzeniu z nich grup i ich etykietowaniu (nadawaniu nazw).

3.8.6 Rola inżyniera wiedzy w zadaniu akwizycji wiedzy

Zadaniem **eksperta dziedzinowego** jest wyrażenie wiedzy zastosowanej w konkretnej sytuacji społecznej i poziomie technologicznym, nabytej przez eksperta z różnorodnych źródeł [320]: **podręcznikowej** (zawierającej wieloletnie osiągnięcia danej dziedziny nauki), **doświadczenia** pochodzącego z praktyki, **wiedzy ogólnej** i **zdroworozsądkowej**. **Pozyskiwanie wiedzy** odbywa się za pośrednictwem **inżyniera wiedzy**, często nowicjusza [160] w danej dziedzinie. Stąd zdobywanie wiedzy przez twórcę systemu eksperckiego wymaga dużo czasu i odbywa się poprzez samodzielne **studia** i **obserwacje**, polegające na zbieraniu wiedzy podręcznikowej, faktograficznej, analiz a przede wszystkim na interpretacji wypowiedzi ekspertów o dziedzinie problemowej, wysłuchaniu lub obserwowaniu sposobów podejmowania diagnoz i decyzji a następnie **uporządkowaniu**

(kategoryzacji, filtrowaniu, odrzuceniu fragmentów zbędnych i niespójnych – najlepiej przez opracowane reguły kontrolujące [104] spójność danych, mieszczące się dane w określonym zakresie, kompletność danych, wartości domyślne, funkcje konwersji), **konceptualizację** zgodnie z ustaloną ontologią umożliwiającą **formalizację** i **zakodowanie wiedzy** [302]. Mimo tych wysiłków trudno zapewnić kompletność bazy wiedzy zapisanej w postaci reguł: może ona nie zawierać wszystkich sytuacji wyjątkowych i fragmentów trudniej poddających się matematycznemu zdefiniowaniu. Dlatego **reprezentacja wiedzy** powinna w jakimś stopniu być **zgodna ze sposobem**, który **ludzki ekspert** wykorzystuje w celu **przechowywania swojej wiedzy**. Należy także określić czy ekspert opracowując wyniki analiz korzysta z pozyskanych wartości parametrów, podstawiając je do znanych sobie wzorów matematycznych czy też dokonuje wyboru spośród możliwych odpowiedzi (np. na podstawie kryteriów podejmowania decyzji).

Środowisko rozwoju pozwalające na prowadzenie pozyskiwania wiedzy przez inżyniera wiedzy, udostępniać powinno [7] za pośrednictwem specjalizowanego języka (pomocniczego względem języka opisu wiedzy) następujące usługi: **edytor tekstu** (do wprowadzania poleceń oraz aktualizacji bazy wiedzy), **interpretację poleceń na bieżąco**, **system pomocy** (wyjaśnianie składni i semantyki poleceń języka reprezentacji wiedzy), **narzędzia wspierające wyszukiwanie błędów** w bazie wiedzy (wizualizacja uruchamianych reguł, dopisywanych faktów, możliwości wstrzymania na żądanie wykonywania procedur wnioskowania), **bibliotekę przypadków i przykładów** (przydatnych na etapie walidacji bazy wiedzy), animację (jedną z metod wizualizacji), **sterowanie procesem wnioskowania** (wybór strategii wnioskowania, kierowanie propagacją ograniczeń, ustalanie priorytetów reguł). Inżynier wiedzy podczas **sesji z ekspertem** może stosować szereg **technik prowadzenia wywiadu** w celu wydobywania informacji zarówno w formie słownej (zapisywane w notatniku bądź edytorze tekstów) jak i nietekstowej (nagrania dźwiękowe, wideo, notatki na tablicy czy rysunki na kartkach). Obejmują one **techniki konwersacyjne**, polegające na skłonieniu eksperta do samodzielnego opisu jego zachowania **oraz techniki obserwacyjne** polegające na obserwacji przez inżyniera wiedzy rozwiązywania przez eksperta rzeczywistych problemów lub sytuacji symulowanych. Rezultatem stosowania technik konwersacyjnych są **opisy tekstowe**, natomiast technik obserwacyjnych **tekstowe protokoły** i **nietekstowe reprezentacje wiedzy**. Istnieją także techniki wielowymiarowe, w których korzysta się ze **środków niewerbalnych** [32]. Zwykle dąży się do wyrażenia wiedzy za pomocą języka łatwego w technicznym przetwarzaniu. Jeśli baza wiedzy rozrasta się szybko i staje się coraz bardziej złożona należy zapewnić możliwość podjęcia decyzji, jaki problem system ekspercki ma aktualnie rozwiązać, z jakiej wiedzy skorzystać oraz spośród jakich strategii i metod rozwiązywania problemu można wybierać. Już Feigenbaum [82] zauważył, że sposoby **pozyskiwanie wiedzy** są jedną z **głównych przyczyn trudności** w rozwijaniu systemów wiedzy (*KBS*). Mamy tu do czynienia z problemem **wąskiego gardła**, gdyż występują trudności w wydobywaniu wiedzy od ekspertów dziedzinowych a także w jej strukturalizacji (poprzez

wykorzystanie modeli problemu oraz języka modelowania) w taki sposób, aby reprezentacja wiedzy i techniki jej przetwarzania właściwie odzwierciedlały sposób rozumowania eksperta. Okazuje się bowiem, że **reprezentacja za pomocą reguł wnioskowania** trudno poddaje się zmianom, aktualizacji i ulepszeniom [54] oraz że występują trudności w reprezentacji i wizualizacji za pomocą **reguł opisów obiektów** (klasy, atrybuty, wartości) występujących w dziedzinie problemowej oraz związków między nimi. Wizualizacja procesu wnioskowania, jak twierdzą Selig i Johannes [264] jest możliwa dopiero wtedy gdy „uzyska się dostęp do implementacji języka mechanizmu wnioskowania”.

Jeśli zapoznamy się z **procesem wnioskowania** wykonywanego przez **ludzkiego eksperta**, to możemy zauważyć, że składa się on z dwu etapów: **identyfikacji** i **prognozy**. Jak podaje Cairo [32] **identyfikacja** składa się z diagnozy, weryfikacji, określenia współzależności, monitorowania oraz klasyfikacji. Celem **identyfikacji** jest przyporządkowanie systemu do określonej klasy i następnie zbadanie jego zachowania i stanu (wartości atrybutów). **Prognoza** zajmuje się określeniem jak system zachowa się w bliższej lub odleglejszej przyszłości i dotyczy przewidywania typu zachowania (zmian stanu) i jego atrybutów.

Jak zauważa Eriksson [76] **pozyskiwanie wiedzy wymaga modelowania** tzn. wydobycia wiedzy od eksperta i zapisania jej w sposób zrozumiały dla systemu informatycznego. Jak twierdzi Kangassalo [140] **modelowanie** składa się z **analizy** i **syntezy pojęć** należących do rozważanego zakresu wiedzy a następnie opracowania reprezentacji w postaci modelu konceptualnego. **Proces modelowania** zawiera fazę weryfikacji i walidacji modelu poprzez określenie poziomu zgodności z rzeczywistością, czyli w tym wypadku pokrywania się decyzji proponowanych przez system wiedzy (*KBS*) ze zdaniem ludzkiego eksperta. Porównanie to można wykonać na podstawie czynników ilościowych i jakościowych [32]. **Walidacja ilościowa** korzysta z metod statystycznych (np. analizy wariancji) podczas porównywania wyników uzyskiwanych przez *KBS* ze znanymi wynikami wypracowanymi przez ekspertów. **Walidacja jakościowa** zwraca się ku ocenie subiektywnej ale bogatej w formalizmy: test Turinga, walidacja podsystemów, ocena prognoz.

Nie można stwierdzić, że etap weryfikacji i walidacji musi zakończyć się **przed** wdrożeniem systemu z bazą wiedzy (*KBS*). Na przykład Chander i inni [38] uznają, że *KBS* należy **ustawicznie rozwijać zamiast konstruować** do chwili uznania za **system kompletny**. Czasami wyróżnia się w strukturze systemu eksperckiego **moduł** (podsystem) **doskonalenia wiedzy** (obejmujący także aktualizację wiedzy, niezbędną po uprawomocnieniu się nowych przepisów prawnych, zmianie zasad funkcjonowania obiektu gospodarczego, wreszcie uwzględniania preferencji i poglądów decydenta). **Walidacja bazy wiedzy** prowadzona jest w trzech etapach: **testowania**, **pielęgnacji** (*refinement*) i **aktualizacji** (*maintenance*) Testowanie obejmuje swoim zakresem wykrywanie i usuwanie strukturalnych i funkcjonalnych niezgodności z opracowanym projektem. Cele procesu *refinement* to wg Zlatarevy [341] poprawa a następnie utrzymanie poziomu wsparcia dla podejmowania decyzji jakie przynosi system *KBS*. Prawidłowo pielęgnowana baza wiedzy posiada następujące cechy: brak

niespójności i niekompletności w odniesieniu do wynikowych hipotez opisujących dziedzinę problemową oraz prawidłową klasyfikację przypadków testowych. *Maintenance* to niezbędna aktualizacja i uzupełnianie bazy wiedzy wraz ze zmianami w dziedzinie problemowej. Vamthienen i Wets [312] wskazują, że większość **problemów weryfikacyjnych i walidacyjnych** systemów z regułą reprezentacją wiedzy można rozwiązać **transformując bazę wiedzy** do postaci **tablic decyzyjnych** a następnie wykonując operacje prowadzące do uzyskania **kompletności, spójności i poprawności** bazy wiedzy. Oprócz trywialnych przypadków, funkcjonowanie każdego *KBS* **można poprawić** przede wszystkim podczas projektowania i implementacji systemu. Etap ten obejmuje **rewizję** istniejącej bazy wiedzy na podstawie obserwowanych strukturalnych bądź funkcjonalnych niedomagań systemu poprzez korekcję osiąganych dzięki *KBS* konkluzji. W tym celu modyfikuje się reguły zawarte w bazie wiedzy zmieniając logikę zasad uogólniania i uszczegółowiania pojęć, przesłanek i konkluzji. Według Zlatarevy [341] procedura rewizji bazy wiedzy obejmuje sprawdzenie występowania wszelkiego rodzaju błędów. Jeśli występują trudności w przygotowaniu odpowiednio dużej liczby przypadków testowych, zalecane jest wykorzystanie **przykładów testowych** zastosowanych na etapie konstruowania bazy wiedzy, w sytuacji małej liczby przypadków testowych także należy przeprowadzać rewizję. Należy także zlokalizować źródła zmniejszenia efektywności bazy wiedzy, którymi są głównie **reguły nadmiarowe** oraz **tworzące zamknięte koło**. Wykonywana w ten sposób **reorganizacja (restrukturyzacja)** bazy wiedzy nie powoduje zmian w wyciąganych konkluzjach, **zmienia** natomiast **przebieg ścieżki wnioskowania**. Po przeprowadzeniu restrukturyzacji baza wiedzy charakteryzuje się całkowitym **brakiem alternatywnych ścieżek** – wyjaśnień prowadzących do osiągnięcia danej konkluzji. Dzięki temu **zmniejsza się czas potrzebny do otrzymania konkluzji**, osiąga się **pożądany determinizm** między danymi (faktami) a wynikami (konkluzjami) oraz zapewnia **kompletność bazy wiedzy** we wszystkich gałęziach drzewa wnioskowań. Restrukturyzacja bazy wiedzy wiąże się z jej dekompilacją lub konstruowaniem nowej bazy wiedzy na podstawie pełnego przeglądu wniosków pochodzących ze starej bazy wiedzy. Operacje pielęgnacyjne typu *refinement* przeprowadzane są **okresowo podczas eksploatacji systemu** i polegają na wyszukaniu i usunięciu przyczyn niedomagań w działaniu bazy wiedzy: powtarzających się danych (powodują powstawanie podobnych konkluzji, zaciemniających interpretacje raportu), reguł wywołujących sprzeczności (powodują nieprzydatność zestawu konkluzji dla podejmowania decyzji) oraz nadmiarowych i zagnieżdżonych reguł (które pogarszają efektywność mechanizmu wnioskowania korzystającego z bazy wiedzy, zwiększają czas uzyskania (odpowiedzi)). Wyróżnia się dwa rodzaje pielęgnacji i aktualizacji: wykonywane w celu zachowania zgodności odwzorowania (*update*) i wykonywane w celu zwiększenia dokładności odwzorowania (*upgrade*). Aktualizacja w celu **zachowania zgodności odwzorowania (update)** dotyczy działań podejmowanych w całym cyklu projektowania (życia) *KBS* w celu zachowania na założonym poziomie zgodności bazy wiedzy wraz ze zmianami w dziedzinie problemowej (obiekcie). Sygnałem do wykonania aktualizacji typu *update*

jest zaobserwowanie niespójności nowych przypadków z już obecnymi w bazie wiedzy. Bez aktualizacji typu *update* model, którego walidacja przebiegła pomyślnie, mógłby utracić swoje własności z powodu zmian np. w otoczeniu i przestać odwzorowywać rzeczywisty system. Aktualizacja w celu **zwiększenia dokładności odwzorowania** (*upgrade*) jest **nieprzerwanym procesem** wykonywanym przede wszystkim w trakcie eksploatacji *KBS* i związanym z pozyskiwaniem nowej wiedzy. Baza wiedzy poddana temu rodzajowi aktualizacji charakteryzuje się brakiem redundancji, zawierania się pojęciowego (*subsumptions*) i prawidłowym klasyfikowaniem nowych przypadków. Należy zauważyć, że aktualizacja typu *upgrade* pokrywa się z zadaniem pozyskiwania wiedzy z obserwacji nowych przypadków. Dokonując **aktualizacji bazy wiedzy**, czyli uzupełnienia i wymiany elementów bazy należy uwzględnić fakt, że niekontrolowane dołączanie wszystkich nowych faktów i reguł może skutkować niespójnością bazy wiedzy. **Niektóre reguły** powinny być traktowane jako **podstawowe dla teorii** dotyczącej **dziedziny problemowej** i **chronione przed zmianami** (wywołanymi przez reguły uogólniające, zawsze ze współczynnikiem pewności mniejszym od 1, wyprowadzone na podstawie obserwacji). Meyer i Guessoum [197] proponują zastosowanie **strategii konceptualnego filtrowania**, polegającej na aktualizacji dokonywanej tylko po uzgodnieniu z kontekstem i konfiguracją ustaloną przez użytkownika bazy wiedzy.

Środowisko konsultacji (*consultation environment*) będące wynikiem pracy inżyniera wiedzy, jest obsługiwane za pomocą systemu menu, ikon graficznych przez użytkownika nie-eksperta który zainteresowany jest **konsultacją, czyli wydobyciem i zastosowaniem wiedzy zakodowanej w systemie eksperckim**. Obok tradycyjnego, **konwersacyjnego trybu korzystania z *KBS***, charakteryzującego się pasywnym udostępnianiem narzędzi i metod z dziedziny zarządzania i badań operacyjnych zauważyć można nieco odmienny, **przyszłościowy tryb dialogu z *KBS*** uwzględniający możliwość krytykowania przez system idei i działań użytkownika. To drugie podejście, **akcentujące motywowanie i aktywizację użytkownika** zostało opisane w pracy Angehrna [10]. Oprogramowanie pełniące **rolę asystenta menedżera** (środowiska konsultacji) w realizacji eksperckich funkcji powinno posiadać następujące cechy: **zdolność do komunikowania się z aplikacjami i użytkownikiem**, korzystania z **funkcji innych aplikacji** (wiedza o pozostałych modułach systemu), rozumienie **wymagań i zadań zleczanych przez użytkownika** (uwzględnianie możliwości zmiany preferencji i zmian otoczenia - adaptacyjność) oraz interaktywnego działania (kooperacji) poprzez dialog z użytkownikiem (korzystanie z podpowiedzi i udzielanie sugestii), w tym samodzielne definiowanie (dokonywanie wyboru metody rozwiązania) i prowadzenie (bez ciągłego nadzoru użytkownika) odpowiednich działań (autonomiczność). **Środowisko konsultacji** wypełnia wtedy **zadania zarządzania wiedzą** wśród których wyróżnić można [14]: (1) **tworzenie** (przez zbieranie i przetwarzanie danych) i **rozsyłanie** informacji, (2) **zwiększanie zrozumienia** dla informacji wśród jej użytkowników (menedżerów), (3) **wartościowanie** (filtrowanie) zbioru informacji w celu wyodrębnienia informacji istotnych (ale szczegółowych), **przydatnych** w zarządzaniu organizacją, (4)

rejestrowanie, przechowywanie i **przetwarzanie wiedzy** na przyszłe potrzeby.

System ekspercki jest zatem **interfejsem człowieka do zasobów wiedzy**, sposobem zastępowania żywego, ludzkiego eksperta oraz filtrowania, systematyzacji i przetworzenia do postaci użytecznej w działaniu (operacjonalizacji) takiej wiedzy, która zwykle nie jest przechowywana w postaci umożliwiającej jej natychmiastowe wykorzystanie (system ekspercki może być pośrednikiem między operatorem, użytkownikiem systemu informatycznego a bazą danych posiadających znaczne rozmiary, utrudniające efektywne korzystanie z niej). **Pośrednictwo człowieka** (eksperta oraz inżyniera wiedzy) w procesie pozyskiwania wiedzy umożliwia uzyskiwanie zadowalających rozwiązań w sytuacjach charakteryzujących się niewielką liczbą przykładów lub trudnościami w ich właściwej formalizacji i zakodowaniu dla potrzeb uczenia maszynowego. Wiedza poddawana transformacji do bazy wiedzy w systemie eksperckim charakteryzuje się **brakiem jednorodności**: mogą to być rejestracje lub **opisy zachowań** i typowych sytuacji (bazy danych), **poglądów** (różni eksperci, będący profesjonalistami, mogą preferować różne metody rozwiązywania tego samego problemu), **fakty** charakteryzujące się **różnym stopniem pewności** a nawet **niekiedy sprzeczne** ze sobą, **reguły** i **fakty zdroworozsądkowe**, **semantyka** zależna od kontekstu, **metareguły** (opisują charakterystyki dostępnych reguł wnioskowania, definiują prawa dostępu do poszczególnych reguł, określają źródła na podstawie których konstruuje się reguły wnioskowania, dokumentują strukturę powiązań między regułami i danymi) oraz **metody uczenia się** tzn. utrzymania aktualności wiedzy (metawiedza).

3.8.7 Moduł objaśniający w systemach eksperckich

Moduł objaśniający w systemie eksperckim dostarcza odpowiedzi na pytania: Jak?, Dlaczego?, Dlaczego nie? Co się stanie, jeśli? Który? itp. Pytanie **Jak?** pozwala na **prześledzenie ciągu reguł** prowadzących do osiągnięcia badanej konkluzji, co pozwala określić jakie atrybutów obiektów były badane i jaka jest wrażliwość rozwiązania na zmiany warunków wyjściowych. Pytanie: **Dlaczego?** umożliwia wykonanie raportu o **przesłankach prowadzących do uzyskanej konkluzji** – zwraca uwagę na konieczność istnienia w bazie wiedzy rozważanych obiektów oraz podkreśla rolę wartości ich atrybutów. Przesłanki (część formuły tworzącej regułę, pełniącą rolę warunku) mogą składać się z zestawu warunków logicznych. Konkluzja może mieć postać wywołania pojedynczej procedury lub zestawu procedur (np. przygotowujących opis problemu, przypisujących wartość atrybutowi obiektu, generujących nowy obiekt). Odpowiedź na pytanie: **Co się stanie, jeśli?** możliwa jest wtedy, gdy system nie tylko pamięta ścieżkę wnioskowania (kontekst) ale także umożliwia powrót do węzła w ścieżce wnioskowań aby **prześledzić ewentualne zmiany konkluzji** spowodowane modyfikacją atrybutów problemu. Odpowiedź na pytanie: **Dlaczego nie?** służy wykazaniu miejsca w drzewie

wnioskowania (reguły wnioskowania) **odpowiadającego za wyeliminowanie ze zbioru alternatyw** badanej konkluzji, przesłanki lub obiektu. W ten sposób można dowiedzieć się, w jakich warunkach możliwe było osiągnięcie danej konkluzji (rozszerzenie *backward chaining*). Do zanegowania oczekiwanej konkluzji według Metzlera [195] prowadzić mogą następujące przyczyny: (1) wybrano obiekt z danej klasy, ale różniący się wartością atrybutów; (2) brak obiektu, którego dotyczy pytanie w bazie wiedzy lub brak odpowiedniej reguły, (3) obiekt poddany został usunięciu lub modyfikacji, (4) mimo że reguła dotycząca obiektu istniała, to inny obiekt (zawarty w innej regule) był odpowiedniejszy, (5) obiekt występował w regule, ale nie spełniał ograniczeń (przesłanki reguły fałszywe - brak możliwości uaktywnienia reguły).

3.8.8 Agenda a architektura tablicowa

Agenda jest miejscem poprzez które poszczególne **moduły** (źródła) wiedzy mogą się **komunikować między sobą**. Agenda nie dostarcza zadaniom informacji, w jaki sposób wykonywane są poszczególne zadania, kiedy i jak będą wykonywane. Poszczególne zadania oddziałują natomiast na siebie dopisując (gdy w ciągu operacji wnioskowania zachodzi taka konieczność) i usuwając się nawzajem (jeśli nadane zostały im takie uprawnienia) z listy (agendy). Klasyfikacja topologii struktur agend dotyczy kolejności rozpoczynania zadań znajdujących się w agendzie i obejmuje agendy z priorytetami, agendę o liniowej strukturze oraz agendę zarządzaną zgodnie poziomami priorytetów. **Agenda z priorytetami** charakteryzuje się pierwszeństwem wykonywania zadań posiadających najwyższe priorytety a jej wadą może być to, że istnieje niebezpieczeństwo niewykonania zadań o niskim priorytecie. W **agendzie o liniowej strukturze** zadania uporządkowane są wg posiadanych priorytetów i wykonywane od najwyższego do najniższego priorytetu, nawet jeśli w trakcie wykonywania zadań o stosunkowo niskim priorytecie nadejdzie zadanie o priorytecie wysokim, to oczekuje ono aż wszystkie zadania występujące w poprzednim stanie agendy zostaną wykonane. Natomiast trzeci typ agendy, **obsługującej priorytety**, różni się od agendy o strukturze liniowej tym, że zadania organizowane są w pakiety posiadające ten sam poziom priorytetu i nowe zadanie, jeśli posiada wyższy priorytet czeka na wykonanie dopóki wszystkie zadania z rozpoczętego pakietu nie zostaną wykonane. Jeśli chodzi o zadania posiadające niski priorytet, to najszybciej rozpocznie się ich wykonywanie w systemach z agendą typu liniowego, następnie agendy z poziomami priorytetów a najwolniej w agendzie zarządzaną zgodnie z priorytetami.

Według Hayes-Roth [112] **architektura tablicowa** pozwala na wykorzystanie w jednym systemie eksperckim **różnych rodzajów wiedzy** zmniejszając w ten sposób rozmiar przestrzeni poszukiwań. Odbywa się to na dostępnym dla wszystkich źródeł wiedzy (modułów) wspólnym obszarze nazywanym **tablicą**, w którym uzyskiwane są rozwiązania. **Źródło wiedzy** można także

rozważać jako agenta (czyli system, który odbiera informacje z otoczenia, reaguje na te informacje i samodzielnie podejmuje odpowiednie działania aby wypełnić swoje zadanie) zdolnego do rozwiązania problemu [32]. Zdaniem Churchmana [42] w przypadku **wielu źródeł wiedzy źle ustrukturalizowanej** i szybko zmieniającej się, przydatne są dwa sposoby pozyskiwania wiedzy: kantowskie i heglowskie. **Podejście kantowskie** polega na zebraniu wielu formalnych poglądów o uzupełniającej się naturze. Podejście to nie zdaje jednak egzaminu w sytuacji wielu możliwych postępowań (działań, wniosków) wynikających z wielości uzupełniających się rozwiązań (prawda jest względna). Dopuszcza się wyprowadzenie różnych rozwiązań (na podstawie różnych teorii) z tych samych danych. System wspomagania decyzji może wtedy jedynie przedstawić zestawienie proponowanych rozwiązań. Podejście kantowskie, **oparte na reprezentacji**, nie uznaje możliwości pojawienia się konfliktów i nie przewiduje dodatkowych reguł. Rozstrzygnięcie pozostawia się decydentowi. Można także wykorzystując **alternatywne teorie** dopasować tak dane wejściowe, aby rozwiązania były tożsame [69]. Wydaje się że do tej pory w przypadku (co najmniej) większości systemów eksperckich korzystano bezkrytycznie z tego podejścia. Jak wskazuje Radosiński [246] „proces budowy systemów eksperckich cechuje silny indywidualizm metodologiczny, objawiający się w arbitralności przyjmowanych założeń konstrukcyjnych”. Potwierdza także, że za pomocą systemów eksperckich (nawet zbudowanych na bazie tego samego „szkieletu” *ES shell*) korzystających z baz wiedzy opracowanych z udziałem różnych ekspertów dziedzinowych uzyskać można różne wskazówki i rady. Dzieje się tak dlatego że każdy ekspert, posiadając odmienne doświadczenia, poglądy i *odmienną osobowość* będzie tworzył odmienne bazy wiedzy (choć niekoniecznie sprzeczne). **Zobiektywizowanie baz wiedzy i procesu wnioskowania** (czy przez architekturę tablicową czy równoległe struktury hybrydowe) jest **próbą pokonania subiektywizmu i indywidualizmu metodologicznego** obecnego w systemach eksperckich. **Podejście heglowskie**, oparte na dialektyce, zakłada syntezę wielu całkowicie przeciwstawnych (opozycyjnych, antytetycznych) poglądach. Synteza polega na przedstawieniu wielu (być może różnych) rozwiązań uzyskanych **zgodnie z wieloma teoriami** z tych samych danych oraz opracowaniu nowej teorii będącej uogólnieniem (na wyższym poziomie abstrakcji) obu teorii. Z drugiej wymaga **wielokrotnego rozważania** danego problemu, przeprowadzenia **szeregu eksperymentów** i modyfikowania poglądów. W tym kontekście Radosiński [246] podkreśla że obiektywne wnioski dotyczące badanego systemu gospodarczego (przedsiębiorstwa) uzyskamy nie wtedy gdy będziemy polegali na szerokiej i szczegółowej bazie wiedzy wygenerowanej przy udziale pojedynczego eksperta, a na uznanym i wielokrotnie weryfikowanym (podręcznikowym) zestawie reguł wnioskowania o sytuacji ekonomicznej przedsiębiorstwa.

3.9 Wnioskowanie oparte na przykładach. Ramy

Case-Based reasoning (CBR) według Lawa i innych [160] jest metodą rozwiązywania problemu (dla określonego, bieżącego przypadku) polegającą na **wzorowaniu się na rozwiązaniach** dostępnych dla jak **najbardziej podobnych przypadków**, które wystąpiły już w przeszłości. W ten sposób dąży się do **opracowania modelu predykcyjnego** (opisującego za pomocą wyrażeń matematycznych obserwowane zdarzenia lub prognozującego zdarzenia przyszłe na podstawie określonych założeń) cechującego się zmniejszaniem niepewności dotyczącej przyszłości. Metoda CBR oparta jest na **obserwacji rozumowania eksperta**, który w trakcie procesu rozwiązywania problemu **sięga pamięcią** (oczywiście także wykorzystując posiadane komputerowe bazy danych, kartoteki papierowe czy zapiski) do przeszłości i przypomina sobie znane mu problemy, prezentuje je (sobie w umyśle lub na forum) oraz wzoruje się na ich rozwiązaniach. Jest to **rozumowanie przez analogie** i oparte na **wyszukiwaniu precedensów**. W taki sposób od dawna wykonują swoją pracę prawnicy i lekarze a także planujący strategie w przedsiębiorstwach. Jest to podejście odmienne od regułowej reprezentacji wiedzy: tu **przechowuje się i przetwarza dane strukturalizowane** w postaci przypadków. **Zaletami CBR** jest łatwiejsze niekiedy pozyskiwanie wiedzy i jej pielęgnacja (utrzymanie) oraz możliwość efektywnego korzystanie z baz danych (baz przypadków).

Samodzielne techniki poszukiwania rozwiązania wykorzystywane w *CBR* (algorytmy w rodzaju najbliższego sąsiada) **nie są jednak zadowalające** (wymagają dużo pamięci, niezbędne jest przeprowadzenie selekcji atrybutów i opracowanie funkcji podobieństwa), baza przypadków może nie spełniać warunku dostatecznej systematyczności i dlatego zwykle *CBR* **łączone są z innymi metodami** rozwiązywania problemów, polegających w większym stopniu na teoretycznych podstawach rozważanego zagadnienia: reguł wnioskowania, logiki rozmytej. Przechowujące **wiedzę proceduralną i deklaratywną** dziedzinowe bazy wiedzy wykorzystywane w systemach eksperckich (regułowego wnioskowania) uzupełniane są zatem o wiedzę faktograficzną, epizodyczną w postaci przypadków strukturalizowanych za pomocą ram. **Pojęcie ramy** zostało zdefiniowane w 1975 przez Minsky'ego [201] jako struktury danych której zadaniem jest **reprezentacja powtarzalnych sytuacji**, czynności, stanów, rzeczy. Jest to pojęcie zbliżone znaczeniowo do *schemata* systemu filozoficznego Kanta [141]. **Elastyczna struktura ramy** pozwala na zawarcie w niej opisu informacji o sposobach obsługi ramy (jako całościowej struktury), przewidywań stanu przyszłego ramy, wymaganiach ramy dotyczących dostarczenia niezbędnych do funkcjonowania ramy informacji. Ramy są przykładem **obiektozoorientowanej reprezentacji wiedzy** (innym są sieci semantyczne), dobrze nadającej się do przedstawiania problemów klasyfikacji, natomiast gorzej do przedstawiania relacji przyczynowo-skutkowych, wnioskowania dedukcyjnego. Rama może występować jako **definicja pewnej klasy obiektów** lub jako **egzemplarz obiektu** należącego do danej klasy. Każda rama **składa się z pól** zawierających **nazwę kategorii** przypadku oraz tzw. *slots* (gniazd, szczelin) zawierających **nazwy** i

wartości atrybutów danego egzemplarza: faktów i procedur. Jest to tzw. **reprezentacja intencjonalna**, polegająca na wyróżnianiu spójnych i oszczędnych (jeśli chodzi o wielkość opisu) pojęć. Gniazda mogą zawierać **wartości domyślne** a także **dopuszczalne zbiory wartości**, które może przyjmować atrybut. Połączone ze sobą ramy tworzą system ram, będący reprezentacją czynności, relacji przyczynowo-skutkowych, a także pozwalający na przedstawienie wariantowego postrzegania pojęć i rzeczy symbolizowanych przez ramy. Minsky [201] wyróżnia **poziomy ram**, związane z **lingwistyczną złożonością treści** reprezentowanych przez ramę. **Ramy widzialnej syntaktyki** (*Surface Syntactic Frames*), określają struktury wiążące czasowniki i rzeczowniki, użycie przymiotników i reguły składniowe opisują kolejność słów. **Ramy widzialnej semantyki** (*Surface Semantic Frames*) określają znaczenie słów przejawiają się w działaniu – dynamice, uczestnikach, narzędziach, trajektoriach, strategiach i ich celach, efektach ubocznych. **Ramy tematyczne** (*Thematic Frames*) przedstawiają plany działań z wyszczególnionymi tematami, postępowaniem, planowaniem strategii i problemami związanymi z ich wykonywaniem, prowadzą do uzyskania odpowiedzi na pytania o oddziaływanie informacyjne na użytkownika, sposoby uzupełnienia wiedzy i możliwości wykorzystania w rozwiązywaniu problemów. **Ramy narracyjne** (*Narrative Frames*) stosowane są w reprezentacji kolejnych etapów wydarzeń z przeszłości, wyjaśniania. Wspierają konstruowanie i wypełnianie ram tematycznych poprzez przydatność w wyrażaniu opinii, ukierunkowania uwagi i odpowiedzi na pytania o sprawcę (agenta), zamiary (intencje), skutki, beneficjenta (adresata) oraz wykorzystywane narzędzia/technologie). Języki służące opisowi ram (*KRL* i *FRL*) **definiują nazwy poszczególnych klas, zbiór ich atrybutów** (gniazd) oraz **usytuowanie względem innych ram** (klas) – np. **dziedziczenie, zawieranie się**.

Sieć semantyczna obejmuje wszystkie pojęcia teorii i bazę wiedzy. Manipulacji w niej mogą podlegać tylko ustalone pojęcia, ich atrybuty oraz pojęcia niższego rzędu, pochodzące od tych pojęć (zasada dziedziczenia). Sieci semantyczne są jedną z form obiektowej reprezentacji wiedzy. Składają się z węzłów połączonych łukami, reprezentującymi relacje (np. semantyczne lub regułowe). Wiedza zakodowana w postaci sieci semantycznych ma charakter **deklaratywny** i przedstawia graficzne relacje (dwuargumentowe) łączące poszczególne obiekty. Języki reprezentacji wiedzy korzystające z relacji semantycznych wyposażone są w środki służące ich modelowaniu typu **relacja dziedziczenia, zawierania się** czy **implikacji** za pomocą wyrażeń typu: „tworzy”, „jest częścią”, „pozwala na”, „składa się na”. Jak wynika z powyższego opisu, koncepcje wykorzystane w reprezentacji za pomocą sieci semantycznych **dają się przetworzyć do zapisu w języku logiki predykatów**, wymuszając niejako zapis zdań orzekających zawierających słownictwo pochodzące z języka naturalnego.

Ramy i sieci semantyczne wykazują **podobieństwo** także jeśli chodzi o **sposoby korzystania ze zgromadzonej wiedzy**, wynikające ze struktury reprezentacji wiedzy: **przeszukiwanie hierarchiczne** (lokalizacja zbioru obiektów wraz z odpowiadającymi im procedurami spełniających

warunki poszukiwań lub rokujących przydatność w rozwiązywaniu danego problemu), **rozpoznawania wzorców** (lokalizacja obiektów/ram/predykatów, które można uznać za - do pewnego stopnia - zgodne z założonymi wzorcami, opisanymi a pomocą atrybutów i ich wartości). **Rozszerzenia ram**, języki obiektowo zorientowane, udostępniają dodatkowe środki opisu obiektów, np. za pomocą klauzul hornowskich. **Definicje obiektów** zawierać mogą także pola danych określane jako prywatne, tzn. dostępne tylko dla procedur (metod) związanych z daną klasą obiektów. Nazywane jest to **mechanizmem enkapsulacji** i pozwala na dokonywanie zmian w reprezentacji wiedzy faktograficznej (hierarchia dziedziczenia obiektów) bez konieczności przeprojektowywania części proceduralnej (polimorfizm), dokonującej operacji na obiektach. **Obiektowo zorientowana reprezentacja wiedzy** udostępnia także mechanizmy definiowania relacji między obiektami. Mogą to być relacje typu zawierania się (*implicit*) które polegają na przedstawianiu powiązań bezpośrednio za pomocą atrybutów obiektów będących ze sobą w relacji, tzn. że odpowiednie pole obiektu A zawiera wartość wskazującą jednoznacznie na obiekt B. Relacje mogą być także przedstawiane w postaci dodatkowej oddzielnej struktury grupującej identyfikatory powiązanych ze sobą obiektów. Jest to jawne (*explicit*) definiowanie relacji, pozwalające na dostęp wszystkich klas do informacji dotyczącej charakteru powiązań.

3.10 Tablice decyzyjne

Tablice decyzyjne (*decision tables*) są konkurencyjną względem reguł wnioskowania (*production rules, clauses*) formą reprezentacji wiedzy. Najważniejsza różnica dotyczy formatu zapisu. W tablicach decyzyjnych wiedza zakodowana jest w nich w postaci funkcji której argumentem jest produkt kartezjański wartości (stanów) warunków służących do wyznaczenia decyzji a wartością (przeciwdziedzina) jest produkt kartezjański wariantów decyzyjnych. Jak wykazali Boryczka i Słowiński [24] istnieją metody wydobywania reguł decyzyjnych z tablic decyzyjnych za pomocą aplikacji zbiorów przybliżonych. W pracy [44] wyszczególniono następujące **cztery elementy tworzące tablicę decyzyjną**:

1. Określenie warunku służącego do wyznaczenia wariantu decyzyjnego (którego wybór jest zalecany)
 2. Opis znaczenia i kontekstu warunku – zapisywane po lewej stronie tablicy, kolejno od góry do dołu
 3. Konkretny warunek zapisany np. w postaci wyrażenia arytmetycznego, generujące pewien podzbiór wartości warunku (zjawiska) – zapisywane po prawej stronie tablicy, kolejno od góry do dołu – odpowiednio do (1); (3) po lewej stronie, bezpośrednio wymienionych uprzednio warunkach umieszczone są identyfikatory (opisy) poszczególnych wariantów decyzyjnych
 4. Po prawej dolnej części tablicy znajdują się zakodowane wartości reprezentujące związki (lub ich brak) między wartościami warunków (mogą być to koniunkcje warunków) i wariantów decyzyjnych.
- Vanthienen i Wets [312] na podstawie **kryterium wzajemnej wyłączości kolumn** wyróżniają dwa typy tablic decyzyjnych: *single hit*, w których nie ma dwu kolumn z takimi samymi wartościami zestawu warunków oraz *multiple hit*, w których występuje powyższa sytuacja (trzeba wtedy przyjąć określoną strategię wyboru wariantu). Wspomniani autorzy podają następujący **przepis na zbudowanie tablicy decyzyjnej**: (1) **ustalenie warunków**, przestrzeni ich stanów (np. prawda i fałsz) i zbioru wariantów decyzyjnych oraz ich **uporządkowanie**; (2) określenie **zbioru reguł wnioskowania** (zapisanych w obowiązującym formacie) – dzięki wyrażeniom opisanym w punkcie (1) nim łączą się w relacje; (3) wypełnienie tablicy decyzyjnej na podstawie zbioru reguł wnioskowania, poczynając od **wprowadzenia warunków** (przesłanek) a **kończąc na wariantach decyzyjnych** (konkluzji) – liczba wierszy tablicy decyzyjnej może być większa niż liczba reguł (gdy złożona reguła zawiera sumę logiczną warunków rozdziela się ją na odpowiednią liczbę reguł prostych); (4) **sprawdzenie** (i ewentualna rewizja) **tablicy** ze względu na **kompletność** (wszystkie kombinacje stanów warunków mają przyporządkowaną konkluzję), **spójność** (wszystkie kombinacje stanów warunków mają przyporządkowaną dokładnie jedną konkluzję) oraz **poprawność** (postępowanie weryfikacyjne mające na celu ustalenie zgodności między wynikową tablicą decyzyjną oraz przekonaniem i zamierzeniami projektanta); (5) **sprawdzenie** występowania **nadmiarowości** i

jej **likwidacja** (uproszczenie i zmniejszenie liczby wierszy i kolumn w tablicy). Dokonali także **klasyfikacji symptomów niezgodności** z postulatami kompletności, spójności i poprawności. Dzięki usunięciu poniżej wymienionych niedomagań można oczekiwać **rozwiązania większości problemów** weryfikacyjnych i walidacyjnych charakterystycznych dla systemów opartych na regułowej reprezentacji wiedzy: (1) pozbycie się **redundantnych reguł** (powielających się), zawierających się w sobie reguł, reguł które nie mogą być uaktywnione oraz posiadających identyczne konkluzje; (2) **usunięcie niespójności** (reguł posiadających identyczne przesłanki i różne konkluzje (sprzeczne lub niesprzeczne)) (3) **usunięcie reguł cyklicznych** (konkluzje jednej reguły są przesłankami drugiej); (4) **zapobieganie braku kompletności** (nieużywane, a zadeklarowane wartości atrybutów (zmiennych będących składnikami przesłanek) oraz nieosiągalne, a zadeklarowane konkluzje); (5) **zapewnienie poprawności** (badanie semantycznej zgodności osiągniętych konkluzji z teorią podejmowania decyzji w danej dziedzinie). Przykładem zastosowania tablicy decyzyjnej wykorzystującej logikę wielowartościową jest moduł reprezentacji wiedzy w systemie eksperckiego FINEVA [188].

3.11 Techniki odkrywania wiedzy w bazach danych

Przetwarzanie danych ekonomicznych jest wykonywane w celu analizy danych. Gdy menedżer **odkryje** i **zidentyfikuje zależności** w bazach danych (do których uzyskał dostęp), to może opracować na ich podstawie **decyzje** nawet w horyzoncie strategicznym. **Bazy danych** powstają w wyniku przechowywania informacji o poszczególnych transakcjach gospodarczych, klientach, dostawcach itd. Bazy danych są jednocześnie częścią systemów ewidencji zdarzeń gospodarczych – systemów sprawozdawczości. Interdyscyplinarne **przetwarzanie wielkich zbiorów danych**, będące kombinacją **metod statystyki** i **uczenia maszynowego** określane jest jako *Knowledge Discovery in Databases* (odkrywanie wiedzy w bazach danych). Przesłanką wykorzystania złożonych algorytmów przetwarzania wielkich ilości danych jest złożony charakter danych będących odzwierciedleniem intensywnej i rozległej terytorialnie działalności gospodarczej, w której codziennością jest są nieliniowe, obciążone zakłóceniami zależności między wartościami poszczególnych atrybutów (pól) w bazach danych.

Galindo i Tamayo [90] wyróżniają następujące trzy wzajemnie wspierające się czynniki sprzyjające udanemu **zastosowaniu interdyscyplinarnej metodologii KDD**: (1) **wdrożenie i wykorzystywanie wielkich baz danych** (prowadzonych w firmach i instytucjach publicznych) zawierających pełne informacje o wszystkich dokonanych transakcjach finansowych – należy tu wspomnieć o kluczowym znaczeniu hurtowni danych; (2) **rozwój metod obróbki statystycznej** (uogólnianie wniosków) i **technik uczenia maszynowego**, co umożliwi opracowanie bardziej zaawansowanych modeli i głębszą analizę złożonych zależności, (3) **rozwój technologii komputerowej** – szybsze przetwarzanie dzięki maszynom wieloprocesorowym, sieciom komputerowym i wielkim pamięciom masowym. *Data mining* lub *knowledge discovery*, jak podają Mejía-Lavalle i Rodríguez-Ortiz [193] jest **interdyscyplinarnym procesem** nietrywialnego **wydobywania** ze zbiorów danych **informacji ukrytych**, uprzednio nieznanych oraz charakteryzujących się **potencjalną użytecznością** (zdolnością do osiągania dalekosiężnych celów). *Data mining* wykorzystują różnorodne techniki wydobywania wartościowych informacji z użytkowanych przez przedsiębiorstwa baz danych lub hurtowni danych.

Odkrywanie informacji należy rozumieć jako identyfikowanie wzorców i regularności występujących w zbiorach danych. Rozszerzona definicja podana przez Frawleya i innych [88] dotyczy przede wszystkim aspektów wyjściowego formatu procesu *KDD*: mając zbiór faktów (danych) oznaczony przez D , język oznaczony jako L i jakąś miarę pewności C , wzorcem nazywamy zdanie S zapisane w języku L które opisuje relację między podzbiorem D_s zbioru ze współczynnikiem pewności C_s , taką, że S jest prostszy niż wykaz wszystkich faktów ze zbioru D_s . **Wiedzą nazywamy wspólnie** wszystkie wzorce które mają jakąś wartość (są godne zainteresowania) dla użytkownika i są wg niego dostatecznie pewne. **Wszystkie wzorce** spełniające powyższą definicję powstałe dzięki

działaniu programu analizującego dane nazywamy odkrytą wiedzą (DK – *discovered knowledge*). Venter i inni [313] wyróżniają (ze względu na udostępniane funkcje) **następujące etapy procesu KDD**: (1) **przygotowanie danych** (uzgodnienie formatów), usuwanie zakłóceń i uruchomienie usług hurtowni danych, (2) **eksplorację ukierunkowaną na dane** – ograniczenie ilości przetwarzanych danych poprzez ustalenie interesujących podzbiorów danych, OLAP – **odpowiedzi na pytania** w rodzaju: „co się stało i dlaczego”, dokonywanie cięć danych na warstwy, tworzenie zestawień, (3) **analizę wyników** przetwarzania danych (4) *data mining* (kopanie danych)– poszukiwanie **interesujących wynikowych wzorców** w bazie danych, posiadających odpowiedni poziom zgodności z wzorcami ustalonymi przez użytkownika, (5) **prezentację i przeglądanie** (nawigacja) **znalezionych wzorców** – interaktywne, graficzne przedstawienie struktury odkrytej wiedzy.

Wydobywanie (odkrywanie) **wiedzy** przeprowadzane jest w następujących etapach (wyróżnionych ze względu na zasadnicze procedury przetwarzania informacji): **selekcji** (decyzja o wyborze do analizy rozważanego zakresu – podzbioru danych i ich atrybutów), **wstępnego przetwarzania** (usunięcie zbędnych informacji – odszumianie, pozyskanie i umieszczenie danych zastępczych – likwidacja braków, normalizacja i standaryzacja; bez tego etapu istnieje zagrożenie wypracowania błędnej reprezentacji wiedzy, szczególnie w predykcyjnej *Data Mining* – wykorzystującej ciągi uczące i służącej przewidywaniu odpowiedzi systemu), **przetwarzanie do formatu zgodnego** z oprogramowaniem *KDD* oraz **właściwe data mining** – przeprowadzenie **klasyfikacji, grupowania i wizualizacji**.

Jak twierdzi Hurwicz [121] nawet najlepiej zaprojektowane i najszybciej działające oprogramowanie będzie nieprzydatne, gdy dane będą nieadekwatne do stanu rzeczywistego. Wyróżnić można **wiele przyczyn** odpowiedzialnych za **błędy** znajdujące się w danych. Są to np. błędy człowieka wprowadzającego dane, wieloznaczności treści wprowadzanych do bazy danych, powielanie tych samych danych, niewłaściwe zaprojektowanie zestawu pól w rekordzie danych. Zawierające **olbrzymie ilości danych hurtownie danych** ze względu na swój rozmiar tym bardziej narażone są na występowanie błędnych danych. Dlatego też hurtownie danych wymagają nieraz **wykonania kilkuset przebiegów** (wykorzystuje się w tym celu oprogramowanie czyszczące), eliminujących takie przeszkody jak błędne lub nieużyteczne dane, wykonujących korekcję błędnych danych które dają się naprawić, standaryzację danych (np. uwzględnienie stopy inflacji i jej zmian, zastosowanie jednakowych jednostek miar). Narzędzia *data mining* służą uzyskaniu odpowiedzi [277] na pytania: „co jest interesujące”, „co może się zdarzyć”, a wynikiem ich działań są **modele predykcyjne**. Zazwyczaj *data mining* dotyczy określenia **ukrytych trendów** lub **typów zachowań** na podstawie **dużej ilości danych** (uboczne następstwo rozwoju informatycznych technik przechowywania i przetwarzania danych oraz wzrostu aktywności gospodarczej) przechowywanych w **wielowymiarowych bazach danych**. Wynikiem tych wysiłków jest **reprezentacja wiedzy** w postaci relacji logicznych, drzew decyzyjnych, reguł wnioskowania. **Odkrywana** jest w ten sposób **nieznana**

wcześniej wiedza, która będzie pomocna w ulepszaniu aktualnie stosowanych procedur. Co więcej, **wydobywanie wiedzy** czy informacji pomocnych w prognozowaniu odbywa się bez konstruowania złożonych zapytań zakodowanych w języku manipulacji danymi.

Weiss i inni [321] przyjmując jako kryterium klasyfikacji **metody odkrywania regularności** w zbiorach danych wyróżniają następujące grupy **metod odkrywania wiedzy**:

1. **Tradycyjne** metody statystyczne np. analizę regresji, techniki liniowej, kwadratowej i logistycznej dyskryminacji, analizę MANOVA
2. Nowoczesne **metody statystyczne**: najbliższego sąsiada, analizę czynników niezależnych (*projection pursuit*) -polegającą na wyszukiwaniu takiego rzutowania z wielo- na małowymiarową przestrzeń które wskaże najwięcej szczegółów dotyczących danego zbioru danych, a następnie wydobyciu ich (powierzchni, klastrów itp.) i osobnym analizowaniu
- (3) **Drzewa decyzyjne** oraz metody indukcyjne z regułową reprezentacją wiedzy (systemami eksperckimi) i algorytmy takie jak CART - identyfikujące struktury obecne w danych
- (4) **Sieci neuronowe** i podobne metody
- (5) **Wnioskowanie bayesowskie** i sieci bayesowskie.

Należy zauważyć, że różnica między *data mining* a analizą danych polega na tym, że **analiza danych** zajmuje się **opracowaniem i zastosowaniem metod wyszukiwania informacji**, dążąc do jak **najszybszego wykonania zadania** – odpowiedzi na pytania w rodzaju „ile wynosi?” „jaka część z całości?”, natomiast *data mining* ukierunkowane jest na **identyfikację** w bazach danych **nieznanych uprzednio regularności**, trendów, tendencji i wzorców zachowań. Analiza danych wykonywana jest z wykorzystaniem **instrukcji języków dostępu do baz danych** (np. *SQL*), zazwyczaj wyposażone w graficzny interfejs (menu typu „wskaz i kliknij”, przyciski ekranowe ułatwiające wyszczególnienie warunków, kryteriów grupowania tworzących strukturę zapytań do baz danych) czyli generatory odwołań do baz danych. Narzędzia *data mining* korzystają z wszystkich udogodnień dostępnych w generatorach odwołań do baz danych, wyposażone są także w instrukcje pomocniczego przetwarzania danych, służące **konstruowaniu powtarzalnych** (tworzących standard dla konkretnego przedsiębiorstwa) **procedur** sporządzania i formatowania raportów i wszelkiego rodzaju porównań (także statystycznych). Korzystają zatem z usług udostępnianych przez narzędzia typu OLAP, których zadaniem jest dostarczenie odpowiedzi na pytania w rodzaju „co się stało w ostatnim okresie?” i „dlaczego się stało?”, bazując na dokładnym rozpoznaniu modelu danych i wymagań użytkownika - menedżera.

3.12 Algorytm genetyczny

Algorytm genetyczny [339] jest **techniką poszukiwania rozwiązań** wykorzystującą stochastyczną zmienność w celu **symulacji mechanizmów doboru naturalnego** i rozmnażania się istniejących w świecie biologii. Algorytm genetyczny jest zatem uogólnieniem i przeniesieniem zaobserwowanych mechanizmów biologicznych na grunt modeli matematycznych w celu optymalizacji wyrażeń opisujących systemy nawet całkowicie odmienne od organizmów biologicznych. **Koncepcję** zastosowania **teorii ewolucji w modelach systemów sztucznych** opracował Holland [118]. Podstawą algorytmów genetycznych jest teoria ewolucji, zakładająca przetrwanie najlepiej przystosowanego osobnika i podobieństwie „jakości” potomstwa do „jakości” genów rodziców. Mimo że **nie zawsze** można przeprowadzić dowód optymalności rozwiązania znalezionej dzięki zastosowaniu algorytmu genetycznego, to uważa się je często za przydatne [41]. Postępowanie zgodnie z zasadami algorytmu genetycznego polega na **przeszukiwaniu kolejnych** (następstwo czasowe) **populacji** poprzez dokonywanie **losowego wyboru osobników** (reprezentantów) do **rozmnażania się**. Podobnie jak w przyrodzie, proces ten odbywa się przez **rekombinację genotypów** (czyli materiału genetycznego) prowadzącą do powstania nowych osobników (a właściwie genotypów). Dodatkowo zapewnia się możliwość występowania **zmienności cech** u potomstwa poprzez losowe występowanie **zjawiska mutacji**. Użytkownik lub programista wykorzystujący lub tworzący oprogramowanie implementujące algorytm genetyczny **ustawia następujące parametry**: rozmiar populacji, częstość rekombinacji, częstość mutacji. Ogólne skonfigurowanie algorytmu obejmuje sposób kodowania osobników (reprezentujących konkurujące ze sobą rozwiązania problemu), rodzaj rekombinacji (rozmnażania się) oraz działanie operatora mutacji. Wśród zalet algorytmu genetycznego Ho i Lee [116] wymieniają jego **solidność** i **wytrwałość**, natomiast za wady uznają problemy z **prawidłowym zakodowaniem opisu** w postaci chromosomu oraz zapewnieniem względnie **szybkiego wyliczenia wartości funkcji przystosowania** (funkcji oceny).

Obserwacje dotyczące świata istot żywych posłużyć mogą do opracowania koncepcji modyfikacji algorytmu genetycznego np. **rozproszonych algorytmów genetycznych** (*distribute genetic algorithms*) w których zapewnia się zróżnicowanie osobników na **kilka populacji** (o różnych warunkach początkowych, parametrach i wzorcach krzyżowania się i mutacji), początkowo odizolowanych od siebie w okresie przed inicjacją głównego algorytmu genetycznego, a w zasadniczej części algorytmu dopuszcza się do kontaktów prowadzących do rekombinacji. Podstawą rozproszonego algorytmu genetycznego jest operator migracji, opisany przez prawdopodobieństwo migracji. **Migracja** działa w następujący sposób: dokładna kopia chromosomu najlepiej przystosowanego osobnika z każdej podpopulacji z pewnym prawdopodobieństwem w każdym pokoleniu zastępuje najgorszego osobnika z innej, także losowo wybranej podpopulacji. Odpowiednio

niska wartość prawdopodobieństwa migracji pozwala uniknąć **przedwczesnego ujednoczenia** całej populacji.

Algorytm genetyczny składa się z kilku etapów (kroków). **Początkowym etapem** jest wygenerowanie (losowe – np. rozkład równomierny, za pomocą określonego programu/algorytmu lub zaproponowane przez użytkownika) populacji zawierających zbiór reprezentacji rozwiązań problemu. Poszczególne rozwiązania **kodowane** są za pomocą łańcucha binarnego i **odwzorowują wektor współczynników** o wartościach należących do zbioru liczb rzeczywistych [41]. Wszystkim rozwiązaniom przyporządkowuje się **funkcję przystosowania** (*fitness*; minimalizującą lub maksymalizującą), której argumentem są wartości wygenerowane za pomocą modelu matematycznego rozważanego zagadnienia. Niektóre warianty algorytmu genetycznego **normalizują aktualną populację** poprzez uwzględnienie w funkcji przystosowania **ekstremum pokolenia** oraz rozbudowują ją o składnik odwzorowujący karę za rozwiązanie niedozwolone (np. przekraczające ograniczenia jeśli problem optymalizacyjny je posiada). Następnie uruchamiane są operacje polegające na **selekcji** (prawdopodobieństwo wyboru jest funkcją wartości przystosowania), **krzyżowaniu** (*crossover*) i **mutacji** (skutkującej zmianami zbioru reprezentacji, czyli uzyskania nowego chromosomu) prowadzące do powstania nowego pokolenia.

Celem operacji **selekcji** jest zapewnienie **przekazywania informacji** o „dobrej” jakości rodziców na następne pokolenie. Dlatego **chromosom** (reprezentujący osobnika) wykazujący wyższą wartość funkcji przystosowania ma **większą szansę wyboru na rodzica**. Należy zwrócić uwagę na występowanie w literaturze dwu odmiennych znaczeniowo terminów *performance* i *fitness*. W algorytmie genetycznym wyliczana jest wartość *performance* (wartość dopasowania) dla poszczególnych chromosomów która następnie jest wykorzystywana dla porównań chromosomów w danym pokoleniu (jest to jednak wartość bezwzględna i można ją wykorzystać dla porównań międzypokoleniowych np. w kryterium stopu). Natomiast wartość *fitness* (wartość przystosowania) **jest wynikiem normalizacji** wartości *performance* (np. w zakresie 0-1) w obrębie danego pokolenia. Na podstawie wartości *fitness* można bezpośrednio wyznaczyć symulowaną szansę wyboru na rodzica nowego chromosomu.

Operacja **krzyżowania** polega na pobraniu chromosomów od rodziców, rozcięciu ich w punkcie (lub w wielu punktach) nazywanym *cross-over* oraz **dopasowania** rozciętych części w celu **utworzenia nowego chromosomu** będącego **krzyżówką** chromosomów obu rodziców (jeśli poszczególne geny w chromosomach obu rodziców nie są jednakowe, to chromosom potomka jest różny od chromosomu danego rodzica w tych miejscach – genach – które odziedziczył po drugim rodzicu). Podczas krzyżowania, algorytm **pobiera chromosomy** od osobników wybranych za rodziców i rozcina je w punkcie *cross-over*. **Dwa nowe chromosomy** będą wynikiem dopasowania pierwszej części od pierwszego rodzica i drugiej części od rodzica drugiego.

Zastosowanie operatora **mutacji** jest podobne do krzyżowania. Jego celem jest **utworzenie**

nowego, zmodyfikowanego chromosomu czyli potomka. Operacja mutacji polega na wybraniu jednego z genów a następnie **zmianie** (zwykle losowej) jego wartości. **Porównując** operacje **krzyżowania** i **mutacji** widzimy, że nowy chromosom w przypadku krzyżowania zależy od wartości genów rodziców (kwestia dotyczy tylko miejsca wymiany wartości genów) natomiast dzięki mutacji powstaje chromosom posiadający gen różny od genu rodzica. Dzięki mutacji zapewniane jest różnicowanie populacji chromosomów.

Nowe pokolenie składa się z rozwiązań (osobników) które pomyślnie przeszły test przeżycia (prezentowały odpowiednio wysoką wartość funkcji przystosowania). Jeśli nowe pokolenie zawiera także osobniki z poprzedniego pokolenia (rodziców), to można dodatkowo wykonać **operację wyboru** (*cull*) najlepiej przystosowanych i usunięcia identycznych osobników aby **liczebność populacji** (niejednakowych osobników) **była stała**. Proces ten jest kontynuowany (powtarzany) np. do chwili znalezienia zadowalającego rozwiązania. **Genotyp** zapisywany jest jako chromosom (reprezentujący zestaw cech kandydujących do rozwiązania zadania), w którym zakodowane są binarnie ciągi (zestawy) wartości parametrów reprezentujące kandydata do rozwiązania problemu.

Chromosom składa się z **genów** (kodujących poszczególne cechy). Gen może posiadać wiele wartości, nazywanych **allelami**. Chromosom reprezentowany jest przez ciąg (b_1, b_2, \dots, b_m) gdzie b_1 itd. nazywane są **allelami** czyli liczbami zwykle zerojedynkowymi (uwaga: możliwe jest zastosowanie bogatszego „alfabetu” np. od 0 do 9, co jest rozwiązaniem bardziej naturalnym – szybszym i łatwiejszym w realizacji informatycznej - jeśli dotyczy reprezentacji parametrów problemu opisanych liczbami rzeczywistymi), gdyż kodowanie binarne jest najbardziej wydajne, jak pisze Goldberg [98] jeśli chodzi o liczbę ciągów przy tej samej liczbie bitów informacji. **Zestawy genów** (zakodowane parametry rozwiązania) mogą reprezentować całościowe struktury, które **należy optymalizować** (np. w optymalizacyjnych problemach kombinatorycznych jest to reprezentacja bezpośrednia, bez dodatkowego przekodowywania). Pewna **wada reprezentacji** chromosomów przez liczby binarne – brak własności polegającej na tym, że dwa punkty leżące blisko siebie jeśli chodzi o przestrzeń problemową są także blisko siebie w przestrzeni reprezentacji, może być „zmniejszona” poprzez zastosowanie **kodowania Gray’a**, w którym dwa punkty bezpośrednio ze sobą sąsiadujące w przestrzeni problemowej różnią się tylko wartością jednego bitu. W ten sposób nieznaczna mutacja genu (na jednej pozycji bitowej) nieznaczną zmianę wartości reprezentowanej przez chromosom w dziedzinie problemowej. Procedury przekształcające kod binarny w kod Gray’a znaleźć można w pracy Michalewicza [198].

Operator **selekcji** dokonuje **wyboru** chromosomów, których geny i wartości *alleli* mają stanowić podstawę dla operatora mutacji i które mogą zostać powielone w następnym pokoleniu. Można zaproponować **wiele różnych postaci operatora selekcji** – np. mechanizm ruletki czy preferowanie chromosomów charakteryzujących się wyższą funkcją przystosowania (mechanizm rankingu). Mechanizm pierwszego rodzaju wykorzystuje analogię ruletki, której obwód reprezentuje

całość populacji, natomiast liczba gniazd przydzielona danemu chromosomowi jest proporcjonalna do wartości funkcji przystosowania (wyższa wartość funkcji przystosowania skutkuje większą liczbą przypisanych gniazd a stąd większym prawdopodobieństwem wylosowania).

Operator krzyżowania (*cross-over*) zwany także **operatorem rekombinacji** oznacza połączenie części dwu chromosomów (lub ich kopii) w celu powstania nowego chromosomu (reprezentacji osobnika). Zakładamy zatem, że części uznanych za korzystne (na etapie selekcji) chromosomów, po wymieszaniu się utworzą nowy, **co najmniej tak samo korzystny chromosom** (nowego osobnika, dziedziczącego korzystne cechy (geny-allele) rodziców). Operator rekombinacji (seksualnej) może działać w następujący sposób: tworzone są dwie struktury chromosomowe (bez alleli), następnie kopiowane są geny z identycznymi allelami z chromosomów rodziców, na końcu odbywa się wypełnienie pozostałych genów wybranym allelami jednego z rodziców (rodzica losuje się każdorazowo). Inna postać operatora rekombinacji polega na zwiększaniu liczebności populacji, np. przez generowanie jednocześnie kilku potomków, z których część jest kopią rodzica a reszta wynikiem rekombinacji.

Bez uwzględniania operatora mutacji (aseksualnej) materiał genetyczny reprezentowany w populacji nie tylko nie wzbogacałby się, ale groziłaby mu **stopniowa utrata zróżnicowania**, prowadząca do wyradzania się. **Operator mutacji** dokonuje co pewien (losowo wygenerowany) czas **zmiany wartości** niektórych (także losowo wygenerowanych) **genów**. W ten sposób populacja może być uzupełniona o chromosomy zawierające wartości genów które były nieobecne w poprzedniej populacji (nigdy nie występowały lub zostały zagubione podczas procesu reprodukcji). Mutacja jest reprezentowana przez następującą procedurę: po pierwsze **sprawdza się** czy wylosowano (zgodnie z założonym prawdopodobieństwem) wartość odpowiadającą za uruchomienie mutacji; jeśli nie to chromosom nie ulega zmianie, jeśli tak, to **wartość genu (allel) przyjmuje losową wartość** (inną niż oryginalna). Można założyć, że **częstość mutacji** zmniejsza się wraz z pozycją w rankingu osobników [294].

Generyczne programowanie genetyczne (*GGP*) stanowi **połączenie** programowania genetycznego [154] i indukcyjnego programowania logicznego [245]. *GGP* zajmuje się wykorzystaniem oceny (ewaluacji) programów komputerowych rozwiązujących problemy. *GGP* używane jest w celu zwiększenia szybkości maszynowego uczenia się lub stopniowej poprawy jakości programów opracowanych w różnych językach programowania. **Podstawą działania GGP** jest **zdolność interpretacji** źródłowego kodu programu w języku gramatyk logicznych (*logic grammars*). Jest to język posiadający **własne symbole** (podobne nieco do stosowanych w Prologu), **termy** (zmienne logiczne, funkcje, wartości stałe), **zmienne** (oznaczone znakiem „?”), **argumenty** (reprezentujące zależność od kontekstu – w trakcie wykonywania programu, a także **semantykę procedur** poprzez struktury drzewiaste). Jako przykład reprezentacji wiedzy dziedzinowej za pomocą języka gramatyk logicznych Wong [327] podaje następującą regułę gramatyczną, określającą że

użyteczny program składa się z dwu użytecznych komponentów:

program-jest-użyteczny \rightarrow *pierwszy-komponent(?X), {jest-użyteczny(?X, ?Y)}, drugi-komponent(?Y)*.

Początkowa populacja dla *GGP* może być opracowana w sposób stosunkowo łatwy, gdyż *GGP* pozwala na wykorzystanie programów napisanych w różnych językach. W następnych krokach algorytmu wykonywane są takie operacje jak **reprodukcja**, **mutacja** i **krzyżowanie**. Wynikiem działania programu *GGP* jest **struktura drzewiasta** przedstawiająca kolejne etapy prowadzące do wyłonienia z formalnej gramatyki logicznej **użytecznego wynikowego programu**. *GGP* może być łączone z indukcyjnymi algorytmami pozyskiwania wiedzy, takimi jak *AQ15* [199]. *GGP* przeprowadza **ewaluację poszczególnych zestawów reguł** proponowanych przez **algorytm indukcyjny**, kontrolując **agresywność** i pozwalając na ominięcie lokalnych minimów.

3.13 Czynniki wpływające na powodzenie SWD

Dopiero w ostatnich czasach [106] doświadczalnie ustalono zestaw czynników wpływających na powodzenie (sukces) SWD. **Założono ich niezależność** aby można je było stosować także w projektach BPR. **Wyróżniono następujące czynniki:** zadowolenie użytkownika, wsparcie udzielane przez kierownictwo, doniosłość problemu, subiektywną trudność problemu, cechy osobowe inżyniera wiedzy wdrażającego SE, cechy osobowe użytkownika końcowego, charakterystyka narzędzia budowy SE, zaangażowanie użytkownika.

Brak zadowolenia użytkownika (*user satisfaction*) z SE skutkuje niekorzystaniem z niego, czyli bezużytecznością. Gupta i Montazemi [107] uważają, że dla użytkownika atrakcyjność narzędzia wiąże się ze zdolnością adaptacji narzędzia do poglądów decydenta na naturę problemu, czyli subiektywnego zachowania decyzyjnego użytkownika. **Wsparcie udzielane przez kierownictwo** (*managerial support*) dotyczy użytkownika, rozwoju i utrzymania SE. **Doniosłość problemu** (*problem importance*) dotyczy powiązania SE z celami strategicznymi firmy – klasyczne pozytywne przykłady to *Autorizer's Assistant* zastosowany w rozważaniu wniosków kredytowych przez *American Express* oraz *XCON* wspomagający kluczowy dla ówczesnej firmy *DEC* problem konfiguracji zamawianego komputera zgodnie z wymaganiami klienta. **Subiektywna trudność problemu** (*problem difficulty*) – jeżeli rozwiązanie problemu sprawia trudności użytkownikowi to można się spodziewać że wsparcie udzielane przez SE będzie docenione. **Cechy osobowe inżyniera wiedzy** wdrażającego SE (*developer characteristics*) to przede wszystkim zdolności i umiejętności związane z pozyskiwaniem wiedzy. **Cechy osobowe użytkownika końcowego** (*end-user characteristics*) obejmują nastawienie (jeśli jest negatywne to poniesione koszty będą zmarnowane), oczekiwania wobec SE, umiejętności informatyczne, zaufanie względem SE, wyrażenie zgody na poznawanie zasad użytkowania systemu. Należy wspomnieć że istnieje niebezpieczeństwo wrogiego potraktowania SE przez użytkowników obawiających się zmiany oczekiwań względem wykonywanej przez nich pracy i niebezpieczeństw jej utraty (z powodu jej zastąpienia przez sztuczne systemy inteligentne). Wydaje się jednak że użytkownicy systemów inteligentnych powinni zostać zobowiązani do konsekwentnego poznawania systemu i zdobycia umiejętności umożliwiających korzystanie z ekspertyzy oferowanej przez SE w celu zwiększenia swojej wydajności pracy. **Charakterystyka narzędzia** budowy SE (*ES shell characteristics*) jest uwarunkowana rodzajem zadania wspieranego przez SE. W przypadku intensywnego wykorzystywania w procesie podejmowania decyzji zewnętrznych baz danych wymagany będzie **elastyczny interfejs** integrujący je z SE. W warunkach wielkiej liczby danych niezbędne będzie **przetwarzanie wstępne** opisane przez Famili i innych [79] i wykorzystanie **technologii hurtowni danych** [181] aby pokonać trudności związane z informacją nadmiarową (irrelevantną) oraz uwzględnić czynniki miejsca, czasu i okoliczności. Ważna staje się wtedy możliwość oszacowania czasu potrzebnego do uzyskania odpowiedzi (wyników) z SE. Oczywiście

jeśli nie przewiduje się korzystania z systemów zarządzania bazami danych to interfejs będzie funkcjonalnie ograniczony. Warto wspomnieć także, że cechą przyjazności dla użytkownika powinny mieć nie tylko interfejs inżyniera wiedzy ale także interfejs użytkownika końcowego. **Zaangażowanie użytkownika** (*user involvement*) jest na tyle ważne, że postuluje się sporządzanie i analizę notatek o zaangażowaniu pracowników w projekt zastosowania SE. Co więcej, uważa się że jeżeli **inicjatywa inwestowania** i projektowania SE wychodzi **bezpośrednio od użytkowników końcowych** a następnie obserwuje się aktywny ich udział w procesie wdrażania to spodziewany jest stosunkowo wysoki poziom zadowolenia z użytkowania SE. Szczególnie istotne wydaje się jak największe uczestnictwo użytkowników na etapach weryfikacji i walidacji projektu. Przyczyną jest większe prawdopodobieństwo wykrycia błędnie zdefiniowanego zagadnienia (problemu) biznesowego.

3.14 Przyczyny niepowodzeń i przesłanki postulatu hybrydowości

Do przyczyn niepowodzeń **systemów eksperckich**, i szerzej **systemów wspomagających zarządzanie wiedzą** [Liebowitz] [174] zalicza: (1) brak integracji strategii projektowania i wdrażania systemów zarządzania wiedzą ze strategicznymi celami organizacji, (2) brak aktywnego wsparcia ze strony naczelnego kierownictwa, (3) błędy w planie projektowania i wdrażania systemów zarządzania wiedzą. Pokrywają się one zatem ze społeczno-organizacyjnymi czynnikami sukcesu SE (ale w znaczeniu negatywnym).

Duchessi i O'Keefe [63] zdefiniowali organizacyjne niepowodzenie SE jako **brak korzystania z niego na co dzień** lub **brak wymiernych korzyści**: zmniejszenia kosztów, skrócenia cyklu produkcyjnego, zmniejszenia czasu potrzebnego na usunięcie awarii. Na podstawie ich badań przypadków udanych i nieudanych wdrożeń systemów eksperckich w tych samych firmach (General Electric, IBM, Eastman Kodak) można sformułować następujące wnioski. Systemy eksperckie który **poniosły porażkę** były zaprojektowane jako rozwiązania proste, implementujące pojedynczy sposób rozwiązania problemu. **Systemy które odniosły sukces miały złożoną strukturę, były przystosowane do rozbudowy, zaprojektowane zostały jako hybrydy (wykorzystujące także inne techniki poszukiwania rozwiązań niż systemy eksperckie).**

Zaniedbywanie identyfikacji i kontrolowania czynników wpływających na powodzenie zastosowania technologii informatycznych może prowadzić do trudności w ich wdrażaniu lub nawet odrzucenia przez członków organizacji. Proponowano zidentyfikować [124] wiele **różnych zestawów czynników**, często wybieranych na podstawie osobistych poglądów. Guimaraes i inni [105] twierdzą że barierą użyteczności SWD jest **brak wsparcia projektu** przez naczelne kierownictwo. Yoon i inni [332] prezentują **bardziej techniczny punkt widzenia** i uważają, że powodzenie zastosowania SWD wynika bezpośrednio z doświadczenia twórców systemu, poziomu technologicznego narzędzi (*ES shell*), cech użytkownika i jego zaangażowania. Guimaraes i inni [106] podają **przykłady niepowodzenia** systemów eksperckich i twierdzą, że wiele z nich spowodowana jest **pośpiechem i koniecznością zmian związanych szczególnie z projektami BPR.**

Jak twierdzą Subramanian i inni [287] na skuteczność zastosowania SE (w szerokim sensie znaczenia) znacząco wpływają: **metoda pozyskiwania wiedzy** wybrana do rozwoju (sformalizowane i niesformalizowane wywiady, kwestionariusze, śledzenie procesu pracy eksperta), **baza wiedzy** (reguły wnioskowania, sieci semantyczne, ramy), kolejno **wymagane i wykonane** w systemie **zmiany** (zmiany reprezentacji wiedzy, aktualizowanie). **Bieżąca** (krótkoterminowa) **skuteczność** zastosowania SE ustalana jest na podstawie informacji o **ekonomicznym wymiarze efektów** (wynikach finansowych, zyskach) sporządzonych przez użytkownika (można je przedstawić na skali), natomiast **skuteczność długoterminowa** ustalana jest na podstawie **liczby modyfikacji** systemu eksperckiego jak również zabiegów związanych z konserwacją systemu.

Wydaje się że **bardzo ważnymi czynnikami wpływającymi negatywnie** na powodzenie SE były zawsze brak niezbędności SE, możliwość rozwiązywania problemów bez SE, niewystarczająca doniosłość problemu (lub możliwość poniesienia znacznych strat w razie wykonania błędnej decyzji). **Niestety jednowymiarowe SE**, stosujące tylko pojedyncze techniki, nie miały szans wykazania efektu synergicznego i z czasem organizacje wracały do utartych procedur niekorzystających z SE. Kolejnym ważnym symptomem kojarzonym z implementacjami ponoszącymi porażkę był **brak możliwości prezentacji praktycznej**, biznesowej (efektów finansowych, przyspieszenia procesów wytwarzania, skrócenia przestojów) przydatności SE. Czynnikiem bardzo wyraźnie wpływającym na ocenę SE jako odnoszącego sukces czy ponoszącego porażkę była także możliwość szybkiej i **bezzstronnej weryfikacji zaleceń** wydawanych przez SE. Jednocześnie zaangażowanie SE do rozważania problemów, w przypadku których **potencjalna błędna decyzja** związana jest z **potężnymi stratami finansowymi** przy pojedynczej technice wyznaczania rozwiązania i braku możliwości weryfikacji skutków decyzji przed jej podjęciem (przypadek rozważania wniosków kredytowych), odstrasza pracowników finansowo odpowiedzialnych za decyzje. Wydaje się, że rozwiązaniem jest tu **równoległe wykorzystywanie** większej liczby **technik** oraz **weryfikacja symulacyjna**. **Zauważyć** można także fakt, że systemy które odniosły sukces były „od samego początku” projektowane do zastosowania w konkretnym środowisku (interpretacja danych i ustalenie ich kontekstu), przy założeniu **architektury modułowej** i **przystosowaniu do rozszerzeń funkcji**. Niestety systemy które poniosły porażkę charakteryzowały się **przeciwными cechami** (długi czas dostosowania do istniejących systemów informatycznych w przedsiębiorstwie).

Systemy eksperckie, które **poniosły porażkę**, cierpiały na **brak bezpośredniego finansowego wsparcia** przez kierownictwo. Mimo że członkowie ścisłego kierownictwa popierali projekty, to jednak **wyższe priorytety** przyznali innym zadaniom organizacyjnym. **Średni szczebel** kierownictwa zwykle popierał utrzymywanie SE, jednak **nie był bezpośrednio zaangażowany** i odpowiedzialny za projekty. Niekiedy (w przypadku systemów zdalnego wsparcia) członkowie kierownictwa nie byli przekonani o przydatności z informatyzowania procesów, byli **zwolennikami odpowiedzialności człowieka** i wspierali *status quo* (bezpośredni udział pracowników obsługi w rozwiązywaniu problemów). Zauważyć należy, że systemy które poniosły porażkę **załoga postrzegala z wyraźną niechęcią** i nieufnością, jako przedsięwzięcie które było inicjowane i podtrzymywane przez **personel techniczny** (zagrożenie zwolnieniami, utratą kompetencji) dla partykularnych interesów. W podobny sposób postrzegano nowe metody i narzędzia podejmowania decyzji, jako nierespektujące dotychczasowych procedur i dokonujące **zbędnych (subiektywnie) zmian organizacyjnych**. Przykładem może być system doradztwa przy usterkach wyposażenia przyjmowany który był **pozytywnie** przez **użytkowników urządzeń** (komputerów) natomiast z **wrogością** i niepokojem przez **członków personelu technicznego**.

Główną przyczyną niepowodzenia SE (rozumianego jako pozostanie w fazie prototypu), jak

twierdzi Shepherd [269] jest zbyt mała staranność przykładana do **czynników społeczno-organizacyjnych** na etapie implementacji. Należy zauważyć że same, choćby najlepiej oceniane, cechy systemu eksperckiego (zarówno zaawansowanie technologiczne i korzyści z jego stosowania) nie wystarczają, gdyż ważniejszy może być sposób w jaki SE jest **pryswajany przez działającą organizację**. Wiggins i inni [323] twierdzą, że przeciwnik wykorzystania systemu eksperckiego może stać się jego zwolennikiem gdy **uświadomi sobie**, że SE **zwiększa jego wydajność** oraz gdy projektant wziął pod uwagę raczej **sposoby wsparcia podejmowania decyzji** przez użytkownika niż najbardziej nawet wydajne algorytmy (chodzi o nie przeciwstawianie SE użytkownikowi. Zdaniem Bryanta [29] do **ekonomicznych i technicznych przyczyn niepowodzeń** SE należą: (1) błędny wybór problemu do rozwiązywania z wykorzystaniem SE, (2) przekroczenie spodziewanych kosztów i horyzontu czasowego wdrożenia, (3) błędna walidacja, spowodowana przekroczeniem kosztów i przede wszystkim opóźnieniem wdrażania SE, (4) ograniczenia systemu operacyjnego na platformie którego działa SE, (5) niespełnienie wymagań użytkownika. Podkreślić należy także znaczenie przyczyn o charakterze organizacyjnym: brak akceptacji użytkowników systemu, konieczność zmiany priorytetów organizacji i jej członków, niepokój dotyczący odejścia z pracy twórców i opiekunów systemu, problemy etapu przejścia od projektu do wdrożenia SE.

Wśród przyczyn niepowodzeń SE Lee i Kwon [164] wymieniają niewystarczające wysiłki dotyczące planowania rozwoju systemu aplikacji systemów eksperckich. Mimo **że nie ma dobrze sformalizowanych metod**, czyli że **nie można wygenerować planu** dla każdego konkretnego przypadku zastosowania SE, to jak twierdzi Turban [301] warto stosować następujące **sposoby planowania SE** (mimo ich fragmentaryczności):

1. **Lista krytycznych czynności** (*checklist*). Jak podają Lee i Kwon [164] za Prerau [238] można wprowadzić na listę nawet 200 punktów kontrolnych, odwzorowujących wszystkie najdrobniejsze etapy rozwoju SE. Największą użyteczność wykazuje podczas **kontroli** chociaż może stanowić obciążenie dla wykonawcy. Jej wadą jest **niewuwzględnianie relacji** między poszczególnymi podstawowymi etapami i czynnikami wpływającymi na ich realizację (chyba że przydzielili się wagi poszczególnym pozycjom i zaznaczy powiązania). Oprócz tego opracowanie listy kontrolnej **nie zapewnia sporządzenia planu** wdrożenia SE
2. Podejście polegające na posługiwaniu się **krytycznymi czynnikami powodzenia** (*critical success factors*) opiera się na zidentyfikowaniu kluczowych czynników powodzenia. Właśnie **identyfikacja i wybór czynników jest najtrudniejszy**. Podobnie jak poprzednia, metoda ta nie zapewnia dobrego ustrukturalizowania badanych elementów. Jeśli jednak uda się prawidłowo zidentyfikować czynniki to może być bardzo przydatna, szczególnie **gdy konieczne jest szybkie opracowanie działającego systemu**
3. Dostosowane do konkretnego środowiska **podejście strategiczne**. Polega ono na opracowaniu zestawu wskazówek wspomagających formułowanie reguł postępowania. Według Lee i Kwon [164]

wskazówki opisywane w literaturze są jednak zbyt ogólne aby mogły znaleźć praktyczne zastosowanie

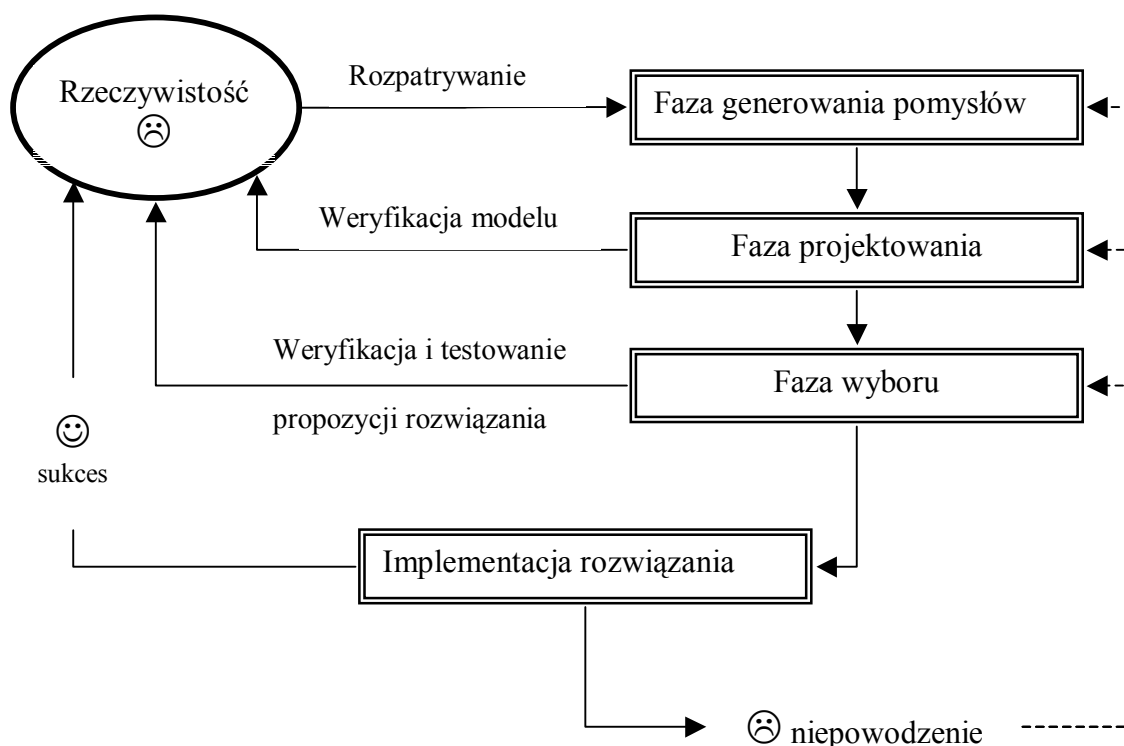
4. **Studia przypadków.** Podejście to można traktować jako **jedynе przydatne źródło doświadczeń** gdy nie jesteśmy w stanie odkryć i sformułować zasad planowania za pomocą trzech poprzednich podejść. Studia przypadków polegają na **wyczerpujących opisach** pewnej ograniczonej (od jednego do kilku) liczby przypadków na podstawie opracowań własnych i doniesień literaturowych. Jeśli da się opisać w sposób formalny wymagania względem planu rozwoju i poszczególnych przypadków to można próbować automatyzacji za pomocą techniki *case-based reasoning*.

Kolejnym czynnikiem, który może przyczynić się do niepowodzenia SE jest wadliwie zaprojektowany i wykorzystywany **komponent pozyskiwania wiedzy**. Subramanian i inni [287] uważają komponent pozyskiwania wiedzy za **potencjalne wąskie gardło** w rozwoju SE gdyż tylko prawidłowe przeniesienie wiedzy eksperta do systemu komputerowego może zapewnić funkcjonowanie SE zgodnie ze standardami działania eksperta-człowieka. Jeśli metody pozyskiwania wiedzy będą źle zaprojektowane, to wg Turbana [300] staną się barierą dla zakodowania użytecznej wiedzy posiadanej przez eksperta. Wśród elementarnych przyczyn niepowodzeń w zakresie pozyskiwania wiedzy wymienia się [287]: **problemy w komunikacji werbalnej z ekspertem, różnice kulturowe** między ekspertem a inżynierem wiedzy, problemy z uzyskaniem spójności między poglądami różnych ekspertów, trudności z uświadomieniem sobie przez eksperta sposobu rozumowania.

3.15 Projektowanie SWD

Projektowanie SWD ma na celu [36]: (1) strukturalizację procesu podejmowania decyzji, która umożliwia zastosowanie narzędzi analitycznych w generowaniu wariantów decyzyjnych; (2) ułatwienie zintegrowanego korzystania z poszczególnych narzędzi analitycznych w procesie tworzenia, konfigurowania, oceny i modyfikacji wariantów decyzyjnych; (3) udostępnienie elementarnych czynności analitycznych, takich jak pozyskiwanie, przetwarzanie, prezentacja, manipulowanie danymi w różnych przekrojach.

Konstruowanie SWD posiadającego złożony, konfigurowalny, intensywnie wykorzystujący grafikę interfejs, współpracującego ze współczesnymi technologiami informacyjnymi, takimi jak np. hurtownie danych, *OLAP*, zdalny dostęp za pośrednictwem usług i protokołów internetowych (*Web*, *http*, *CGI*, *PHP*) wykorzystującymi języki manipulacji danymi (*SQL*) jest **skomplikowanym zadaniem analitycznym** [270], wymagającym doświadczenia i opracowania odpowiednich metodologii. Polegają one na implementowaniu dostępnych szablonów (np. posługiwaniu się technologiami graficznego projektowania interfejsu użytkownika, komunikowania się za pomocą sterowników zewnętrznych systemów zarządzania bazami danych), **integracji różnorodnych technik informatycznych** (architektury klient-serwer, wywoływania zewnętrznych procedur analitycznych i statystycznych, *data mining*, systemów bazowanych na wiedzy, sztucznej inteligencji) wreszcie technikami z obszaru inżynierii oprogramowania (modelowania struktur danych, projektowania obiektowego itp.). Schemat cyklu życia SWD przedstawiono na rysunku [Rys. 3-5].



Rys. 3-5 Cykl życia SWD i motywacje działań podczas budowy SWD. Źródło: [303]

Przed **technologią systemów inteligentnych** [270] stoi zadanie przetworzenia i prezentacji kierownictwu przedsiębiorstwa „powodzi” danych, informacji, doświadczeń, prognoz, sprawozdań pochodzących z wielu różnorodnych źródeł. Zadanie powyższe realizowane jest poprzez kontrolowanie, przesiewanie, interpretowanie informacji (*data mining*) w celu poradzenia sobie z zalewem nadchodzących danych. Konieczne jest także efektywniejsze wykorzystanie systemów wspomagania decyzji prezentujących przetworzone informacje (w tym o najwyższym stopniu uogólnienia i abstrakcji jak w systemach informowania kierownictwa), projektowanych i konfigurowanych na miarę użytkowników. Według uznanych **autorytetów z dziedziny systemów wspomagania decyzji** przed twórcami i badaczami **w najbliższych latach** pojawią się **następujące wyzwania** [270]:

(1) przetwarzanie informacji niepewnych i niepełnych oraz danych jakościowych na wiedzę która może wspomóc podejmowanie decyzji, (2) opracowanie metodologii efektywniejszego wykorzystania narzędzi i metod inteligentnych systemów przetwarzania wiedzy, (3) dalsze przystosowanie narzędzi programistycznych (szybsze tworzenie przyjaznego interfejsu użytkownika, szybsze uzyskiwanie wsparcia przez decydentów), (4) rozwój własności uczących, rozbudowa modułu wyjaśniającego (transfer metod podejmowania decyzji z bazy metawiedzy do świadomości użytkownika).

3.16 Problematyka projektowania interfejsu człowiek-komputer w systemach wiedzy

3.16.1 Definicja interfejsu człowiek – komputer w SWD

Podstawowym zadaniem interfejsu SWD jest udostępnienie użytkownikowi możliwości, procedur, komend wewnętrznej architektury SWD w celu wspomagania zadań wykonywanych przez niego na różnych etapach podejmowania decyzji. Za **interfejs człowiek-komputer** według Myersa [210] możemy uważać zintegrowane w jednym programie komputerowym **trzy jednolite interfejsy**: **programowy** (obsługujący graficzną, okienkową powłokę systemu, widoczną dla użytkownika, dostarczający elementarnych procedur ekranowych), **aplikacyjny** (obsługujący bezpośredni kontakt z użytkownikiem i elementarne sprzężenia zwrotne) oraz **sterujący** (określający odwzorowanie wyników analiz za pomocą obiektów wizualnych, dynamikę zmian w wyglądzie aplikacji będących skutkiem oddziaływania użytkownika na elementy ekranowe). Nieco inna definicja interfejsu podana przez Verrijin-Stuart [314] dotyczy także systemów wielodostępnych: „interfejs jest formalnym wyrażeniem relacji między systemem a środowiskiem lub między systemem a wielosystemowym środowiskiem czyli zewnętrznym użytkownikiem”.

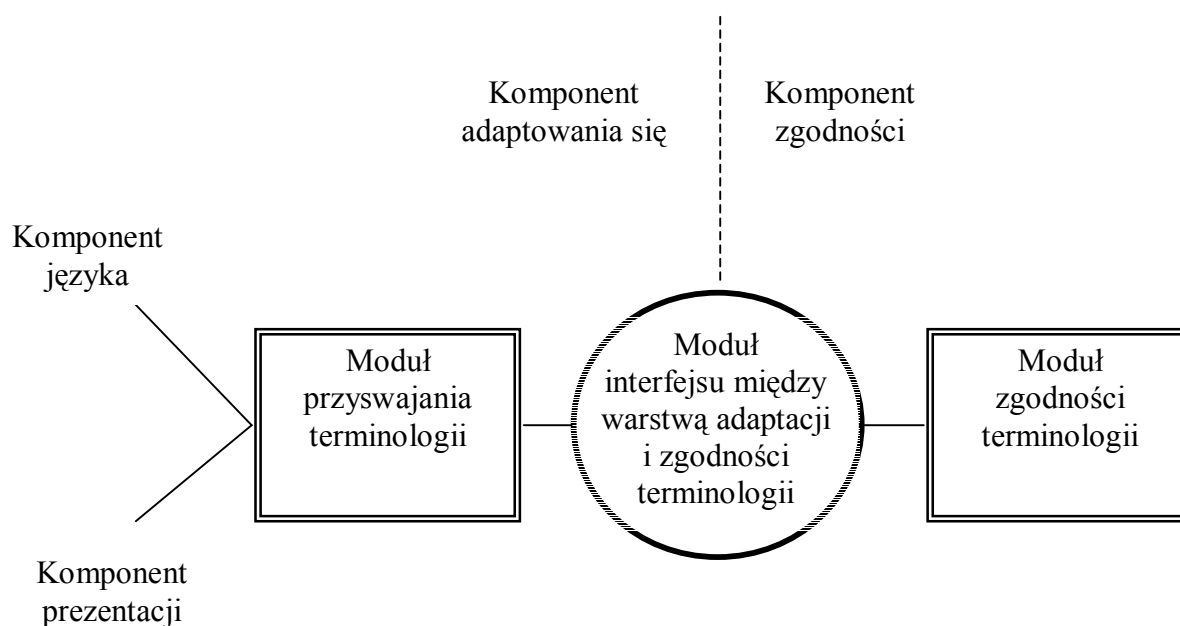
3.16.2 Modele interakcji z systemem przetwarzania wiedzy

Współdziałanie z **systemem przetwarzania wiedzy** (*knowledge system*) za pośrednictwem interfejsu ma charakter intensywny i interaktywny. Wynika on wprost z etapów w ramach których decyzje są podejmowane [19], gdyż zespół analityka i decydenta formułuje zestaw hipotetycznych wariantów decyzyjnych, konfiguruje (tworzy) modele odwzorowujące poszczególne warianty, bada i wartościuje wyniki otrzymane na podstawie wariantów (eksperyment symulacyjny) oraz podejmuje właściwą decyzję (posługując się przyjętym kryterium). Udostępnienie usług wewnętrznej warstwy SWD, tzn. współpracujących ze sobą modułu transformacji opisu problemu w języku komunikacji z interfejsem użytkownika (*problem processing system*) na reprezentację akceptowalną przez moduł przetwarzania wiedzy realizowane jest poprzez dwukierunkowe przesyłanie danych i informacji (komunikowanie się) między użytkownikiem a modułami interfejsu. W komunikacji udział biorą komponenty: interpretujące czynności, tzn. zapytania i polecenia użytkownika (*language systems*) oraz komponent wizualizacji (*presentation system*).

Model systemu przetwarzania wiedzy został przedstawiony w pracy Dos Santosa i Holsapple'a [60]. **Warstwa interfejsu** składa się z modułu interpretacji poleceń użytkownika oraz modułu wizualizacji. Moduł interpretacji poleceń użytkownika implementowany może być w postaci menu, opcji wyboru, języka komend, języka reprezentacji wiedzy, języka naturalnego lub kombinacji

wszystkich powyżej wymienionych. Implementacja możliwa jest dzięki wykorzystaniu tzw. systemu sterowania interfejsem użytkownika (*UIMS*) [143] będącego zestawem (biblioteką) elementarnych narzędzi obsługujących interfejs człowiek-maszyna. Jeśli chodzi o stronę sprzętową, interfejs umożliwia korzystanie z urządzeń zewnętrznych takich jak klawiatura, mysz, ekran dotykowy, mikrofon [258].

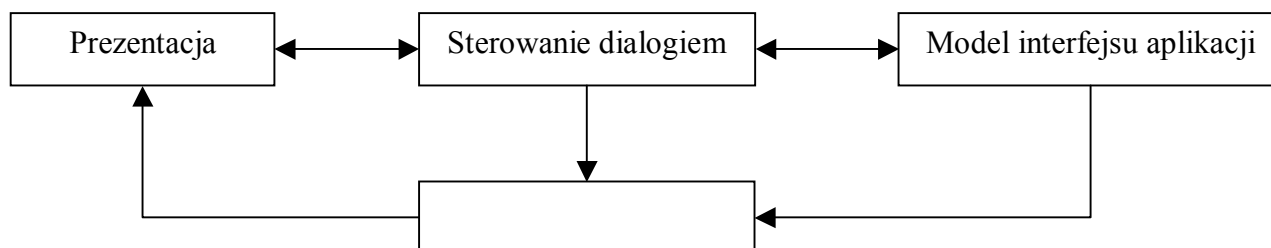
Niektórzy autorzy [258] w przypadku większego stopnia adaptacyjności interfejsu proponują wyróżnić w module interfejsu (dwukierunkowego) dwa wymiary (warstwy): **adaptacyjny** (*adaptive*) i **zgodności** (*consistent*) [Rys. 3-6]. **Warstwa adaptacyjna** (przyswajania) odpowiada bezpośrednio za komunikację z użytkownikiem, w razie potrzeby i w miarę możliwości (graficzny interfejs użytkownika) współpracuje blisko z modulem transformacji opisu problemu (*PPS*). **Warstwa zgodności** odpowiada za odbieranie danych wprowadzonych za pomocą aktualnej konfiguracji zdefiniowanych w warstwie adaptacyjnej (urządzenia zewnętrzne, graficzny interfejs użytkownika). Warstwa ta także obsługuje przygotowanie i udostępnienie (jedynie wstępnie ustrukturalizowanych) danych wyjściowych (materiałów umożliwiających sporządzenie sprawozdań i zestawień finansowych) wysyłanych przez moduł transformacji opisu problemu. Jego adresatem jest użytkownik konfigurujący postać prezentacji (wykresy, animacje, zestawienia liczbowe, analizy i raporty).



Rys. 3-6 Dwuwarstwowy model interfejsu dziedzinowego systemu eksperckiego. Źródło: [258]

Nieco bardziej formalne spojrzenie [249] (sieci przejść – *transition networks* - najstarsze, języki formalne – przydatne przede wszystkim w obsłudze zewnętrznej warstwy języka komend, modelowanie zdarzeniowe i modelowanie za pomocą przedstawiania przykładów (*generation by example*)– powszechne w oprogramowaniu RAD) na strukturę interfejsu użytkownika prezentowane jest w modelu **Seeheim** [101]. W modelu Seeheim [Rys. 3-7] wyróżniono trzy komponenty: **interfejs aplikacji** (komunikacja między interfejsem człowiek-komputer a modułami wewnętrznymi: przetwarzania wiedzy i transformacji opisu wiedzy), **komponent sterowania dialogiem** (definiowanie

struktury komend i hierarchii dialogu) oraz **komponent prezentacji** (czynności wykonywane przez użytkownika powodują generowanie tokenów, czyli sygnałów uruchamiających poszczególne scenariusze ich obsługi).

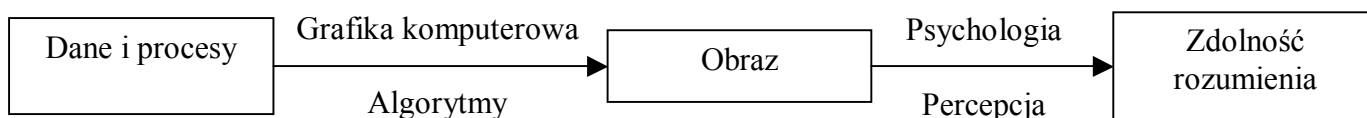


Rys. 3-7 Struktura interfejsu użytkownika wg modelu Seeheim. Źródło: [249]

Rathnam i Mannino [249] wyróżniają dwa wymiary, techniczny i funkcjonalny modułu interfejsu (komponentu języka). **Wymiar techniczny** dotyczy zastosowania języka komend, graficznego zorientowanego problemowo systemu menu i wizualnego modelowania. **Wymiar funkcjonalny** obejmuje reprezentacje (przekodowanie wejścia interfejsu na wewnętrzne obiekty SWD), operacje (uruchamianie procedur realizujących komendy wydawane przez użytkownika), zapytania do baz danych, faktów i reguł oraz obsługę (uruchamianie) komunikacji zwrotnej.

3.16.3 System hybrydowy – wizualizacja komputerowa

Istotą **wizualizacji komputerowej** wg Korzenia [153] udostępnienie procedur umożliwiających tworzenie form graficznych (obrazów) na ekranie monitora. Formy te mogą być dowolnie definiowane przez projektanta systemu i konfigurowalne przez użytkownika. Zainteresowanie wizualizacją [102] wynika z trudności **interpretacji dużej ilości złożonych** (wielowymiarowych) **danych**. **Celem wizualizacji jako technologii** jest transformacja tej złożoności (danych) do postaci umożliwiającej zrozumienie, interpretację i spożytkowanie informacji ukrytych w morzu danych. Jak twierdzi Green [Rys. 3-8], **proces transformacji od abstrakcyjnych danych do rozpoznania** składa się z dwu etapowego odwzorowania: **liczb na obrazy** (artefakty graficzne) wymagającego opracowania procedur-algorytmów z zakresu grafiki komputerowej oraz **obrazów do postaci wewnętrznego zrozumienia** (opisywanej w kategoriach teorii ludzkiej percepcji i psychologii).



Rys. 3-8 Wizualizacja jako efekt dwuetapowej transformacji wartości liczbowej do stanu umożliwiającego rozumienie i wnikliwość. Źródło: [102]

Wizualizacja jest przetworzeniem danych na formę graficzną [265]. Dzięki procesowi wizualizacji dane prezentowane są w taki sposób, aby możliwe stało się **odkrywanie informacji**

(przez decydenta), **podejmowanie** na jej podstawie **decyzji** oraz szybka ich **weryfikacja**. Przedmiotem postrzegania użytkownika (decydenta) są wizualizację wzorców, grup obiektów będących ze sobą w określonej relacji. Zarządzanie procesem wizualizacji odbywa się za pomocą interfejsu użytkownika. Wizualizacja [317] jest jedną z faz analizy danych. Jej niezbędność wynika z **przydatności do zapoznawania się z danymi** i formułowania zgrubnych **ocen jakościowych**. Wizualizacja jest niezastąpiona w **szybkim stawianiu hipotez** (odkrywaniu ukrytych zależności) dotyczących **istniejących trendów**, minimów i maksimów, wzorców typowych zachowań, odchyłeń od prognozowanych trajektorii. Może dotyczyć informacji przekazywanych w czasie rzeczywistym lub poddanych kompresji czasowej np. podczas serii eksperymentów symulacyjnych. Wizualizacja (także dynamiki) przebiegów czasu rzeczywistego szczególne zastosowanie znajduje w **zarządzaniu procesami logistycznymi** w produkcji. Z drugiej strony wizualizacja niezwykle ułatwia zarządzanie i posługiwanie się bardziej statycznymi geograficznymi bazami danych przestrzennych (mapy cyfrowe, sygnalizacja miejsca położenia elementów floty transportowej, śledzenie procesów dystrybucji). Tutaj [304] wyróżnia następujące **cztery kategorie funkcji pakietów wizualizacji**:

1. Zbieranie i wysyłanie danych
2. Prezentacja sytuacji procesowej
3. Wykrywanie i sygnalizacja zdarzeń
4. Rejestrowanie i udostępnianie danych (celem wykorzystania w innych narzędziach analitycznych).

Interpretowane **zbiory danych** reprezentowane są przez **różnorodne elementy graficzne**. Element graficzny w zależności od typu przekazuje dodatkowe informacje za pomocą atrybutów. Do **atrybutów graficznych** zaliczyć można położenie, rozmiary, kształty, przezroczystość i jej stopień a przede wszystkim kolory wypełnień i krawędzi, ich odcienie, nasycenie oraz fakturę. Można wykorzystać także widzialność (pojawianie się i znikanie elementu ekranowego), rotację elementu i zaznaczenie (reagowanie na wybranie urządzeniem wskazującym np. myszą) [304]. Interesujące możliwości otwiera także jak wydaje się przed tworzeniem **obrazów trójwymiarowych** (stereoskopowych). **Wymiar czasowy** z powodzeniem może być przedstawiony poprzez **animację**, ewentualnie **miganie** ze zróżnicowaną częstotliwością i czasem trwania. Wymiar czasowy zwykle dotyczy tego wymiaru **hipersześcianu danych** dla którego chcemy obserwować własności dynamiczne.

Podczas **badania zjawisk gospodarczych**, prowadzonych na podstawie zarówno sprawozdawczości powstającej w przedsiębiorstwie, jak i przeprowadzonych analiz statystycznych (agregacji danych) jak i wyników symulacji z wykorzystaniem modeli ekonometrycznych zachodzi potrzeba **zmierzenia się z wielowymiarowością** (ponad 4) zbiorów danych. Podobne problemy wizualizacji występują także w astrofizyce i badaniami nad DNA [15]. Tylko odpowiednio rozbudowane i skonfigurowane narzędzia wizualizacji pozwalają na **implementację wielu pomysłów** dotyczących **czytelnego przedstawienia wielowymiarowych zbiorów danych**. Jeśli chcemy

wykorzystać istniejące procedury tworzenia wykresów, to można **podzielić całość danych** na kilka **podzbiorów** a następnie przyporządkować je do wybranego wymiaru. Nie można jednak w ten sposób zaprezentować wszystkich wartości danych. Zanika także wtedy możliwość obserwacji wielu innych zależności. Toit i inni [298] przedstawili **metodę rzutowania danych** na wyodrębnione z otoczenia proste lub poprowadzone płaszczyzny. Jej zaletą jest **możliwość prezentacji wartości wszystkich danych** (np. wokół prostej), ale jednocześnie zniekształca ona wzajemne położenie (współrzędne) poszczególnych danych. Ciekawą technikę wizualizacji za pomocą mimiki syntetyzowanej komputerowo twarzy przedstawili Simoff i Sudweeks [274]. Wspomnieć należy także o **technice równoległych osi** przedstawionej przez Inselberta i Dimsdale [127].

4 Systemy hybrydowe

4.1 Hybrydowość

4.1.1 Pojęcie hybrydowego systemu wspomaganie decyzji

Hybrydowy KBS [83] charakteryzuje się możliwością przetwarzania różnych typów wiedzy wyrażonej przez różne techniki reprezentacji wiedzy wykorzystującej różne metody wnioskowania i agregacji czyli łączenia informacji o wielu zróżnicowanych wielkościach i przedstawianie np. za pomocą łącznego wskaźnika. Przesłanką projektowania i implementacji struktur (systemów informatycznych) hybrydowych jest prowadzenie analiz ekonomicznych z wykorzystaniem wzajemnie się uzupełniających, do tej pory oddzielnie rozważanych metod, np. ilościowych i jakościowych, algorytmów genetycznych i wnioskowania, konekjonistycznej i regułowej reprezentacji wiedzy, symulacji i wizualizacji itp. Wydaje się że hybrydowym może być także system wspomaganie decyzji wspierającym przetwarzanie opinii zespołu ekspertów (architektura tablicowa, grupowe podejmowanie decyzji itp.), co powinno prowadzić do agregacji poszczególnych modeli decyzyjnych w celu podjęcia bardziej optymalnej (lub umotywowanej) decyzji – np. poprawnej diagnozy o wiarygodności kredytowej przedsiębiorstwa. Ze względu na złożoność strukturalną, zadaniową i metodologiczną hybryd systemy wspomaganie decyzji, które stanowią dla nich platformę informatyczną także cechują się znacznym stopniem złożoności komunikacyjnej, funkcjonalnej i strukturalnej.

4.1.2 Pojęcie modułowości

Modułowość i **abstrakcja** polegające na konceptualizowaniu typów danych, procesów obliczeniowych i komunikacji w formie procedur są odpowiedzią projektantów systemów na złożoność, wielkość i cechę nieprzewidywalności dotyczących dziedziny problemowej wymagającej zastosowania systemów komputerowych. Często jedynym sposobem jest opracowanie **specjalizowanych modułów** (posiadających własną reprezentację wiedzy i metod rozwiązywania problemu) dla każdego aspektu problemowego. **Modułowość** polega na **delegowaniu poszczególnym modułom** (względnie inteligentnym agentom) pewnych wycinków, etapów, czy wyodrębnionych procesów związanych z rozwiązywaniem problemów. W takiej sytuacji można zaprojektować odpowiednio **proste** ale **zróżnicowane** i **specjalizowane moduły** których utrzymanie jest prostsze i tańsze. Zalety modularności polegają na wykorzystaniu upraszczania [26] i rozproszonego przetwarzania informacji. Wynika stąd **potrzeba komunikowania się** między poszczególnymi modułami, aby zapewnić koordynowanie (moduł czy agent funkcjonuje na podstawie znanej mu lokalnej perspektywy) działań poszczególnych modułów w celu znalezienia kompromisowych rozwiązań (uniknąć optymalizacji lokalnych zadań, bez uwzględniania głównego zadania).

4.1.3 Przesłanki hybrydowości

Ze względu na **złożoność przedmiotu badań** (wiele komponentów, procesów, danych), **różnorodność stosowanych metod analizy**, systemów inteligentnych (w tym tradycyjnych: badań operacyjnych, systemów eksperckich, analizy systemowej, symulacji itd. oraz *soft computing*: logiki rozmytej, sieci neuronowych, algorytmów ewolucyjnych, zbiorów przybliżonych, *case-based reasoning*, inteligentnych agentów, wizualizacji danych itd.) i prowadzenie jej w **dynamicznych warunkach**, poszukuje się **nowych sposobów** prowadzenia badań będących **syntezą** już dostępnych **metod i technik**. Oczekuje się, że dzięki **syntezie metod** ich zalety będą się potęgowały, wady zanikały a oprócz tego osiągnięty zostanie **efekt synergiczny**. Wydaje się, że w tym celu należy zastosować nowoczesne techniki informatyczne a przede wszystkim opracować nowe metody odwzorowania przedsiębiorstwa lub przynajmniej wykorzystać dostępny model (symulacyjny) badanego systemu – przedsiębiorstwa. Aby synteza i jej pozytywny efekt czyli synergia przybrały postać konkretną, założyć należy **zintegrowanie** (za pomocą technik komputerowych) dotychczas oddzielnie wykorzystywanych **technik analitycznych**. Taką integrację różnorodnych technik można porównać z **krzyżowaniem różnorodnych osobników** w celu uzyskania potomstwa wykazującego zwiększoną jakość działania. Według Kopalińskiego [152] przejawem synergii takiej struktury jest **heterozja** czyli „wigor mieszańców, zwiększona żywotność i wybujałość zwierząt i roślin pochodzących z krzyżówek form genetycznie odmiennych”. W przyrodzie heterozja dotyczy **najczęściej tylko pierwszego pokolenia** a jej przyczyną jest „wysoki stopień heterozygotyczności tego pokolenia, z czym wiąże się niska frekwencja genów kodujących cechy niekorzystne dla rośliny (naddominacja alleli jednego genu, lub współdziałania dominujących alleli różnych genów)” [226]. W przyrodzie obserwuje się zanikanie tego zjawiska w dalszych pokoleniach – stąd wniosek, że podtrzymać można je przede wszystkim w sposób sztuczny, przemysłowy. **Efektom heterozji** (nazywanej w przypadku przemysłowych zabiegów hybrydyzacją) jest **hybryd** (hybryda) czyli mieszaniec **wykazujący zwiększoną żywotność i płodność**. W przypadku systemów nieorganicznych pozytywnie oceniany będzie hybryd (hybryda) – artefakt (program komputerowy, metoda) wykazujący zwiększoną użyteczność, jakość rozwiązań itp. Ultsch [309] używa określenia hybryda w kontekście **form hermafrodytowych** tworzonych poprzez **fuzję lub krzyżowanie**. Uwaga: **dodawanie funkcji do systemu** (modelu, organizmu, organizacji) **nie wystarczy** aby system wynikowy nazwać **hybrydą**: dołączenie do komputera urządzenia peryferyjnego nie czyni go automatycznie hybrydą, gdyż do zaistnienia zjawiska **heterozji** niezbędne jest „wymieszanie”, **wzajemne oddziaływanie cech** a nie ich dodawanie. Poszczególne techniki w systemach hybrydowych przeważnie uzupełniają się, ale mogą także ze sobą konkurować.

Hybrydowość nie jest uniwersalnym lekarstwem na wszystkie problemy i o jej zastosowaniu należy decydować wtedy gdy istnieje **uzasadnione przekonanie**, że dzięki niej **dowiemy się czegoś nowego** (aspekt jakościowy) lub **rozwiązanie osiągnięte** zostanie **w krótszym czasie, mniejszym kosztem** lub charakteryzować będzie się **większą dokładnością** lub **pewnością**. Za hybrydowością przemawia jednak sformułowana przez Wolperta i Macready'ego [326] hipoteza *No Free Lunch Theorem* (NFL). Na jej podstawie dowodzi się, że żaden z algorytmów (bez dodatkowej wiedzy dziedzinowej) nie może być „najlepszy” we wszystkich problemach. Jeśli dany algorytm w **pewnej grupie problemów** uzyskuje przewagę (tzn. jest lepiej do nich dobrany) to jest taka grupa innych problemów w której przewagę wykaże inny algorytm. W kontekście algorytmów przeszukiwania przestrzeni rozwiązań, w tym algorytmów ewolucyjnych i w szczególności genetycznych, Dembski [56] twierdzi że „ewolucja niczego nie poszukuje a darwinizm nie jest algorytmem poszukiwania rozwiązań”. Ho i Pepyne [117] formułują NFL następująco: „bez obecności szczególnych założeń strukturalnych, żaden algorytm optymalizacji nie może średnio działać lepiej niż ślepe wyszukiwanie”.

Systemem hybrydowym wg G. i R. Kilgourów z University of Otago [146] można nazwać system (komputerowy) który **korzysta z więcej niż jednej techniki rozwiązania problemu**. Poszczególne **techniki rozwiązywania problemów** (nie tylko z zakresu sztucznej inteligencji) to: sieci neuronowe (z uczeniem nadzorowanym i uczeniem bez nadzoru), regresja, agregacja danych, logika rozmyta, algorytmy genetyczne, systemy eksperckie (z reprezentacją wiedzy opartą na regułach), drzewa decyzyjne, algorytmy grupowania (najbliższego sąsiada, k-najbliższych sąsiadów, średniej minimalnej odległości - MMD, drzewa minimalnego, algorytmy iteracyjne korzystające z pojęć zbiorów przybliżonych i rozmytych), sztuczne życie, symulacja.

4.2 Cele opracowania hybrydowych systemów wspomaganie decyzji

Tworzenie hybryd polega na połączeniu dwu lub większej liczby różnych podejść (metodologii) w celu utworzenia nowego pojedynczego podejścia (metodologii).

Jak piszą Leigh i inni [168] wyróżnić można **dwa podejścia** dotyczące **tworzenia i zastosowania narzędzi** jakimi są **systemy wspomaganie decyzji**: **klasyczne** (oparte głównie na teoriach) i **romantyczne** (oparte raczej na danych). **Podejście romantyczne** akcentuje zatem obecność ciągłych i szybkich zmian środowiska, znacznego stopnia niepewności (co więcej – o zmiennych charakterystykach). **Brak stabilności** (zakładanej w podejściu klasycznym) obserwowany w rzeczywistych problemach, jest **przyczyną zawodności modeli ekonomicznych** (nawet jeśli są zdolne do adaptacji do zmian) wykorzystywanych w celu prognozowania, podejmowania decyzji i oceny stanu (diagnozy i klasyfikacji do określonych grup ryzyka) przedsiębiorstwa. Dlatego też czynione są wysiłki **wykorzystania wielu technik, teorii, technologii** w celu **poprawy wyników procesu wspomaganie decyzji** (zwiększenia dokładności i obiektywności, przyspieszenia przetwarzania informacji i wiedzy, zdolności adaptowania się modeli i wykorzystania dostępnych danych). Przesłanką takiego postępowania jest postulat tworzenia systemów inteligentnych, czyli **zdolnych do dostosowania się do warunków** w których będą wykorzystywane [130], jeśli nie samodzielnie to co najmniej **bezpośrednio przez użytkownika**. Warto także przypomnieć, że decydenci stają się hybrydowi (części ludzka i komputerowa mieszają się ze sobą), jednak człowiek i maszyna korzystają z różnych informacji [18].

Hybrydowe narzędzia wspomaganie podejmowania decyzji konstruowane są za pomocą połączenia i różnego stopnia integracji różnorodnych technik, teorii, technologii, modułów informatycznych i sprzętu komputerowego. Jak piszą Matsatsinis i Samaras [189] **skupienie uwagi badawczej na integracji SWD z narzędziami sztucznej inteligencji** (systemy eksperckie, *KBDSS* i innymi) jest wynikiem „dwudziestu lat szybkiego rozwoju technologii informacyjnej i informatyki”. Jednym z efektem tego rozwoju możliwość integracji **modeli** i metod (tu skorzystać można z doświadczeń twórców SWD, znajdujących się od dawna „bliżej” problemów decyzyjnych) oraz **technik inteligentnych** (w tym systemów eksperckich – tu skorzystać należy z doświadczeń z zakresu przetwarzania wiedzy oraz technologii informatycznych). Celem budowy systemów hybrydowych jest bowiem [189] „integracja SE oraz metod wielokryterialnych, statystycznych badań wielu zmiennych czy algorytmów znanych z programowania matematycznego, które dostarczają decydującym wyższego poziomu wsparcia w trakcie procesu podejmowania decyzji”.

Już w 1988 roku O’Keefe [221] twierdził, że w **centralnym trendzie zastosowań SWD** widoczne jest zainteresowanie **hybrydowością** mającą na celu wykorzystanie **mocnych stron składowych** technik i połączenie odpowiednich elementów z poszczególnych technologii (analizy decyzyjnej i wiedzy zakodowanej w postaci reguł wnioskowania). Natomiast w roku 1995 Liebowitz

[172] pisał, że **funkcjonujący samodzielnie system ekspercki jest rzadkością**, natomiast powszechne stało się integrowanie SE z sieciami neuronowymi, analitycznymi systemami wspomaganiami decyzji, narzędziami *CASE* i multimediami.

Jak wskazują Matsatsinis i Samaras [189] w hybrydach SE i SWD moduł SE służy: (1) „wspieraniu interpretacji rezultatów pochodzących z modeli matematycznych”, (2) „pomocy przy formułowaniu podstawowych założeń leżących u podstaw zastosowania tych modeli”, (3) „rekomendowaniu określonych wariantów decyzyjnych aby lepiej zrozumieć alternatywne rozwiązania, z którymi spotyka się decydent”. Szczególnie wiele **efektów synergicznych** oczekiwano od hybrydowego **połączenia systemu eksperckiego i modeli symulacyjnych**. Jednocześnie **widziano potrzebę badań** nad pozyskiwaniem i reprezentacją wiedzy, transferem technologii, weryfikacją i walidacją modeli w kontekście zintegrowanych, hybrydowych systemów inteligentnych.

Hybrydowość SWD polega na **pragmatycznej i skoordynowanej** (równoległej, szeregowej, hierarchicznej czy prowadzonej poprzez komunikację międzymodułową) **współpracy**, współzawodnictwa w wykonywaniu poszczególnych zadań lub nadzorowania pracy modułów implementujących techniki sztucznej inteligencji, prognozowania, uczenia maszynowego, sieci neuronowych, algorytmów genetycznych, symulacji, regułowych systemów eksperckich. Dutta [67] argumentuje, że **przesłanką integracji** zawartych w SWD modeli matematycznych (w tym optymalizacyjnych) oraz technik sztucznej inteligencji jest **dostrzeżenie słabości metod funkcjonujących osobno** i potencjału wzmocnienia „siły” każdej z nich. Warto przypomnieć w tym miejscu, że **główne zalety technik optymalizacyjnych** to możliwość dobrego udokumentowania i powszechnego zrozumienia sposobu rozwiązywania problemu i dokonywania oceny rozwiązań w kategoriach ilościowych. Za **główną słabość technik optymalizacyjnych** możemy uważać **problemy z analizą wykonalności znalezionych rozwiązań**. Wynikają one z następujących czynników [67]: niedostatecznego stopnia realności modeli oraz przeznaczenia do rozwiązywania problemów głównie ilościowych i dobrze ustrukturalizowanych, tendencji do utknięcia w lokalnym minimum, trudności w aktualizacji modeli, potrzebie posiadania specjalistycznej wiedzy i doświadczenia przy formułowaniu modelu i interpretacji rozwiązań.

Niektóre z **niedomagań** mogą być **przewyciężone** poprzez techniki poszukiwania rozwiązań optymalnych wywodzących się z **obserwacji procesów przyrodniczych** (są one niekiedy zaliczane do technik sztucznej inteligencji) takie jak algorytmy genetyczne, symulacyjne wyżarzanie, *tabu search*. Przeważnie jednak **nie wykorzystują one wiedzy heurystycznej** (oprócz *tabu search*) i rozpoczynają swoje działanie praktycznie od zera, **pomijając** dostępne **doświadczenia i wiedzę**. Można się zatem spodziewać, że **techniki sztucznej inteligencji** (a głównie te korzystające z wiedzy eksperckiej, naśladujące sposób rozumowania człowieka oraz wspierające analizę wykonalności decyzji) **pomogą zmniejszyć trudność problemów** (ograniczyć wielkość zbioru) dotyczącą rozwiązań wynikowych (wariantów decyzyjnych).

4.3 Przykłady systemów wspomagania decyzji rozszerzonych rozbudowanych o hybrydowość

Przykładem integracji standardowej dla SWD techniki optymalizacyjnej oraz systemu eksperckiego na polu zarządzania inwestycjami jest K-Folio [163]. **Optymalizacja dotyczy tu zależności zysk-ryzyko** ujętej w postaci problemu programowania kwadratowego Markowitza [186] (w którym funkcją celu jest tzw. wariancja portfela czyli suma wariancji i kowariancji poszczególnych walorów a liniowe ograniczenia określają dolną granicę zwrotu z portfela) natomiast **za pomocą systemu eksperckiego** (z reprezentacją wiedzy w postaci ram) definiowane są **zalecenia** (preferencje) **eksperta dziedzinowego** (specjalisty z zakresu inwestycji kapitałowych). Dzięki połączeniu obu technik użytkownik uzyskuje **bardziej wiarygodne wsparcie** w podejmowaniu decyzji inwestycyjnych.

Dutta [67] dowodzi, że **głównym problemem podczas przedsięwzięć integrowania platformy SWD** i jej metod (programowania, modeli statystycznych, symulacji) z **technikami sztucznej inteligencji** jest konieczność **dostosowania metod** tej integracji do **dziedziny zastosowań**. Dlatego też niezbędnym jest opracowanie narzędzi umożliwiających integrację procedur, struktur, danych. Metoda **zintegrowanego modelowania strukturalnego** została opisana przez Geoffriona [92]. **Strukturalne modelowanie** opiera się na tworzeniu **diagramów struktury danych** (*Entity Relationship Diagrams*) oraz diagramów przepływu danych (*DataFlow Diagrams*). Ważną zaletą metody SM jest możliwość **wydobywania modeli z istniejących aplikacji**, co pozwala na przystosowanie kodu aplikacji do nowych środowisk programowych. Wśród pozostałych zalet modelowania strukturalnego wymienić można **bogate możliwości języka modelowania** pod względem manipulacji semantycznych i jego obiektowość (reprezentowanie hierarchii obiektów), niezależność modelu i danych od środowiska programistycznego, wsparcie dla reprezentacji różnorodnych, współzawodniczących modeli, zarządzanie (składowanie, wydobywanie, modyfikacje) bazą modeli analogiczne do zarządzania bazą danych.

Jo i Han [134] przedstawili badania nad wynikami zastosowania rozmaitych konfiguracji **hybrydy** sieci neuronowej, wnioskowania na podstawie przypadków (*case-based*) oraz analizy dyskryminacyjnej jako metody statystycznej. Hybryda miała na celu **klasyfikację firmy** do poszczególnych **klas zagrożenia bankructwem** (jako przedmiot walidacji). Klasyfikację przeprowadzano na podstawie wartości **funkcji dyskryminacji**. Model **zintegrowanego** (hybrydowego) **klasyfikatora** zawierał **cztery fazy przetwarzania danych**: trenowanie (uczenie), testowanie wstępne, ustalanie wartości wag (funkcji dyskryminacyjnej) z wykorzystaniem minimalizacji błędu za pomocą programowania kwadratowego z ograniczeniami, testowanie końcowe oraz **pięć typów danych**: dane uczące (o charakterze zewnętrznym), dane testowe (dla sieci neuronowej, zewnętrzne), wartości prognozowane za pomocą poszczególnych metod (wewnętrzne),

dane uogólnione (służące do oceny dokładności i efektywności klasyfikacji poszczególnych metod, zewnętrzne) oraz wyniki klasyfikacji (dane wewnętrzne). Ze względu na **niekompatybilność wartości wyjściowych** dla poszczególnych metod zastosowano ich **standaryzację**. Po przeprowadzaniu szeregu eksperymentów z bazą danych o firmach, które uległy bankructwu autorzy doszli do wniosku, że **integracja metod może dawać lepsze wyniki** (jeśli chodzi o błąd klasyfikacji) niż każda z metod zastosowania oddzielnie (wiele zależy także od kombinacji zestawów uczących i testowych).

Ahn i inni [4] dowodzą, że hybrydę sieci neuronowych i aplikacji zbiorów przybliżonych można zastosować nie tylko do badania zagrożenia bankructwem, ale także znacznie **szerszej problematyki przewidywania niepowodzeń biznesowych**. Niepowodzenia biznesowe skutkują **oprócz** najbardziej dotkliwego **ogłoszenia bankructwa**, także niespłaceniem kredytów, opóźnieniami w wypłatach dywidendy, zaleganiem z płatnościami wobec dłużników, brakiem środków na rachunku bieżącym. Organizację wykrywania tych wszystkich zagrożeń określa się mianem **systemu wczesnego ostrzegania**. Oprócz osobiście zainteresowanych podmiotów, także **kredytodawcy** bacznie obserwują **symptomy niepowodzeń** biznesowych. Hashemi i inni [110] oraz Jelonek i inni [132] donoszą o zastosowaniu **hybrydy zbiorów przybliżonych** (operujących na wartościach dyskretnych) i **sieci neuronowych** w bankowości. Jej zasada działania polega na wstępnym przetwarzaniu danych przez aplikację zbiorów przybliżonych (odkrywanie ukrytych zależności między atrybutami obiektów oraz ograniczenie liczby tych atrybutów bez wpływu na informacje wymagane do podjęcia decyzji) zanim przekazane zostaną do sieci neuronowej celem dokonania klasyfikacji. Według Ahna i innych [4] **za pomocą aplikacji zbiorów przybliżonych** można pozyskać wiedzę w postaci dwu rodzajów **reguł: deterministycznych i niedeterministycznych**. Reguł niedeterministycznych nie da się sensownie wykorzystać, gdyż za ich pomocą nie da się przeprowadzić jednoznacznej klasyfikacji nowo badanego przypadku (następuje konflikt). Dzięki zastosowaniu aplikacji zbiorów przybliżonych **unika się skonfliktowania wyjść i wejść** sieci neuronowych (konieczności podjęcia decyzji w sytuacji różnych wartości na wyjściu przy tych samych wartościach wejściowych).

Ahn i inni [4] przedstawili zastosowanie **dwu odmian przetwarzania danych** za pomocą **zbiorów przybliżonych: horyzontalnego** (redukcji atrybutów nadmiarowych) oraz **dwuwymiarowego** (wykonującego także usuwanie skonfliktowanych obiektów; może powstać w ten sposób nowa reguła decyzyjna). Hybryda działa na zasadzie **filtrowania badanych przypadków**: gdy badany obiekt (przypadek) może być obsłużony przez regułę to klasyfikacja przebiega zgodnie z tą regułą, natomiast gdy na podstawie dostępnych reguł nie można wyznaczyć wyniku klasyfikacji, to uruchamiane jest przetwarzanie danych przez aplikację zbiorów przybliżonych i zmodyfikowany (podlegający odpowiedniej redukcji) zestaw reguł wykorzystywany jest w trenowaniu (uczeniu) sieci neuronowej. Każdy przypadek (obiekt) podlegający klasyfikacji w zadaniu oceny zagrożenia bankructwem charakteryzowany jest przez **zestaw wartości wskaźników finansowych** (zebranych przed klasyfikacją) takich jak: zysk netto / aktywa ogółem; zysk netto / sprzedaż, kapitał akcyjny

uprzywilejowany / aktywa razem, zobowiązania ogółem + zobowiązania z tytułu emisji dłużnych pap. wartościowych / aktywa razem, kapitał pracujący / aktywa razem, przepływy pieniężne netto / zobowiązania ogółem, obrót zapasami, aktywa bieżące / zobowiązanie krótkoterminowe. Z badań przeprowadzonych z wykorzystaniem danych o firmach spoza sektora finansowego wynika **przewaga efektywności modeli hybrydowych** oraz **dwuwymiarowego wstępnego przetwarzania** z wykorzystaniem aplikacji zbiorów przybliżonych.

Jak wykazali Lee i inni [166] stopę błędnych klasyfikacji w metodzie punktowej (*credit scoring*) można **zmniejszyć** wykorzystując **strukturę hybrydową**. W omawianym rozwiązaniu badano **prawidłowość klasyfikacji** (dobry czy zły klient) wniosków o kartę kredytową scharakteryzowanych za pomocą zestawu dziewięciu cech (wiek, płeć, stan cywilny, dochody, wykształcenie itd.). Zastosowano **dwuetapową klasyfikację** z wykorzystaniem liniowej analizy dyskryminacyjnej (*LDA*) oraz wariantu z logistyczną regresją (*LR*) i sieci neuronowej (*backpropagation*, jedna warstwa ukryta). Na wejściu **struktury klasyfikacyjnej** ulokowano procedurę analiz statystycznych (*LDA* lub *LR*) której **wyniki** traktowano jako **wartości wejściowe** dla procedury **sieci neuronowej**. Sieć neuronowa posiadała zatem **wielowymiarowe wejście** (zmienne wyjaśniające wpływające na wiarygodność wnioskującego o kredyt i wynik metody dyskryminacyjnej) oraz **jednowymiarowe wyjście** (wynik klasyfikacji: przyznać czy nie przyznać kredytu). Porównania efektywności liniowej analizy dyskryminacyjnej i sieci neuronowej w problemie *credit scoring* dokonali Coats i Fant [43]. **Celem klasyfikatorów** było oznaczenie firmy jako przejawiającej "wigor" oraz „osłabienie”. Badania ich wykazały większą przydatność sieci neuronowych jeśli chodzi o **przewidywanie osłabienia** kondycji firmy. Wydaje się że przyczyną powodzenia dwuetapowej klasyfikacji może być wsparcie algorytmu sieci neuronowej w początkowym okresie obliczeń i dostarczenie dodatkowej, przydatnej informacji o badanych przypadkach (używanych na etapie uczenia sieci a następnie testowania). Chen i Huang [41] przeprowadzili eksperymenty w zakresie *credit scoring* z **hybrydą sieci neuronowych z algorytmem genetycznym**. Zadaniem sieci neuronowej był **wstępny podział wniosków kredytowych** na dwie grupy (przyjętych i odrzuconych). Następnie dokonano przetworzenia wniosków odrzuconych zgodnie z zasadami tzw. *inverse classification* opisaną przez Mannino i Koushika [184]. Założeniem tej techniki jest wykorzystywanie w procesie uczenia sieci neuronowej zarówno **przyjętych** jak i **odrzuconych** wniosków oraz **wzbogacenie sieci neuronowych o możliwości objaśniające** (nawet niewielkie). Chodzi tu o określenie jaki dystans dzieli obiekt (wniosek) o określonych atrybutach od przypisania do przeciwnej klasy. Zadaniem **algorytmu genetycznego** (drugi etap analizy) jest w tym układzie **optymalizacja**. Chodzi o znalezienie takiej wartości zmian (w postaci zwiększenia lub zmniejszenia wartości) parametrów atrybutów, na których opiera się klasyfikacja wykonywana przez sieć neuronową, aby wnioski wstępnie odrzucone mogły być rozpatrzone pozytywnie. Dzięki takiemu

rozwiązaniu możliwa stała się **analiza wrażliwości** oceny wniosków kredytowych na **zmiany wartości atrybutów**.

Konwencjonalny **rachunek kosztów działań** (ABC) ma pewne przewagi nad tradycyjnymi rachunkami kosztów (posługujących się systemem kluczy rozdzielających koszty). Polegają one na mniej arbitralnym przypisywaniu kosztów pośrednich (w tradycyjnym rachunku kosztów wykorzystuje się zwykle pojedyncze kryterium takie jak np. czas pracy). Metoda rachunku kosztów (posługująca się pojęciami kosztów działań, nośników kosztów, przypisująca koszty działań poszczególnym produktom) opisana przez Coopera [47] posiada **dwie słabe strony**: brak uznanych kryteriów reguł wyboru nośników kosztów oraz założenie o liniowym charakterze funkcji kosztów (koszty całkowite mają być proporcjonalne do pojedynczego nośnika kosztów). Założenie to zostało zakwestionowane przez Horngrena i innych [120]. Kim i Han [148] wykazali, że **łącznie w sposób szeregowy** procedurę **algorytmu genetycznego** oraz **sztucznej sieci neuronowej** można pokonać te słabe strony. W pierwszym etapie obliczeń zastosowano **procedurę algorytmu genetycznego**, natomiast w drugim etapie **sztuczną sieć neuronową** typu *feed-forward*, z **wsteczną propagacją błędów** oraz **funkcją aktywacji** typu sigma. Zadaniem **algorytmu genetycznego** jest **zidentyfikowanie** suboptymalnych nośników kosztów i **wyznaczenie** liczby ukrytych neuronów sieci neuronowej. Optymalizacja za pomocą algorytmu genetycznego odbywa się w obrębie populacji chromosomów reprezentujących parametry modelu natomiast jako funkcję przystosowania przyjęto **średnią różnicę** (odchylenie) między wartościami oczekiwanych (znanych) i przewidywanych kosztów rozważanego produktu. Zadaniem **sztucznej sieci neuronowej** jest estymacja kosztów za pomocą liniowej i nieliniowej regresji. W drugim etapie zastosowano liniową funkcję wartości wyznaczonych nośników kosztów. W trzecim etapie ma miejsce uczenie sieci neuronowej na podstawie dostępnych danych doświadczalnych. Jak wykazały badania, w rozwiązaniu wykorzystującym architekturę hybrydową, uzyskano **dokładniejsze oszacowanie kosztów** przy jednoczesnym **uproszczeniu struktury** sieci neuronowej (zmniejszenie liczby neuronów w warstwie ukrytej).

4.4 Zalety i wady poszczególnych technik

Systemy wspomaganie decyzji stosowane w decyzyjnej analizie ekonomicznej wykorzystują szereg metod i technik analitycznych (sztuczną inteligencję, symulację, optymalizację, wizualizację, arkusze kalkulacyjne, raportowanie). Każda z nich posiada zarówno **zalety** jak i **wady** (manifestujące się nawet całkowitą bezradnością). Przejawiają się one **w zależności od złożoności i charakteru problemu decyzyjnego, dostępnych danych i metod rozwiązań**, wymagań dotyczących **formy rozwiązań**. Z jednej strony znana jest **większa przydatność systemów eksperckich** niż sieci neuronowych w przygotowaniu słownych analiz, raportów i zaleceń (ogólnie: uzasadnień podejmowanych decyzji) a z drugiej strony, w przypadku szybko zmieniającego się środowiska i warunków gospodarowania wymagających dostosowania bazy wiedzy, inżynier wiedzy **będzie preferował sieć neuronową** (posiadającą zdolność uczenia się i dostosowania do nowych warunków) względem systemu eksperckiego (choć efektywniej przetwarza on wiedzę o reprezentacji regułowej, ale o charakterze statycznym). Różnice między poszczególnymi technikami reprezentacji i przetwarzania wiedzy Watkins ujął w następujący sposób [318]: „można narysować wykres posiadający dwie osie: na jednej wymiar wiedzy ustrukturalizowanej i nieustrukturalizowanej, na drugiej przetwarzanie numeryczne i nienumeryczne”. **Twarde, kontekstowe** techniki sztucznej inteligencji takie jak systemy eksperckie ułożyć można w ćwiartce „wiedza ustrukturalizowana, przetwarzanie nienumeryczne (symboliczne)”. **Miękkie, bezkontekstowe** techniki sztucznej inteligencji takie jak sieci neuronowe i algorytmy genetyczne przystosowane są do przetwarzania numerycznej wiedzy nieustrukturalizowanej. **Logikę rozmytą**, w pewnym sensie **łączącą cechy** „twardych” i „miękkich” technik inteligentnych umieścić można w ćwiartce „wiedza ustrukturalizowana, przetwarzanie numeryczne”. Podobną taksonomię [306] przedstawili Tzafestas i Stamou.

Warto zauważyć, że jak dotychczas **nie opracowano uznanych metod i narzędzi** symbolicznego przetwarzania wiedzy nieustrukturalizowanej. Być może **narzędzia hybrydowe** poradzą sobie z powyższym problemem poprzez (1) **strukturalizację wiedzy** (*data mining, knowledge discovery in data bases*, hermeneutykę) i w następnym etapie **przetwarzanie symboliczne** (systemy eksperckie) lub (2) **zmianę reprezentacji wiedzy** z symbolicznej na numeryczną (systemy eksperckie w architekturze neuronalnej) a następnie z jej **strukturalizację** (systemy rozmyte). Jak piszą Zhang i Zhang [338] planowanie finansowe (którego elementem jest analiza finansowa) wymaga zastosowania wielu komponentów, „współdziałających na złożone i zmieniające się sposoby”. Ze względu na tę **złożoność i dynamikę analizowanego podmiotu** wyniki analiz mogą być trudno zrozumiałe, trudno przewidzieć także następstwa decyzji podjętych na podstawie analizy. Opierając się na wykazie zaproponowanym przez Zhanga i Zhanga [338] wyszczególnić można następujące **uwarunkowania procesu analizy i planowania finansowego przedsiębiorstwa**:

1. Konieczność zarządzania wielką liczbą danych, dostarczanych często w **formie nieorganizowanej, niezgodnej z ustalonymi formatami** lub nawet niedostępnej w postaci elektronicznej
2. Konieczność uwzględnienia **różnorodnych informacji** pochodzących z **różnych źródeł**, znacząco wpływających na podejmowane decyzje: charakterystyka i informacje pochodzące z segmentu rynku, ważne wiadomości podawane przez media, sprawozdania i raporty finansowe, modele zgodnie z którymi analizuje się raporty, model przedsiębiorstwa, procesów rynkowych, zasady rachunkowości, analizy dostarczane przez analityków
3. W wielu źródłach informacji obserwuje się **czynnik niepewności, dynamiczne zmiany środowiska, opóźnienie analizowanej informacji** względem **rzeczywistości** itp.

4.5 Konekcjonistyczna a symboliczna reprezentacja wiedzy

Bigdoli [20] twierdzi: „przed zastosowaniem hybryd, system ekspercki działa poprawnie, ale tylko do błędu powodującego katastrofę. Dzięki rozwiązaniu hybrydowemu system ekspercki może uczyć się na podstawie przeszłych doświadczeń a nawet **radzić sobie z nienotowanymi dotychczas sytuacjami wyjątkowymi**, prowadzącymi dotychczas do **katastrofy**.”

Motywacją tworzenia **systemów hybrydowych** jest **przekonanie o dodatnim synergicznym efekcie** zastosowania oprogramowania korzystającego z dwóch lub większej liczby technik, reprezentacji wiedzy i sposobów jej przetwarzania. Spodziewamy się że hybryda wykazywać będzie co najmniej połączone zalety technik składowych oraz że zlikwidowaniu lub co najmniej zmniejszeniu ulegną niedomagania (wady) technik składowych, tzn. że hybryda będzie wynikiem zmieszania wzajemnie dopełniających się cech pozytywnych. Przykładem jest rozumowanie Wermtera i Suna [322], którzy przystępując do projektowania hybrydy sieci neuronowej (reprezentacji konekcjonistycznej) i regułowego systemu eksperckiego (reprezentacji symbolicznej) zestawili zalety poszczególnych technik w następujący sposób:

Do zalet **reprezentacji konekcjonistycznej** zaliczyć można [155] **zdolność uczenia się** i douczania (ze względu na iteracyjny charakter algorytmów uczenia), **tolerancję występowania błędów** w dostarczanej wiedzy (zdolność do filtrowania zakłóceń) i **zdolność uogólniania** oraz w większości zastosowań praktycznych - wysoką trafność klasyfikowania. Do zalet **reprezentacji symbolicznej** zaliczyć można łatwą interpretację, jawny charakter sterowania, stosunkowo szybkie kodowanie (programowanie) prototypów, łatwe dołączanie nowych pojęć i zmiennych, zdolność do abstrakcji wiedzy. **Porównując ze sobą obie reprezentacje wiedzy** i wykorzystujące je techniki można dojść do następujących wniosków: lepsza **aprosymacja** w przypadku sieci neuronowych, **czytelność** pozyskanej wiedzy lepsza w technikach symbolicznych, większa **odporność na zakłócone dane** i wzorce w przypadku sieci neuronowych, oraz (względne) **wady** sieci neuronowych które, jak się oczekuje, zostaną zneutralizowane w hybrydzie: **trudności z interpretacją** i objaśnianiem wiedzy nabytej przez sieć, **rozproszenie reprezentacji wiedzy** (brak odpowiednika bazy wiedzy z systemów regułowych), trudności z **wyjaśnieniem** sposobu dojścia (i zdolności uogólniania) do rozwiązania (sieć neuronowa jest czarną skrzynką), **niezadowalająca szybkość** większości algorytmów uczących, trudności z zabezpieczeniem przed **utknięciem w ekstremum lokalnym**.

Jak piszą Zhang i Zhang [338] “projektowanie i rozwijanie inteligentnych systemów hybrydowych jest **zadaniem trudnym**, ponieważ składają się z **dużej liczby części** lub **komponentów** między którymi zachodzą **liczne interakcje**”. Co więcej, nawet dostępne **współcześnie narzędzia** software’owe **nie wspierają procesu budowy systemów hybrydowych** właśnie ze względu na te złożone i liczne interakcje, które „mogą się zdarzyć w **nieprzewidywalnym momencie**, ze **niezidentyfikowanych przyczyn** i nie będzie można przewidzieć **między którymi technikami**”.

Należy zatem wziąć pod uwagę, że nie jest trudno znaleźć przykłady kombinacji nawet dwu technik (np. sieci neuronowych i logiki rozmytej) których zastosowanie w wielu przypadkach nie jest zachęcające jeśli brać pod uwagę koszt osiągnięcia rozwiązania, tzn. że **nie obserwuje się wzrostu dokładności rozwiązań w akceptowalnym czasie**. Być może pewnym wyjaśnieniem tej sytuacji jest traktowanie przez niektóre źródła, np. Monostoriego [202] także **systemów rozmytych jako techniki przejściowej** między systemami eksperckimi a sztucznymi sieciami neuronowymi. Być może przyczyną problemów ze strukturami *NN-FL* jest **niedostateczne wyważenie** własności systemów rozmytych we współpracy z sieciami neuronowymi i zaprzepaszczenie takich **obiecujących cech hybryd NN-FL** jak [202]: zarządzanie informacją dostarczoną w postaci liczbowej i lingwistycznej, strukturalizowanie wiedzy i jej czytelność dla użytkowników, uczenie dotyczące zarówno wartości parametrów jak i struktury, automatyczny dobór reguł rozmytych, zdolność wyjaśniania wyników itp. Mimo tych powodów do zniechęcenia Ruggiero [253] wymienia następujące **zalety hybrydy NN-FS** w kontekście **analizy transakcji futures**: (1) możliwości prognozowania, (2) możliwość wyboru strategii, (3) uwzględnienie subiektywizmu decydenta.

Jeśli decydent nie rozumie problemu i poszczególnych algorytmów, to nie może odpowiedzialnie wybrać właściwego algorytmu i w takiej sytuacji **wybiera go praktycznie na ślepo**. Wyjściem z tej sytuacji według NFL jest zastosowanie wiedzy **dziedzinowej i doświadczenia** (*a priori*) i na ich podstawie sterowanie (wybraną uprzednio) **techniką poszukiwania rozwiązań** lub **podejmowanie decyzji o wyborze techniki poszukiwania rozwiązań**. Dlatego też **zawsze bezpieczną metodą jest równoległe wykonywanie obliczeń** z wykorzystaniem poszczególnych technik (bez sekwencyjności i sprzężenia zwrotnego) i podejmowanie decyzji na podstawie zestawienia wszystkich wyników. Pozwala to na dokonanie osądu przydatności poszczególnych technik przez decydenta (niestety efekt synergii nie jest możliwy wtedy do osiągnięcia). Pewnym wyjściem jest także **równoległe wyznaczenie rozwiązań** za pomocą pojedynczych technik **oraz** za pomocą hybrydy (musi być to jednak okupione większym zapotrzebowaniem na pamięć operacyjną, większym obciążeniem procesora lub przetwarzaniem równoległym). Następnie postępujemy zgodnie z zaleceniami Armstronga i Collopy'ego [13], czyli **podejmujemy decyzje odpowiednio do sytuacji** tzn. biorąc pod uwagę **sformułowane cele** i występujące **uwarunkowania** (takie jak dostęp do danych i wiedzy określonego typu omawiany przez Armstronga [13]. Przynajmniej należy **porównać ze sobą wnioski** sformułowane na podstawie **wyników poszczególnych struktur**. W przypadku **niezgodności** należy przeprowadzić **dogrywkę** - dla hybryd zawierających **moduły symulacyjne** przeprowadzić **analizę wrażliwości** lub wydłużyć czas symulacji. Można także **porównać wnioski** z wykorzystaniem analizy *ex ante* (szczególnie przydatne w przypadku hybryd zawierających **moduły symulacyjne**). Można także zastosować **dotatkową technikę nadzorującą pracę hybrydy** (w postaci mechanizmu sterowania eksperymentami) wyposażoną w możliwość podjęcia decyzji o modyfikacji struktury hybrydy w zależności od bieżącej sytuacji lub sytuacji przewidywanej.

4.6 Struktura hybrydy jako funkcja celów analizy

4.6.1 Uczenie maszynowe a uczenie ludzkie

Modyfikacja struktury hybrydy w skrajnym przypadku oznacza **możliwość odłączenia modułu** realizującego daną technikę (nieprzydatną w danej sytuacji lub negatywnie wpływającej na działanie pozostałych modułów; np. zwiększającej czas znalezienia rozwiązania) i powrót pozostałych do samodzielnego funkcjonowania. W ten sposób spełniony może zostać **postulat Kilgourów** [146] dotyczący **wykorzystania nisz** przez **poszczególne moduły hybrydy**, czyli zastosowania **takiej techniki i takiej struktury** która zapewnia wybór takiego sposobu wyznaczanie rozwiązania problemu który odpowiada bieżącym warunkom. W tym trybie **praca hybrydy** (jak i poszczególnych technik) **odwzorowuje postępowanie decydenta** będącego **jednocześnie ekspertem** w danej dziedzinie. Jednocześnie spełnione są postulaty Malhotry [183], który proponuje **rezygnację z konstruowania coraz bardziej złożonych baz wiedzy** oraz reguł wnioskowania i **skupienie się na procesie rozumowania człowieka, eksperta**. Potwierdza to Radosiński [246] pisząc że systemy eksperckie odniosły **sukces** w zastosowaniach mających na celu **automatyzację procedur**. Efektywność **zautomatyzowanych procedur** można zapewnić tylko wtedy, gdy **ścieżka (i proces) wnioskowania jest krótki** tzn. dochodzi do wywołania (sprawdzenia) jednej lub dwóch reguł wnioskowania. Jest to przesunięcie z uczenia maszynowego w kierunku **uczenia ludzkiego** i uznanie **konstruktywistycznego podejścia** "teorii osobistego wartościowania" (*personal construct theory*), która wywodzi się z obserwacji ludzkiego działania w obliczu **nowych informacji**. Według niej **nadawanie znaczenia nowym informacjom** (*sensemaking*) odbywa się w zgodzie z następującym schematem:

1. **Konfuzja** czyli **wrażenie niepokoju** lub dyskomfortu po napotkaniu sytuacji, której nie można wyjaśnić za pomocą rozumianych pojęć lub nie wiadomo jak sobie z nią poradzić
2. **Hipoteza** czyli jak poradzić sobie z niepewnością i zmniejszyć dyskomfort
3. **Działanie** czyli wcielenie w życie decyzji, prowadzące do uzyskania doświadczenia dotyczącego sposobów wykorzystania informacji
4. **Porównanie** (efektów wykonanych działań).

Według Malhotry [183] jest to **kierowane poznanie** (*guiding cognition*), podejście opozycyjne względem **przetwarzania informacji**. Wydaje się, że teoria osobistego wartościowania (*PCT*) jest zatem tożsama z symulacją. Malhotra powołuje się na Brunera [28], który zauważa, że człowiek stosuje zawieszenie niewiary (*suspend disbelief*) w celu **stworzenia wielu perspektyw** i "światów" a poszczególne osoby "aktywnie wybierają informacje, formy, hipotezy" aby zmniejszyć niepewność (tzn. niebezpieczeństwo zaskoczenia).

Chociaż - jak piszą Wermter i Sun [322] - w zasadzie **stopień integracji między technikami** składowymi można przedstawić na skali ciągłej, jako continuum, to w literaturze znaleźć można **szereg klasyfikacji struktur hybrydowych**. Można zatem zaproponować następujące propozycje **kryteriów taksonomicznych**, w zależności od:

1. Kierunku przepływu informacji – Kilgour
2. Kierunku sterowania (nadzorowania) – Radosiński
3. Typu relacji – Medsker (częściowo pokrywająca się z klasyfikacją wg kierunku sterowania)
4. Typu architektury wewnętrznej – Wermter i Sun
5. Zasady integracji – Goonatilake i Khebbal [99].

4.6.2 Klasyfikacje struktur hybryd wg kierunku przepływu informacji

W przypadku hybrydy łączącej więcej niż dwie techniki otrzymać można struktury o dużym stopniu złożoności. Kilgourowie [146] wyróżniają trzy sposoby integracji dwu odmiennych technik rozwiązywania problemów, czyli trzy rodzaje hybryd złożonych z **dwu składników**:

Hybrydę o strukturze sekwencyjnej, w której wartości wyjściowe z pierwszego składnika (pierwszej techniki) przekazywane są na wejście drugiego składnika (drugiej techniki). Zazwyczaj w takiej strukturze zadaniem pierwszej techniki jest **przetwarzanie wstępne**, np. oczyszczanie danych wejściowych lub wyznaczenie parametrów pracy drugiej techniki w celu suboptymalizacji. Można powiedzieć, że jest to struktura **najprostszą z możliwych**, hybrydyzacja odbywa się w **pętli otwartej, bez sprzężenia zwrotnego**.

Schemat struktury sekwencyjnej:

Wejście -> Technika A ->Technika B -> Wyjście

Hybrydę działającą na zasadzie wspomagania, w której wyraźnie wyróżniono **technikę** o charakterze **podstawowym** oraz **technikę pomocniczą**. Cechą charakterystyczną takiej struktury jest możliwość funkcjonowania techniki podstawowej **samodzielnie**, natomiast technika pomocnicza stosowana jest w celu **modyfikacji parametrów** techniki (składnika) podstawowego. Można w ten sposób dążyć do **optymalizacji** wyników działania techniki podstawowej wg ustalonego kryterium, które określa warunki i cel aktywacji techniki pomocniczej (a zatem technika pomocnicza może być **w pewnych warunkach nieaktywna**). Technika pomocnicza korzystać może także z tych samych danych wejściowych co technika podstawowa. Zawsze jednak wyniki techniki pomocniczej są przetwarzane przez technikę podstawową - występuje **sprzężenie zwrotne**. Uznany zastosowaniem tej struktury jest współpraca **sztucznych sieci neuronowych** (jako techniki podstawowej) z **algorytmami genetycznymi**, których zadaniem jest wyszukanie wartości parametrów sieci

neuronowej (struktury: liczby warstw, liczby neuronów, postaci funkcji aktywacji, strategii uczenia się) zapewniających optymalne wyniki (np. niską stopę błędnych klasyfikacji).

Schemat struktury ze wspomaganiami:

Wejście -> Technika podstawowa -> Wyjście

^

Technika pomocnicza -----

Hybrydy o **strukturze zagnieżdżonej** posiadają **najwyższy stopień zintegrowania**. W strukturze tej występuje znaczne i częste **przeplatanie się i wymiana informacji** między procedurami (blokami) zrealizowanymi z wykorzystaniem poszczególnych technik (wielokrotne sprzężenie zwrotne). W skrajnej postaci zintegrowanie polega na zastosowaniu jednolitego rozwiązania programowego, w którym obecna jest **pojedyncza ale funkcjonalnie złożona technika**. W strukturze hybrydy różnicuje się funkcjonalnie procedury takiej samej techniki w celu **emulowania** działania odmiennej (dodatkowej) techniki. Za pomocą połączenia **wielu implementacji** (procedur) pojedynczej techniki tworzy się w ten sposób **wirtualną strukturę** sekwencyjną bądź wspomagającą. Ciekawym przykładem struktury zagnieżdżonej w przypadku hybrydy sztucznej sieci neuronowej oraz logiki rozmytej może być emulacja logiki rozmytej za pomocą procedur sieci neuronowej. Zagadnienie to zostało opisane przez Tzafestasa i Stamou [306]. Przesłanką emulacji systemów rozmytych za pomocą sieci neuronowej były zaobserwowane, że między obiema technikami zachodzi **paralelizm**, czyli podobieństwo struktur i zdolność do estymacji bez konieczności dostępności modelu. Jak piszą autorzy, główną zaletą systemów działających na bazie emulacji FL przez NN jest możliwość **ograniczenia liczby niezbędnych reguł wnioskowania**, a tym samym przyspieszenie procesu otrzymywania wniosków (konkluzji). W strukturach FL-NN stosuje się co najmniej dwa typy rozwiązań: **koncepcja rozmytej macierzy relacji** – wymagająca znacznej pamięci oraz zapewnienia kompatybilności zmiennych (między modułem rozmywania a rozmytą macierzą) oraz **koncepcja zastosowania sekwencji rozmytych operacji**. Warto wspomnieć że ze względu na **konekcyjność i zdolność do generalizacji** sieci neuronowe przydatne są w emulowaniu pozostałych technik inteligentnych (podczas gdy np. regułowe systemy eksperckie zupełnie nie nadają się jako platforma emulacji).

Schemat hybrydy zagnieżdżonej:

Wejście -> Technika A + Technika B -> Wyjście

4.6.3 Klasyfikacje struktur hybryd wg typu relacji

Medsker [192] stosując kryterium relacji między współpracującymi technikami wyróżnia następujące **strategie projektowania** systemów hybrydowych:

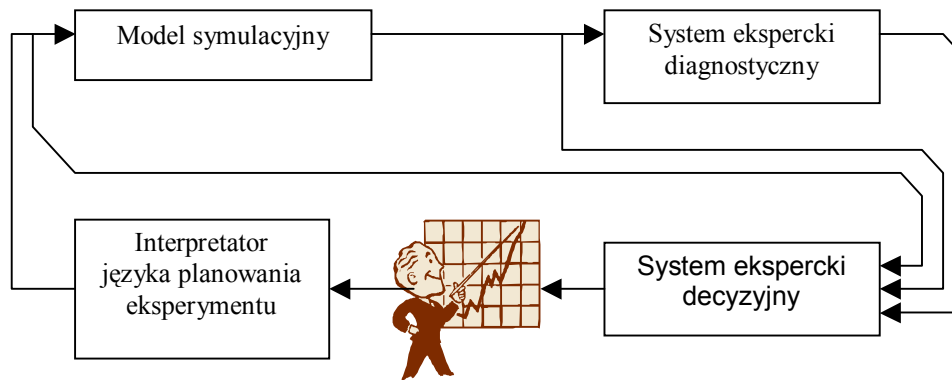
1. **Rozłączne** (*stand alone*); w systemach tych nie zakłada się i nie obserwuje żadnej integracji, komponenty (moduły) oprogramowania działają niezależnie od siebie
2. Przechodzące przeobrażenie, **transformacyjne** (*transformational*); między poszczególnymi technikami nie obserwuje się integracji, jednak procedura nadzorująca może wyłączyć bądź włączyć w pewnym momencie poszczególne techniki
3. **Luźno związane** (*loose coupling*); poszczególne techniki (moduły), mimo że działają oddzielnie, wymieniają między sobą informacje za pośrednictwem plików danych. Moduły pełnią funkcje przetwarzania wstępnego (*preprocessor*), finalizującego (*postprocessor*), obliczeń równoległych (*coprocessor*). Struktury typu *loose coupling* tworzą się także przez interfejsy użytkownika
4. **Ściśle związane** (*tight coupling*); poszczególne techniki (moduły), mimo że działają oddzielnie, wymieniają między sobą informacje za pośrednictwem wspólnego obszaru pamięci operacyjnej (w razie potrzeby także plików danych)
5. **Pełna integracja** (*full integration*); poszczególne techniki (moduły), mogą działać oddzielnie, natomiast korzystają z ze wspólnych struktur danych i reprezentacji wiedzy.

Systemy typu transformacyjnego mogą działać w dwu trybach: (a) **zachowującym uprawnienia modułu** nadzorującego w którym **rozpoczyna się działanie** (tzn. rozwiązywanie problemów) **aktywując** daną technikę (moduł), wykorzystując ją przez jakiś czas, do chwili dezaktywacji, następnie po spełnieniu określonego warunku **dezaktywują ją i aktywują inną technikę** (moduł) oraz (b) przekazującym uprawnienia do aktywnego modułu. W drugim przypadku jest to **co najmniej dwuetapowy, kaskadowy system** w którym rozwiązywanie problemów rozpoczyna się od **przesłania danych wejściowych i aktywacji techniki** (modułu) etapu początkowego a kończy po wykonaniu obliczeń przewidzianych do wykonywania przez moduł początkowy **przekazaniem danych wynikowych** z modułu początkowego do **techniki** (modułu) **etapu finalnego**. Przykładem działania w trybie z zachowaniem uprawnień (a) są systemy w których należy dobrać technikę poszukiwania rozwiązania do charakteru danych wejściowych (np. systemy wczesnego ostrzegania / inteligentni agenci, których reakcja na symptomy niebezpieczeństwa polega na uruchomieniu procedur alarmowo-ratunkowych, natomiast w zwykłych warunkach zajmują się prowadzeniem zwykłej działalności, np. obsługą transakcji w przedsiębiorstwie). Przykładem działania w trybie z przekazywaniem uprawnień (b) są systemy w których etap początkowy polega na usuwaniu zakłóceń.

Charakterystycznymi cechami **struktur luźno związanych** są (a) wyposażenie poszczególnych modułów w procedury wejścia-wyjścia operujące na plikach, (b) opracowanie procedur wykonujących

translację do i z wewnętrznych formatów danych, (c) brak zagnieżdżeń wywołań (jeśli technika (moduł) A wywołała technikę B, to do zakończenia obliczeń przez technikę B technika A nie może być wywołana przez technikę B). **Inicjacja odczytu plików danych** ma miejsce dopiero wtedy gdy **moduł wymaga informacji** pochodzących z pozostałych modułów (w przypadku jej nieobecności moduł zawieszona działanie – oczekuje na udostępnienie danych). Wermter i Sun [322] podkreślają **sekwencyjne działanie** tego typu struktur, rozumiane jako konieczność zakończenia zadania przez aktywny moduł zanim będzie możliwe aktywowanie innego modułu. Ze względu na komunikację **wyłącznie za pośrednictwem plików** wadami struktur typu *loose coupling* są: konieczność opracowania **procedur translacji**, **trudności w realizacji interaktywności** (jeden moduł może zainicjować działanie drugiego i musi czekać na wykonanie przypisanych mu operacji – chyba że istnieje możliwość oszacowania czasu obliczeń). Zaletami struktury „luźnych relacji” jest jej przydatność w konstruowaniu **prototypów systemów hybrydowych** oraz możliwość **łatwej integracji narzędzi** dostępnych w postaci oddzielnych modułów oprogramowania (programów wykonywalnych) za pomocą dodatkowych modułów translacji formatów plików.

Struktury luźno powiązane są także **najkrótszą drogą** do utworzenia struktur hybrydowych typu algorytm genetyczny – sieć neuronowa, logika rozmyta – sieć neuronowa, symulacja – optymalizacja. Potwierdzają to Wermter i Sun [322] pisząc że „(hybrydy oparte na) luźnych powiązaniach pozwalają na różnorodne formy luźnej współpracy między modułami”. Wyróżnili oni następujące przypadki: (a) **technika główna** (np. oparta na reprezentacji symbolicznej) wykonuje zasadnicze zadanie a **techniki uzupełniające** (np. sieć neuronowa) wykonują przetwarzanie wstępne lub służą do przygotowania prezentacji wyników; (b) praca w **trybie master-slave: technika główna** aktywuje w **razie potrzeby** poszczególne techniki uzupełniające; technika główna **dobiera technikę wykonawczą** w zależności od zadania i odpowiedniej do niego reprezentacji wiedzy, (c) równouprawnienie i współpraca poszczególnych modułów. Przykładem struktury *loose coupling* jest hybryda diagnozująca urządzenie zapobiegające niedotlenieniu wody [283] w której logika rozmyta dokonuje wstępnego przetwarzania danych w celu skonstruowania wektora cech. Następnie przeprowadza się klasyfikację: wektor cech przyporządkowuje się jednej z wielu klas (stanów systemu). Zadaniem drugiego komponentu, sieci neuronowej jest klasyfikacja stanów procesu i zidentyfikowanie występujących usterek lub stanów niebezpiecznych (groźących pojawieniem się usterek). Moduł logiki rozmytej wykorzystuje się do wygenerowania ciągu uczącego. Wg autorów rozwiązania zapewnia się w ten sposób rozpoznanie takich sytuacji jak tworzenie się piany, błędne sterowanie temperaturą czy zatkanie przewodów doprowadzających wodę. Schemat hybrydowego SWD Ekanwin działającego jako struktura luźno powiązana przedstawiono na rysunku [Rys. 4-1] – szczegóły w dalszej części pracy.



Rys. 4-1 Zamknięta pętla MS-SE-Operator-Interpretator

W hybrydach złożonych z modułów ściśle powiązanych **struktury danych** do których odwołują się poszczególne moduły są **wspólne** dla współpracujących modułów i **nie wymagają** wykonania operacji **translacji**. **Zaletami** tego typu struktur hybrydowych są: łatwiejsze implementowanie **przetwarzania równoległego**, szybsza praca, **ujednoczenie formatów danych**, możliwość implementacji poszczególnych technik (modułów) w postaci **inteligentnych agentów**. Wśród **niedogodności** warto wymienić: **konieczność** integracji w systemie **modułu zarządzającego** pracą technik (modułów) inteligentnych, konieczność dostępu do **kodu źródłowego** modułów, ewentualne problemy w reprezentacji danych dostępnych w modułach tworzonych w **niekompatybilnych narzędziach programowania**, trudności w wykorzystaniu przetwarzania wsadowego, **mniejsza elastyczność** w konfigurowaniu hierarchii i kolejności technik, **trudności w sterowaniu procesem komunikacji** ze współdzieleniem pamięci.

W systemach **w pełni zintegrowanych** komunikacja może odbywać się na poziomie (a) **współdzielenia plików**, (b) **współdzielenia obszaru pamięci** (zmiennych) przechowującej dane wejściowe i wyjściowe oraz dane sterujące, (c) **współdzielenia reprezentacji wiedzy** tzn. **dostępu** do ścieżek wnioskowania, procedur optymalizacji, chromosomów, neuronów itd. **Wyciąganie i formułowanie wniosków** wykonywane jest na podstawie **współdziałania** modułów lub odpowiedzialny za to jest **wyróżniony moduł**. Moduł wyróżniony (nadzorujący) może być **powołany specjalnie** w tym celu lub korzystać z jednej z technik inteligentnych. Przykładem w pełni zintegrowanego systemu jest hybryda logiki rozmytej z siecią neuronową opisana przez Gabryśia i Bargielę [89]. Jest to system rozpoznawania przecieków w instalacji wodociągowej. Musi zatem poradzić sobie zarówno z błędami pomiaru, korzystać z prognoz zapotrzebowania na wodę jak i identyfikować ubytki wody (przecieki). Projekt został zrealizowany jako **system dwupoziomowy**: pierwszy poziom dotyczy **rozpoznania typowego zachowania się systemu** (np. zużycie nocne, szczyt dobowy) natomiast drugi ma za zadanie wykryć **zachowania nadzwyczajne** – anomalie. W strukturze *NN-FZ* węzły w centralnej warstwie sieci neuronowej (trójwarstwowej) reprezentują hipersześcian wykorzystujący logikę rozmytą: zastosowano klasyfikację min-max a funkcja przejścia jest tożsama z funkcją przynależności (hipersześcianu). Celem uczenia sieci jest umożliwienie klasyfikacji za

pomocą hipersześcianu (stan normalny/typ zachowania/stan nadzwyczajny-przeciek). Schemat w **pełni zintegrowanej hybrydy NN-FZ** [285] składa się z trzech części: warstwy rozmywania, reguł rozmytych oraz defuzyfikacji (transformacji wartości rozmytych (stopniowanych) do postaci wartości liczbowych (ze zbioru liczb rzeczywistych)). Wszystkie części są zrealizowane w technice sieci neuronowych. W **warstwie rozmywania** funkcja rozmyta (typu $If X \geq A$) transformuje poszczególne rodzaje wartości wejściowych (cech o charakterze ciągłym, pochodzące z pomiarów wykonanych z wykorzystaniem przyrządów czy urządzeń) do postaci rozmytej (stopniowane, o charakterze dyskretnym, nazywane zmiennymi lingwistycznymi). Za realizację funkcji przynależności odpowiadają neurony. Następna warstwa obejmuje **reguły rozmyte** operujące na wartościach wyjściowych z poprzedniej warstwy przemnożonych przez odpowiednie wagi (dla każdej wartości rozmytej i dla każdej cechy wejściowej). Za realizację funkcji przynależności tu także odpowiadają neurony a wyjściowa wartość reguły rozmytej jest ustalana w procesie uczenia (za pomocą ciągu uczącego). **Warstwa defuzyfikacji** składa się z neuronów obliczających wartości wyjściowe. **Funkcja przynależności** realizowana jest poprzez połączenie funkcji sigmoidalnych i funkcji liniowych. Wartości wag wyjściowych, ustalane podczas procesu uczenia sieci, wskazują na punkt ciężkości poszczególnych funkcji przynależności. Jako przykłady w pełni zintegrowanych struktur Wernter i Sun [322] podają *CONSYDERR*, w którym poszczególne elementy (reguły) w modułach mają swoje odpowiedniki w innych modułach, jest to jednak hybryda dwu typów sieci neuronowych (lokalnej i rozproszonej) oraz złożoną z wielu modułów hybrydę *CBR-NN* skonstruowaną przez Leesa i innych [167]. Opracowano także inne propozycje dotyczące schematów w pełni zintegrowanych struktur NN-FZ; autorem jednego z nich (pięciowarstwowej) jest Lin [177].

4.6.4 Klasyfikacje struktur hybryd wg kryterium sterowania

Radośniński [246] rozważając kombinacje dwuskładnikowe, przyjmuje jako kryterium klasyfikacji relacje między poszczególnymi technikami i wyróżnia struktury hierarchiczne, rozłączne i homomorficzne. W przypadku struktur (hybryd) **hierarchicznych** jedna z technik (modułów) nazywana jest **techniką główną** a druga **techniką uzupełniającą**.

Przyjmuje się następujące oznaczenie: **ITG(główna)**->**ITU(zupełniająca)**.

Technika główna **steruje** procesem rozwiązywania problemu i **w razie potrzeby aktywuje** technikę uzupełniającą, przesyłając do niej odpowiednie dane, udostępniając jej czas procesora (w przypadku możliwości obliczeń równoległych) lub przekazując sterowanie. Po wykonaniu zleconego zadania przez technikę pomocniczą **udostępnia wyniki** technice (modułowi) głównej, dezaktywuje się i **oddaje sterowanie**. W literaturze donoszono [20], [99], [192] o następujących **pozytywnie ocenianych kombinacjach (hybrydach) technik**: *NN* -> *AG* (algorytm genetyczny służy wyznaczaniu wag sieci), *ES* -> *NN* (sieć neuronowa rozstrzyga konflikty, sieć neuronowa

odpowiedzialna jest za dopasowanie interfejsu do wymagań użytkownika), *ES* -> *AG* (algorytm genetyczny wyznacza wartości współczynników pewności), *ES* -> *FS* (*FZ*) (rozmyta reprezentacja wiedzy, wnioskowanie rozmyte), *FS* -> *NN* (zadaniem sieci neuronowej jest optymalizacja sterowania rozmytego poprzez ustalenie hierarchii ważności reguł), *FS* -> *AG* (zadaniem algorytmu genetycznego jest optymalizacja sterowania rozmytego), *NN* -> *ES* (system ekspercki wyjaśnia rozwiązania wyznaczone przez sieć neuronową, wykorzystanie zdolności wyjaśniających i uczenia się), *AG* -> *ES* (system ekspercki wyjaśnia rozwiązania wyznaczone przez algorytm genetyczny), *NN* -> *FS* (dzięki technice uzupełniającej wzbogacono sieć neuronową o parametry rozmyte), *AG* -> *FS* (dzięki technice uzupełniającej wzbogacono algorytm genetyczny o parametry rozmyte).

Przykładem **hierarchicznej hybrydy** *SE*->*NN* jest system *HYBEXP* opisywany przez Monostorigo i innych [203]. Jest to system zarządzania produkcją, w którym **moduł systemu eksperckiego** wykorzystuje informacje otrzymane z **modułu sieci neuronowej** (monitorującego proces produkcji) a także bazy danych o maszynie wykorzystywanej w procesie produkcji, wytwarzanych elementach oraz wymaganych i dostępnych narzędziach. Przykładowo, jeśli sieć neuronowa rozpozna objawy zużycia narzędzia, to system ekspercki powinien rozważać: szacowany pozostały czas przydatności narzędzia, niebezpieczeństwo niezakończenia operacji z powodu usterki, dostępność narzędzi zapasowych, liczbę elementów oczekujących na obróbkę do czasu przywrócenie sprawności maszyny, koszty wymiany narzędzi itd. Wymienić można szereg zalet podejścia *SE*->*NN* w omawianym systemie produkcyjnym ale wydaje się, że najważniejszym jest **dualne określanie czasu** pozostającego do końca **żywności narzędzia**: poprzez **monitoring** (sieć neuronowa) oraz **techniki analityczne i wnioskowanie symboliczne** (system ekspercki).

O **hybrydach rozłącznych** (nazywanych przez Ultscha [309] systemami kooperującymi (*cooperative systems*) mówimy gdy poszczególne techniki komunikują się między sobą **za pomocą przekazywania struktur danych** i **nie są od siebie zależne strukturalnie**. Oznacza to, że żadna z technik (modułów) **nie ma możliwości sterowania inną**, nie może aktywować, dezaktywować czy spowodować uruchomienie transmisji danych przez inne techniki (moduły). **Sterowaniem** poszczególnymi technikami oraz **zarządzaniem procesem transmisji danych** zajmuje się **operator** lub **nadrzędna procedura** sterująca (oprogramowanie). Jak pisze Radosiński [246] **struktura rozłączna** jest odpowiednia dla złożonych problemów które **da się podzielić na podproblemy**. Taki podział powinien uwzględniać możliwość **dopasowania technik** do poszczególnych podproblemów. Hybryda rozłączna może sprawdzać się gdy rozwiązywanie danego problemu **wymaga działań wieloetapowych**. W takiej sytuacji konfiguruje się poszczególne techniki (moduły) w taki sposób, że **aktywują wzajemną komunikację** wtedy gdy odpowiednio **kończą** lub **rozpoczynają właściwy etap obliczeń**. Wydaje się, że natura **hybryd rozłącznych** predestynuje je do rozwiązywania problemów poprzez **przetwarzanie rozproszone** (węzły sieci komputerowej realizują rozwiązywanie za pomocą poszczególnych technik). Spodziewać się tu można **niewiele większego stopnia trudności** niż w

przypadku oprogramowania i zorganizowania współpracy między poszczególnymi modułami działającymi na tym samym komputerze (praktycznie wystarczy wykorzystać określony protokół sieciowy). Doniesienia literaturowe [236] mówią np. o **rozłącznej hybrydzie** $SE \leftrightarrow \text{optimalizacja} \leftrightarrow AG$ poprawiającej parametry działania turbiny, hybrydy $NN \leftrightarrow AG \leftrightarrow SE$, skonfigurowanej w celu rozwiązywania zadania wieloetapowego, składającego się z kolejno przeprowadzanych operacji: **rozpoznawania wzorców** (sieć neuronowa), **optymalizacji** (algorytm genetyczny) i **objaśniania wyników** za pomocą mechanizmu wnioskowania opartego na regułach (regułowy system ekspercki), czy hybrydy NEURECA $NN-FL-GA$ opisywanej przez Monostoriego [204], w której algorytm genetyczny **dobiera reguły wnioskowania** a sieć neuronowa i logika rozmyta służą **prognozowaniu bankructwa**.

Klasa **hybryd homomorficznych** wyodrębniona na podstawie kryterium relacji przedstawionego przez Radosińskiego pokrywa się z klasą jednolitych architektur (w terminologii Wermtera i Suna). Do hybryd homomorficznych zaliczyć należy także **hybrydy transformacyjne** (w terminologii Medskera), jako ich szczególny przypadek. Cechą wyróżniającą struktur homomorficznych jest **zróznicowanie wyłącznie pod względem funkcjonalnym**: wszystkie techniki składające się na hybrydę implementowane są w za pomocą pojedynczej, uniwersalnej techniki. Ze źródeł literaturowych wynika, że platformą implementacji dla różnorodnych technik (SE, GA, FL itd.) są sieci neuronowe, ze względu na ich zdolność uczenia się i zdolność do odwzorowania wiedzy w postaci konekcyjnej – poprzez zmiany struktur (często wystarczy zmiana wag synaptycznych w sieci). Przykładowo, reguły wnioskowania mogą być **emulowane** w strukturze sieci neuronowej poprzez sieć relacji z przypisanymi (uprzednio) wagami synaptycznymi. Jak pisze Radosiński [246] „pełna integracja wiedzy eksperckiej i parametrów sieci zapobiega redundancji i ułatwia weryfikację zapisanych informacji”.

4.6.5 Klasyfikacje struktur hybryd wg typu architektury wewnętrznej

Rozszerzona taksonomia hybryd została przedstawiona przez Wermtera i Suna [322] w oparciu na klasyfikacji Medskera. **Kryterium** w tym przypadku była **wewnętrzna architektura** hybrydy. Uwzględniono w ten sposób sytuację, gdy odpowiednio skonfigurowana złożona struktura oparta na **pojedynczej technice inteligentnej** realizuje funkcje charakterystyczne dla innej techniki, czyli ją **emuluje** (posiada elementy i zachowuje się w sposób charakterystyczny dla symulowanej techniki). Wermter i Sun wyróżniają **trzy zasadnicze klasy architektur wewnętrznych** w przypadku hybryd, w których jednym ze składników jest **sztuczna sieć neuronowa**: (a) jednolita architektura neuronowa (*unified neural architecture*), (b) architektura oparta na wymianie reprezentacji wiedzy (*transformation architecture*), (c) modułowa architektura hybrydowa (*hybrid modular architecture*).

Cechą charakterystyczną **architektury jednolitej** jest realizowanie zadań przetwarzania symbolicznego za pomocą odpowiedniej konfiguracji neuronów, czyli **emulacja systemów regulowych**. W ten sposób sieć neuronowa zyskuje przede wszystkim **możliwości objaśniające i logicznego wnioskowania**, czyli uwzględnienia kontekstu informacji. Jak zwykle w sieciach neuronowych, **struktury przetwarzania symbolicznego** budowane są podczas uczenia sieci. Można wyróżnić dwa typy architektury jednolitej: lokalny (*localist connectionist architecture*) oraz rozproszony (*distributed neural achitecture*). W pierwszym typie **każdy węzeł** reprezentuje **określone pojęcia** (związki). Jak piszą Wermter i Sun [322] reprezentacja lokalna przydatna jest w zadaniach **modelowania kognitywnego** (chodzi tu o zdolność uczenia się), wykorzystując regułę spadku gradientu (wsteczną propagację błędu), metodę maksymalizacji wartości oczekiwanej itp.

W **architekturze rozproszonej** poszczególne pojęcia (symboliczne) reprezentowane są przez **struktury neuronów**, obecne w całej sieci neuronowej. Jest to **podejście radykalne**, zakładające reprezentację całości problemu i metod jego rozwiązywania za pomocą sieci neuronowej. Według zaleceń metodologicznych **architektury jednolitej ES-NN** na platformie sieci neuronowych, konstrukcja systemu ma wspomagać proces analizy sprawozdań finansowych [216].

W architekturze opartej na **wymianie reprezentacji wiedzy** (*transformation architecture*) mamy do czynienia z techniką (reprezentacją) nadrzędną (główną) i podrzędną (uzupełniającą). Rolę **techniki głównej** pełni sieć neuronowa a **uzupełniającej** reguły system ekspercki. Podstawą wymiany reprezentacji wiedzy między poszczególnymi technikami jest wyposażenie hybrydy w **zestaw procedur wzajemnie tłumaczących odmienne reprezentacje wiedzy** (konekjonistyczną i symboliczną). Umożliwiają one uniwersalne korzystanie z obu typów reprezentacji wiedzy. Celem omawianego typu architektury systemów hybrydowych jest **wydobywanie reguł wnioskowania** z sieci neuronowych poddanych uprzednio procesowi uczenia się. Możliwe jest także **wprowadzanie wiedzy symbolicznej** (reprezentowanej za pomocą reguł) do struktur sieci neuronowych. Można to wykonać stopniowo (przez uczenie sieci neuronowej) lub błyskawicznie (przez dołączenie struktury neuronowej reprezentującej reguły wnioskowania). Warto zaznaczyć, że Ultsch [309] uznaje systemy hybrydowe zaprojektowane zgodnie z architekturą opartą na **wymianie reprezentacji wiedzy** oraz **modułowe architektury hybrydowe** za **właściwe systemy hybrydowe** (*true hybrid systems*).

Wermter i Sun [322] rozważając klasę **modułowych architektur hybrydowych** wyróżniają **trzy typy systemów** (ze względu na kryterium klasyfikacji nazywają je architekturami): **luźno związane, ściśle związane** i **w pełni zintegrowane**. Typologia ta pokrywa się z trzema typami struktur (wyższych) według Medskera. Autorzy podkreślają jednak **realizację** poszczególnych technik w postaci **modułów programowych** oraz że są to architektury **jakościowo wyższe** od architektury opartej na **wymianie reprezentacji wiedzy**. Motywują to faktem, że w *transformation architecture* poszczególne techniki wymieniają **jedynie** między sobą **reprezentację wiedzy**, pomijając problem sterowania i rozdziału zadań w zależności od typu problemu.

4.6.6 Klasyfikacje struktur hybryd wg zasad integracji

Goonatilake i Khebbal [99] twierdzą, że opracowanie **hybryd** różnorodnych technik umożliwi **rozwiązywanie dużo trudniejszych problemów**. Są zwolennikami **dekompozycji** złożonych problemów na podproblemy, rozwiązywania ich przez poszczególne techniki inteligentne przy jednoczesnym **komunikowaniu się modułów** (technik) między sobą. Goonatilake i Khebbal dokonując klasyfikacji systemów hybrydowych brali pod uwagę **zasady integracji jako kryterium**. Dzięki ich klasyfikacji wyróżniono trzy rodzaje struktur: hybrydy na bazie funkcji (*technique enhancement*), hybrydy na bazie komunikacji (*intercommunicating hybrid intelligent systems*) oraz hybrydy na bazie struktury (*polymorphic hybrid intelligent systems*).

W **hybrydach na bazie funkcji** w przypadku zidentyfikowania danego elementu czy funkcji techniki jako **mniej przydatnego** czy efektywnego organizuje się **zastąpienie** go przez **inną technikę** (np. gdy należy zidentyfikować wzorzec dezaktywuje się moduł systemu eksperckiego i aktywuje moduł sieci neuronowej a gdy istnieje potrzeba sporządzenia raportu dezaktywuje się moduł sieci neuronowej i przekazuje sterowanie modułowi regulowego systemu eksperckiego). Silva i Ludermir [273] opisują hybrydę na bazie funkcji, w której sieć neuronowa **wybiera reguły wnioskowania** oraz **dobiera parametry reguł**, ograniczając liczbę reguł w SE. Jak podaje Radosiński [246] w literaturze znaleźć można wiele doniesień o wdrożonych hybrydach działających **na bazie funkcji**: (1) sieć neuronowa przekazuje zadanie **wyznaczania wag synaptycznych** algorytmowi genetycznemu, (2) algorytm genetyczny deleguje zadanie **wyboru populacji początkowej** modułowi sieci neuronowej, (3) regułowy system ekspercki dokonuje **analizy procesu wyznaczania rozwiązania** przez moduł sieci neuronowej i komentuje uzyskane wyniki.

Hybrydy na bazie komunikacji wydają się najbardziej naturalnym rozwiązaniem. Poszczególne rodzaje reprezentacji wiedzy wykorzystywane są przez odpowiednie techniki (moduły) inteligentne. Przykładowo, wiedza ekspercka w postaci dobrze zdefiniowanych reguł będzie przechowywana w bazie wiedzy systemu eksperckiego, dane o charakterze rozmytym, lingwistycznym w systemie rozmytym lub rozbudowanej o reprezentację wiedzy rozmytej sieci neuronowej. Cechą charakterystyczną **hybryd na bazie komunikacji** jest występowanie **modułu realizującego transmisję danych**, niekiedy także **sterowanie za pośrednictwem kanału komunikacji** (pliki dyskowe, protokoły sieciowe, wspólny obszar pamięci). Hybrydy na bazie komunikacji przystosowane są **do rozwiązywania złożonych problemów zdekomponowanych na podproblemy** (kryterium dekompozycji są własności poszczególnych technik dostępnych w hybrydzie). Ten rodzaj hybrydy przydatny jest także w rozwiązywaniu zadań polegającym na **kolejnym wykonywaniu szeregu zadań bardziej elementarnych** np. rozpoznawaniu wzorców (sieć

neuronowa), optymalizacji (algorytm genetyczny), wnioskowaniu symbolicznym (system ekspercki). Poszczególne moduły, wykonujące dane zadanie elementarne **przekazują sobie wzajemnie dane wyjściowe** (wyniki rozwiązań pochodzące z danego modułu). Należy przypomnieć, że wg terminologii Medskera są to hybrydy przechodzące przeobrażenie (*transformational*). Jeżeli na przykład do analizy danego zagadnienia należy wykonać operacje **rozpoznawania wzorów, optymalizacji**, a następnie **wnioskowania symbolicznego**, wtedy zadanie to może być wykonane przez strukturę hybrydową zawierającą trzy, wzajemnie skomunikowane techniki inteligentne: **sieć neuronową, algorytm genetyczny i system ekspercki**.

Cechą charakterystyczną **hybryd na bazie struktury** nazywanych także **polimorficznymi** jest **emulowanie funkcji i procesów** wykonywanych zwykle przez różnorodne techniki na **jednolitej platformie** programistycznej. W przypadku struktur **monolitycznych** (niepolimorficznych) dla wszystkich modułów obserwuje się jednolitość funkcji i struktury, tzn. moduł reprezentujący daną technikę wykonany jest bezpośrednio z wykorzystaniem pojęć danej techniki. W literaturze podaje się szereg przykładów **struktur polimorficznych**; zazwyczaj platformą programistyczną (emulującą pozostałe techniki) są **sieci neuronowe**, np. emulacja systemu eksperckiego przez sieć neuronową, wnioskowanie symboliczne za pomocą sieci neuronowej [6], sieć neuronowa (typu Boltzmanowskiego, wykorzystująca symulacyjne wyżarzanie, wyposażona w możliwości reprezentacji danych i pojęć) emulująca algorytm ewolucyjny [1] czy samoorganizujące się mapy własności (odmiana sieci neuronowych) emulujące struktury pojęć podobne do używanych w języku Prolog [307], różnorodne metody emulacji poszczególnych funkcji systemu eksperckiego przez sieć neuronową [308]. Struktury polimorficzne są tożsame z typem hybryd jednolitych w klasyfikacji Wermtera i Suna [322].

5 Założenia hybrydowego SWD Ekanwin

5.1 System wspomagania decyzji – Ekanwin – założenia

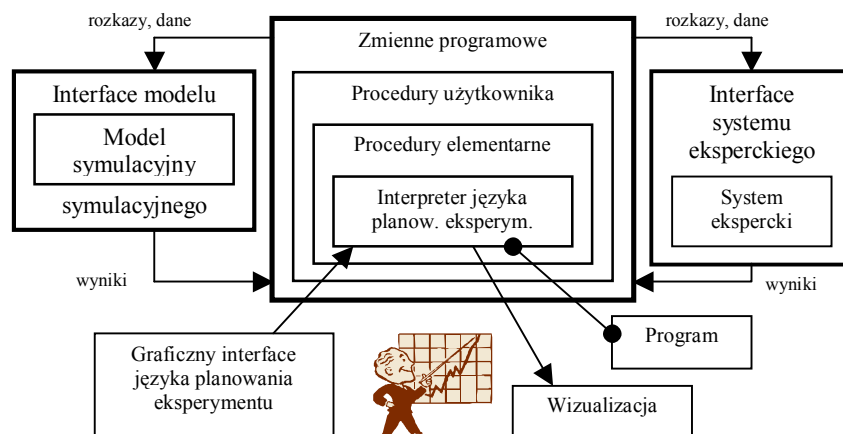
Zastanówmy się teraz, czego użytkownik-operator oczekuje od **inteligentnego hybrydowego systemu wspomagania decyzji**? Odpowiedź jest następująca: oczekuje wsparcia następujących podstawowych działań:

- Diagnostyki ekonomicznej
- Poszukiwania zadowalających rozwiązań
- Korzystania z eksperckiej bazy wiedzy.

Można wyróżnić następujące szczegółowe czynności prowadzące do udanej ich realizacji:

- Analityczny: Ocena sytuacji ekonomicznej przedsiębiorstwa (stan, trendy i prognozy)
- Analityczny: Identyfikacja mocnych i słabych stron jednostki gospodarczej
- Projektowy: Przygotowanie i ocena alternatywnych wariantów (propozycji) decyzji
- Analiza wykonalności i optymalizacja: Poszukiwanie zadowalających rozwiązań
- Konfigurowanie i sterowanie eksperymentem symulacyjnym i technikami sztucznej inteligencji (posługiwanie się interfejsem użytkownika, zarządzanie bazami danych i bazą wiedzy, opracowanie tablic decyzyjnych, ustalanie wartości parametrów algorytmów i eksperymentów),
- Uruchamianie procedur sztucznej inteligencji automatycznie - system wczesnego ostrzegania o zagrożeniach lub informowania o pojawiających się możliwościach względnie okresowo oraz na żądanie,
- Aktywacja technik sztucznej inteligencji przez podsystem modelu ekonomicznego (w trakcie symulacji),
- Wywoływanie procedur podsystemu modelu ekonomicznego przez techniki sztucznej inteligencji (moduł sztucznej inteligencji automatycznie konstruuje plan eksperymentu, korzysta z technik obliczeniowych, analitycznych i prezentacji graficznej).

Poniżej przedstawiono schemat środowiska realizującego założenia dot. funkcji i usług [Rys. 5-1].



Rys. 5-1 Projektowany schemat interakcji hybrydowego systemu wspomagania decyzji: wersja model symulacyjny + system ekspercki. Źródło: [335]

5.2 Przesłanki modułowości hybrydowego SWD Ekanwin

Ekanwin zaprojektowany został jako modułowy system wspomaganie decyzji. Analiza ekonomiczna wykorzystywana w praktyce zarządzania przedsiębiorstwem stanowi podstawę podejmowania decyzji. Jednak ze względu na jej złożoność i uświadomiony nieustrukturalizowany charakter relacji między wynikami analiz a podejmowanymi decyzjami prowadzone są próby stworzenia odpowiedniego aparatu badawczo-eksperymentalnego. Powinien on być zdolny do zmierzenia się ze systemową złożonością przedmiotu i podmiotu zainteresowania analizy ekonomicznej. W tym celu połączono odmienne podejścia bazowane na technologii informacyjnej – techniki sztucznej inteligencji (realizujące postulat wspomaganie podejmowania decyzji), konwencjonalny system wspomaganie decyzji oraz symulacji (weryfikatora projektowanych decyzji) w celu uzupełnienia standardowych metod analizy ekonomicznej.

Konwencjonalny system wspomaganie decyzji zawiera następujące moduły funkcjonalne: model sytuacji biznesowej (w szczególnym przypadku model jednostki gospodarczej), bazę danych (ukonkretniającą „pusty” model, umożliwiającą wykonanie obliczeń zgodnie z modelem), biblioteki metod obliczeniowych, analitycznych i graficznych oraz komunikacji (dialogu) z użytkownikiem. Wrycza [329] wskazuje, że zadaniem bazy modeli jest udostępnianie różnych metod (programów i procedur) stosowanych w zależności od używanych danych w modelach strategicznych, taktycznych i operacyjnych.

Modułowa struktura systemu wspomaganie decyzji umożliwia rozszerzenie jego własności analityczno-decyzyjnych. Dzieje się tak dzięki zastosowaniu informatycznych implementacji różnorodnych metod i technologii sztucznej inteligencji i przetwarzania danych. Według Swaina [290] włączenie symulacji do szerszego kontekstu wsparcia analizy możliwe jest dzięki integracji, która przejawia się coraz większą zdolnością do wymiany informacji między modułem modelu symulacyjnego, sztuczną inteligencją oraz zasobami informacyjnymi przedsiębiorstwa.

Repertuar technik które mogą stać się podstawą modułu (lub nawet wielu modułów) sztucznej inteligencji jest bardzo szeroki. Wystarczy wymienić uczenie maszynowe, odkrywanie wiedzy i wydobywanie danych (*Learning, Knowledge Discovery, and Data Mining*), systemy eksperckie, *tabu search, case-based reasoning*. Należą do nich narzędzia określane jako *soft computing* względnie *computational intelligence* tzn. algorytmy genetyczne, sztuczne sieci neuronowe, symulacyjne wyżarzanie, logika rozmyta, algorytmy ewolucyjne. Integracja modułów modelu symulacyjnego i sztucznej inteligencji jest możliwa dzięki zastosowaniu interfejsu komunikacji między modułami. Jego podstawową funkcją jest dialog między modułami, odbywający się wg zaprojektowanego protokołu. Interfejs międzymodułowy oferuje dostęp do wybranych funkcji elementów poszczególnych modułów, w zależności od aktualnych potrzeb. Zauważyć tu można analogię do interfejsu użytkownika omawianego przez Wryczę [329] zapewniającego manipulację elementami bazy

modeli i bazy danych.

Użyteczność i elastyczność systemu wspomagania decyzji podnosi **wbudowanie modelu przedsiębiorstwa**. Pozwala to na przeprowadzanie eksperymentów polegających na obserwacji prognozowanych zachowań (wyników ekonomicznych) przedsiębiorstwa w odpowiedzi na wprowadzone przez eksperymentatora zestawy wartości zmiennych decyzyjnych. **Model symulacyjny** dostarcza zatem danych o **wynikach funkcjonowania przedsiębiorstwa**. Macierze zmiennych decyzyjnych odwzorowują parametry decyzji w horyzoncie operacyjnym, taktycznym i strategicznym dotyczących produkcji, inwestycji, finansowania, zaopatrzenia, sprzedaży, marketingu itd. Kleiber [150] zauważa, że teoria, modelowanie wraz z symulacją komputerową oraz eksperyment tworzą **trzy filary współczesnej nauki**. Jak argumentuje Martin [187] „symulacja nadaje nowe wymiary eksperymentom... ponieważ umożliwia obserwację wielu bardzo małych, bardzo powolnych lub bardzo szybkich zmian, które normalnie nie nadają się do obserwacji. Skala czasu może zostać tysiące razy wydłużona lub skrócona”.

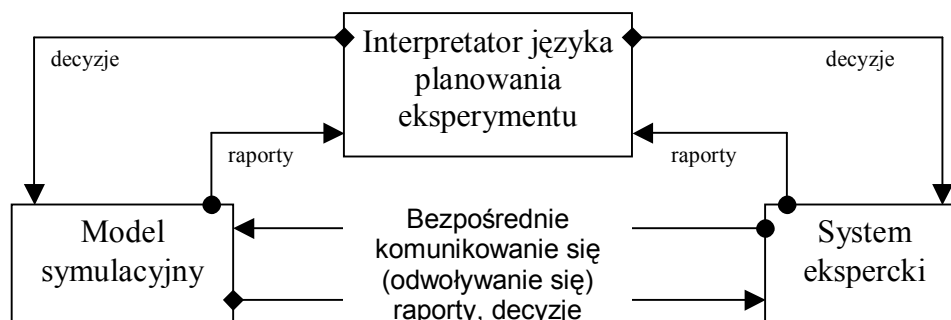
Modułowa konstrukcja SWD Ekanwin umożliwia realizację niektórych własności **architektury klient-serwer**. Opis poszczególnych zalet tej architektury w porównaniu z podejściem zcentralizowanym znaleźć można w pracy Kobsy [151]. **Moduł modelu symulacyjnego** zawarty w EKANWIN pełni rolę **serwera**. Obsługuje on **szereg klientów** - narzędzi symulacyjnych, analitycznych, optymalizacyjnych i wizualizacyjnych. Narzędzia te zostały zrealizowane w postaci **modułów**. Aktywacja, sterowanie i synchronizacja pracy poszczególnych narzędzi realizowane są za pomocą graficznego systemu menu, skrótów klawiaturowych, aktywnych elementów ekranowych (suwaki, pola wyboru itp.) oraz poleceń języka planowania eksperymentu. Strona klienta w ostatnim przypadku reprezentowana jest przez program tworzony w edytorze tekstowym. Ekanwin udostępnia także narzędzia konfiguracyjne i koordynujące pracę poszczególnych podmodułów, z których zbudowany jest serwer.

5.3 Modułowość i hybrydowość SWD Ekanwin

Jeśli weźmiemy pod uwagę klasyfikację systemów hybrydowych (luźno związanych) opracowaną przez Medskera [192] w kontekście struktur dwuskładnikowych: modelowania symulacyjnego i technik inteligentnych, to otrzymamy następujące kombinacje (typy struktur) modelu symulacyjnego i technik sztucznej inteligencji:

- Nadrzędny moduł techniki inteligentnej i podrzędny komputerowy model symulacyjny jednostki gospodarczej
- Podrzędny moduł techniki inteligentnej-nadrzędny komputerowy model symulacyjny jednostki gospodarczej
- Równorzędne sterowanie przez moduł techniki inteligentnej i moduł komputerowego modelu symulacyjnego.

W systemie Ekanwin mamy do czynienia z przypadkiem trzecim [Rys. 5-2].



Rys. 5-2 Struktura równorzędna modelu symulacyjnego i systemu eksperckiego

W Ekanwinie wyróżnić można następujące moduły (komponenty): dydaktyczny model symulacyjny przedsiębiorstwa przemysłowego, interfejs użytkownika, moduł planowania i sterowania eksperymentami symulacyjnymi, system ekspercki, moduł wizualizacji, moduł interfejs komunikacji między komponentami, moduł metod analitycznych. Swain [291] stwierdza: „... modele symulacyjne mogą być sterowane przez rozwiązania uzyskane dzięki oprogramowaniu optymalizującemu, a następnie uruchamiane w celu wglądu w proponowane rozwiązanie lub wsparcia ewaluacji alternatywnych rozwiązań”. W kolejnych punktach zostaną omówione najistotniejsze moduły.

5.3.1 Zastosowanie interfejsu hybrydowego

Implementacja Ekanwina w systemie MS Windows wyposażona została w **interfejs** o charakterze mieszanym (hybrydowym). Komunikacja między użytkownikiem a zespołem modułów przetwarzania wiedzy (*knowledge system*), w tym modułem modelu symulacyjnego, mechanizmem wnioskowania i planowania eksperymentu oraz sterowania modułem transformacji (translacji) opisu problemu (*problem processing system*) będącym łącznikiem z modułem wizualizacji (w tym analizy

wrażliwości) zrealizowana została w postaci menu, formularzy wprowadzania danych i wyboru opcji eksperymentu, drzewiastej struktury eksperymentu symulacyjnego i języka interpretacji komend.

5.3.2 Zadania modułu wizualizacji

Zadaniem **modułu wizualizacji** jest prezentacja wyników obliczeń wykonanych przez moduł przetwarzania wiedzy i moduł translacji opisu problemu oraz zwrotna interpretacja poleceń użytkownika, przykładowo sygnalizowanie błędnie wprowadzonych danych oraz potwierdzanie ich poprawności składniowej. Moduł wizualizacji (prezentacji) **postrzegany** jest przez użytkownika SWD w postaci tabelarycznych zestawień wyników analiz, różnego rodzaju wykresów, hierarchicznie powiązanych ze sobą sprawozdań finansowych pro forma (takich jak wykonywane w SWD ProAct [48], zdań analizy słownej, wyróżnionych pozycji (opcji, wyrazów), schematów, drzew decyzyjnych z wyróżnionymi węzłami lub ścieżkami, wyszczególnionych przetestowanych (odpalonych) reguł zawartych w bazie wiedzy, objaśnień procesu wnioskowania (etapy pośrednie i konkluzja końcowa). Jak piszą Rathman i Mannino [249] wśród współczesnych SWD obserwuje się **przesunięcie od manipulacji** (tzn. projektowania, modyfikowania, konfigurowania eksperymentów) wewnątrz algorytmu ku **bezpośredniemu manipulowaniu modelem** i jego parametrami i **kontrolowania w trybie on-line eksperymentu symulacyjnego**. Podobny postęp dotyczy **przejścia** od przetwarzania danych ku wizualnemu przetwarzaniu informacji. Hybrydowy SWD Ekanwin umożliwia **graficzne konfigurowanie eksperymentów**, modyfikację **na bieżąco** parametrów eksperymentu (zmiennych decyzyjnych) dzięki czemu otrzymujemy **animację** (z dowolną prędkością) wykresów wskaźników, **obserwujemy** powstawanie i usuwanie faktów z bazy wiedzy oraz logiczny stan wybranych do obserwacji reguł wnioskowania.

5.3.3 Aspekty projektowe modułu wizualizacji

Projektant interfejsu człowiek-komputer powinien mieć następujące cechy [135]: dysponować rozległą i **wielodyscyplinarną wiedzą**, posiadać **doświadczenie dziedzinowe** tzn. dotyczące przedmiotu badań, znać możliwości i ograniczenia technologii informacyjnej oraz osób mających posługiwać się tymi technologiami, posiadać umiejętność korzystania z osiągnięć psychologii kognitywnej, brać pod uwagę ergonomię interfejsu oraz posiadać zdolności organizacyjne. Przyjazny interfejs użytkownika (łatwy do przyswojenia i prosty w użytkowaniu) jest warunkiem koniecznym powodzenia w wykorzystaniu systemu wspomagania decyzji, tzn. używania przez menedżera w procesie podejmowaniu decyzji.

Wysoki udział **kosztów opracowania interfejsu** w ogólnych kosztach projektów SWD, chociaż obecnie spada – według Sankara i innych [258] do początku lat 90-tych XX wieku wynosiły od 60 do 70 % - wciąż wpływa na skupienie uwagi twórców SWD na **wewnętrznej architekturze**

SWD (programowanie procedur, dostęp do danych, opracowanie reguł wnioskowania). Wydaje się, że **przyczyny spadku kosztów** projektowania interfejsu są dwojakie: udostępnienie obiektowych, niezależnych od konfiguracji sprzętowej i przystosowanych do wielokrotnego wykorzystania bibliotek usług interfejsu (X-Toolkit, Macintosh Toolbox, różnorodne oprogramowanie typu RAD dla komputerów PC) oraz dołączenie warstwy pośredniej między interfejsem użytkownika a modułami przetwarzania wiedzy [210]. Dzięki warstwie pośredniej ogranicza się konieczność modyfikacji aplikacji przy zwiększaniu jej funkcjonalności. Dlatego też Turban [302] stwierdza, że właśnie **niska jakość interfejsów SWD** jest przyczyną **braku pełnego wykorzystania możliwości wewnętrznej architektury SWD** oraz implementowanych w SWD metod analiz ilościowych. Nawet silnie ustrukturalizowane, posługujące się tekstową reprezentacją języki opisu i tworzenia interfejsy SWD nie są przyjazne dla nieprogramującego użytkownika i co więcej aplikacje wykonane z ich wykorzystaniem przeznaczone są raczej dla programistów niż menedżerów i ekspertów dziedzinowych. Jak stwierdza Fischer [85] przyczynami sukcesu bądź porażki SWD nie są już szybkość reagowania czy wielkość rozwiązywanych problemów a **możliwości komunikacyjne** oraz **interfejs użytkownika**.

5.3.4 Moduł wizualizacji w SWD Ekanwin

Jednym z czynników wpływających na **podwyższenie jakości interfejsu SWD** jest według Sankara i innych [258] jego **zgodność na poziomie wewnętrznym** (systemu operacyjnego, bibliotek procedur autorstwa innych producentów, w tym reprezentacji wiedzy, modelu plików i kompatybilnych urządzeń zewnętrznych) oraz **poziomie zewnętrznym** (zbliżony styl sterowania programem, kompatybilne procedury wejściowe, zgodność użytkowa modułów interpretacji czynności użytkownika i wizualizacji wyników). Przykładami **prób zapewnienia zgodności na poziomie zewnętrznym** dla procedur wejściowych (moduł interpretacji czynności użytkownika) są **techniki definiowania wskaźników ekonomicznych** w Analizatorze Menedżera i Ekanwinie działającym w środowisku MS Windows. Na **poziomie wewnętrznym** procedury obsługi eksperymentów symulacyjnych zapewniają zgodność modelu symulacyjnego, relacyjnej bazy danych i podmodułu sprawozdań finansowych *pro forma* między implementacjami programu Ek_an_ działającymi w różnych środowiskach (tekstowym - MS DOS i graficznym - MS Windows).

Kolejnym wartym uwagi czynnikiem są **możliwości adaptacyjne**. Dotyczą one **wielowymiarowości dialogu** między użytkownikiem a modulem (systemem) przetwarzania wiedzy. Poszczególne **tryby komunikacji** w okienkowym Ekanwinie przyporządkować można **stopniom zaawansowania użytkownika**:

(1) **Interaktywne, graficzne definiowanie drzewa eksperymentów**, uruchamianie przygotowanych uprzednio procedur optymalizacyjnych (zawartych w planie eksperymentu) przez **początkującego**

użytkownika, analiza wrażliwości korespondująca z wyświetlaniem krokowo zmieniających się wartości wskaźników ekonomicznych (prezentacja trendów)

(2) **Samodzielne konstruowanie zestawów** wskaźników ekonomicznych, uruchamianie **wnioskowania eksperckiego i parametryzacja reguł wnioskowania** dla potrzeb analizy wrażliwości, sterowanie kolejnością procedur o charakterze optymalizacyjnym (etapy planu eksperymentu) przez **średniozaawansowanego** użytkownika,

(3) **Konstrukcja reguł wnioskowania** czyli **eksperckiej bazy wiedzy** oraz **samodzielne budowanie procedur planu eksperymentu** (programowanie eksperymentu, odwzorowanie prowadzenia eksperymentu przez ludzkiego operatora w celu np. optymalizacji) przez **zaawansowanego użytkownika** (nieingerującego w źródłowy kod symulatora).

Klasyfikacja rodzajów użytkowników hybrydowego SWD (na przykładzie Ekanwina) jest analogią klasyfikacji **stopni adaptacyjności interfejsu** podanej przez Lianga [171]. Interfejs *user-involved* (oparty na graficznym prowadzeniu dialogu) koresponduje z typem użytkownika początkującego, *user-controlled* (dostosowanie ilości i charakterystyki dostępnych informacji – wyników eksperymentu) koresponduje z typem użytkownika średnio zaawansowanego. Elementy *self-adaptive* (dokumentującego postępowanie badawcze eksperymentatora-menedżera) widoczne są w przypadku typu użytkownika zaawansowanego odwzorowującego **własny styl prowadzenia eksperymentów symulacyjnych i podejmowania decyzji** w programach (planach eksperymentu) adaptujących się do aktualnej sytuacji na podstawie obserwowanych wyników ekonomicznych zawartych w sprawozdaniach finansowych, dostępnych poprzez moduł translacji opisu problemu) opracowanych przez siebie w języku planowania eksperymentu.

Potwierdzają to Rathnam i Mannino [249] wyróżniając następujące **typy użytkowników SWD**: **projektantów systemu** (od modelu interfejsu do modułu przetwarzania wiedzy) posługujących się wiedzą dotyczącą konstruowania modeli matematycznych i inżynierii oprogramowania, **analityków procesów gospodarczych** przebiegających w firmie, posługujących się wiedzą dziedzinową (budujących zestawy wskaźników ekonomicznych, stosujących modele analityczne i wypełniających bazę wiedzy, przygotowujących raporty) oraz **członków kierownictwa** (różnych szczebli) dysponujących wiedzą o organizacji firmy, procesach wytwarzania i uprawnieniami do podejmowania decyzji (rozważających warianty decyzyjne). Grupy użytkowników charakteryzują się zatem odmiennymi zainteresowaniami (potrzebami informacyjnymi). Wprowadzenie do SWD Ekanwin **modułu interpretatora języka sterowania eksperymentem** jest próbą syntezy dwu sprzecznych (na pierwszy rzut oka) postulatów skutecznego realizowania postawionych przed SWD zadań oraz naśladowania sposobu rozumowania analityka (a co najmniej jego postępowania badawczego).

5.4 Cele hybrydowego systemu wspomaganie decyzji Ekanwin

Hybrydowy SWD Ekanwin oferuje wspomaganie podejmowania decyzji w dwu trybach. W tym celu wykorzystuje integrację modeli optymalizacyjnych sformalizowanych w języku sterowania eksperymentem oraz symulację komputerową. W pierwszym trybie jest przeprowadzany automatycznie sterowany eksperyment, którego plan zdefiniowany został za pomocą programu w języku LEKS. Może to być plan służący **sporządzeniu zestawienia wartości zmiennych zależnych** w funkcji zmiennych decyzyjnych, **optymalizacji określonego wyrażenia** itp. W programie optymalizacyjnym odwzorowane są **scenariusze** złożone z **elementarnych czynności** wykonywanych przez eksperymentatora. Modele optymalizacyjne najczęściej składają się ze **zbioru zmiennych decyzyjnych**, **zbioru ograniczeń** oraz **funkcji celu** (jednej lub wielu) [67]. Po zakończeniu sesji z programem optymalizacyjnym można otrzymać **raport dotyczący wybranego wskaźnika** (wyrażenia), sporządzony na podstawie reguł zawartych w bazie wiedzy oraz wykonać **analizę wrażliwości** za pomocą **graficznego interfejsu użytkownika** sprzężonego z **mechanizmem wnioskowania eksperckiego**. W drugim trybie interaktywna, graficzna i wspomagana przez mechanizm wnioskujący **analiza wrażliwości** umożliwi określenie punktu startowego do przeprowadzenia serii eksperymentów mających na celu dalszą optymalizację wyników (np. wybranego wskaźnika czy agregatu wskaźników).

W hybrydowym **systemie wspomaganie decyzji Ekanwin** zastosowano **szereg technik** o odmiennym charakterze, tworzących różnorodne struktury o zróżnicowanej złożoności. Celem badań było zbudowanie analitycznego inteligentnego systemu hybrydowego, zrealizowanego w **technice modułarnej**, łączącego system wspomaganie decyzji, regułowy **system ekspercki**, możliwości **wizualizacyjne**, **symulację** komputerową, **planowanie eksperymentu**, **proceduralny język skryptowy** (programowania), **optymalizację**, otwartość na **rozszerzenie parametrów funkcjonalnych**. Jest to przykład rozszerzonej inteligentnej hybrydy, postulowanej przez Rutkowskiego [254] - narzędzia wspomagającego podejmowanie decyzji, **synergicznego** i pokonującego ograniczenia obserwowane w pojedynczo implementowanych technikach i narzędziach analitycznych. Modułem występującym we **wszystkich kombinacjach** (układach) w systemie EKANWIN jest **symulacja komputerowa**. Warto jednak zauważyć, że ze względu na modelowany w module symulacyjnym **system gospodarczy** (przedsiębiorstwo przemysłowe) także informatyczna aplikacja **symulacji** posiada cechy **hybrydowości**. Wynikają one z **ciągło-dyskretnego** charakteru procesów modelowanych w module symulacyjnym Ekanwina. Pepyne i Cassandras [234] definiują bowiem **hybrydę** jako „system łączący ciągłą i dyskretną dynamikę zachowań” oraz „system łączący podejście zdarzeniowe i ciągło-dynamiczne”.

6 Projektowanie interfejsu systemu Ekanwin dla potrzeb wizualizacji i wnioskowania

6.1 Korzystanie z graficznego interfejsu użytkownika

6.1.1 Moduł wizualizacji i moduł wnioskowania eksperckiego - opis korzystania z interfejsu

Interfejs użytkownika wykorzystuje się w statycznej (polegającej na rozważaniu pojedynczego wariantu w funkcji czasu) analizie ekonomicznej przedsiębiorstwa, którego model jest sercem programu Ekanwin. Jedną z podstawowych zalet symulacji jest jednak wykonywanie **analiz wariantowych** typu *ceteris paribus*, czyli badanie co się stanie jeśli dokonamy zmiany wartości jednej ze zmiennych niezależnych (np. warunków działalności czy zmiennych decyzyjnych). Co więcej, ze względu na badanie zachowania modelu a nie rzeczywistego systemu można wykorzystywać **metodę prób i błędów**, prowadzić analizy wrażliwości wyników przedsiębiorstwa na zmiany wartości zmiennych decyzyjnych, zmiennych stanu czy nawet innych zmiennych wyjściowych. Dlatego też rozbudowano interfejs użytkownika systemu Ekanwin o **możliwość sterowania poszczególnymi komórkami formularza** (macierzy) zmiennych decyzyjnych za pomocą jednego z szeroko wykorzystywanych elementów ekranowych środowiska GUI jakim są paski przewijania (suwaki). Suwaki są naturalnie przystosowane do sterowania analitycznymi procesami decyzyjno-predykcyjnymi takimi jak analiza wrażliwości, symulacja porównawcza czy konkurencyjna.

Istotą realizacji eksperymentów **what-if** poprzez wskaźniki jest powiązanie ich **reakcji na bodźce zewnętrzne**, takie jak przesuwanie, klikanie, ciągnięcie z **uruchamianiem procesu symulacji** oraz **wyświetlaniem wykresu zmienionej trajektorii wartości i aktualizacją wyników wnioskowania eksperckiego** [Rys. 6-4], [Rys. 6-5]. Każda zmiana parametru reprezentowanego przez **pozycję suwaka** (wskazówki potencjometru) powodować będzie zmianę wartości odpowiedniej zmiennej decyzyjnej, wykonanie obliczeń i uzupełnienie drzewa eksperymentów decyzyjnych o dodatkowy węzeł. Jednocześnie stosuje się zwykle kasowanie węzła decyzyjnego reprezentującego wyjściowy zestaw zmiennych decyzyjnych aby ograniczyć nieuzasadniony rozrost drzewa decyzyjnego (w trakcie wielokrotnych eksperymentów). Za pomocą suwaków można dokonywać praktycznie **dowolnych** (w założonym przedziale wartości i obranym skoku) **zmian wartości zmiennej decyzyjnej** o charakterze ciągłym (np. podział zysku, czy – z oczywistymi zastrzeżeniami – cenami wyrobów wyrażonych w jednostkach pieniężnych). Z drugiej strony istnieją **zmienne decyzyjne dyskretne**, których wartości dobierać można jedynie z ograniczonego zestawu (np. identyfikatorów projektów inwestycyjnych określanych przez ich parametry takie jak źródło finansowania, koszty, oczekiwana wydajność itp.).

Złożoność podejmowanych decyzji, choćby ze względu na znaczną liczbę sterowalnych parametrów macierzy zmiennych decyzyjnych, implikuje **trzy podstawowe konstrukcje pulpitu sterującego** skonstruowanego za pomocą suwaków:

1. Każdej pozycji macierzy zmiennych decyzyjnych (w tym poszczególnym okresie) odpowiada odrębny suwak
2. Przebiegiem eksperymentu sterujemy za pomocą tandemu **suwak** (ustalanie wartości zmiennej) + **lista rozwijana** (wybór zmiennej i okresu którego dotyczą zmiany) [Rys. 6-3]
3. Interfejs wspomagający decydenta w określonej klasie problemów, będący rozwiązaniem pośrednim między (1) i (3) [Rys. 6-3].

Można także łączyć (2) i (3). Wadą rozwiązania (1) jest pojawienie się problemów dotyczących projektowania i realizacji przełączania między oknami poszczególnych wykresów i raportów z procesu wnioskowania systemu eksperckiego oraz sterowania parametrami eksperymentu, natomiast w przypadku (2) brak informacji o wartościach pozostałych komórek macierzy zmiennych decyzyjnych (eksperymenty typu *ceteris paribus*).

Funkcje interfejsu użytkownika związane z prowadzeniem **analizy porównawczej** wynikają z **reprezentacji scenariuszy eksperymentu** w postaci drzewa decyzyjnego. Każdy węzeł w drzewie decyzyjnym reprezentuje zarówno podejmowane decyzje, tożsame z ukonkretnieniem macierzy zmiennych decyzyjnych, jak i otrzymane rezultaty (dostępne na żądanie sprawozdania finansowe) oraz relacje z innymi węzłami (typu rodzic-potomek) oraz przynależności do danego pokolenia (odwzorowanego okresu).

W omawianej platformie informatycznej (Ekanwin) zastosowano następujące rozwiązania:

1. Sterowanie przemieszczaniem się (tzn. zmianą aktywnego węzła) wewnątrz struktury drzewiastej za pomocą myszy lub klawiszy kursora. W tym trybie ustalana jest topologia drzewa oraz uruchamiane obliczenia (symulacje)
2. Przegląd wyników poszczególnych wariantów eksperymentu w celu wykonania analiz porównawczych. Dokonywany jest on przez wybór wirtualnych przycisków ekranowych w celu przemieszczenie kursora w tym samym poziomie drzewa, tzn. między „rodzeństwem” lub dalszymi krewnymi w tym samym pokoleniu (modelowanym okresie)
3. Wariant symulacji porównawczej (2) polegający na wyświetlaniu animacji (samoczynnie zmieniającego się wykresu) czy wartości badanej wielkości (względnie jej dynamiki) poprzez uruchomienie krokowej zmiany wartości badanego parametru w rozważanym zakresie. W tym celu należy ustalić granice badanego przedziału wartości, wartość kroku, kierunek zmian oraz zapewnić możliwość przerwania animacji (na żądanie).

Heurystyczne poszukiwanie rozwiązań optymalnych zdefiniowanego problemu analityczno-optymalizacyjnego wymaga udostępnienia decydentowi możliwości oddziaływania na następujące zmienne decyzyjne:

- cenę poszczególnych wyrobów w kolejnych latach sekwencji decyzyjnej
- zamówienia produkcyjne (jw.)
- pozostałe, wyszczególnione w formularzu zmiennych decyzyjnych.

W analizie wrażliwości przydatne może być skorzystanie z animacji wyników zmian badanej zmiennej decyzyjnej.

Biorąc pod uwagę założenie o różniczkowalności funkcji celu, specyfikę sterowania eksperymentem za pomocą suwaków oraz **możliwości animacyjne modułu graficznego** można wyszczególnić następujące procedury poszukiwania rozwiązań optymalnych:

1. Manipulowanie wybranym suwakiem i obserwacja aktualnego przebiegu badanego wskaźnika (funkcji celu, kryterium optymalizacyjnego) oraz zapoznanie się z równoległe dostarczanymi diagnozami tekstowymi; wyróżnić można cztery techniki:

a) Przygotowanie materiału analitycznego poprzez **sporządzenie zestawień wartości badanej funkcji celu**, zlokalizowanie granic przedziału w którym podejrzewamy występowanie ekstremum lokalnego (respektującego ograniczenia) a następnie zmniejszenie wartości kroku i powtarzanie powyższej procedury do momentu znalezienia zadowalającego rozwiązania. Pomocne tu jest monitorowanie zmian badanej wielkości i zapis raportu do pliku tekstowego. Można także przygotować procedurę w języku sterowania eksperymentem **automatycznie rejestrującą** wartości badanej zmiennej oraz funkcji celu a następnie obserwować wykresy funkcji celu podczas aktywacji sąsiadujących („rodzeństwa i kuzynów”) węzłów z tego samego pokolenia (okresu).

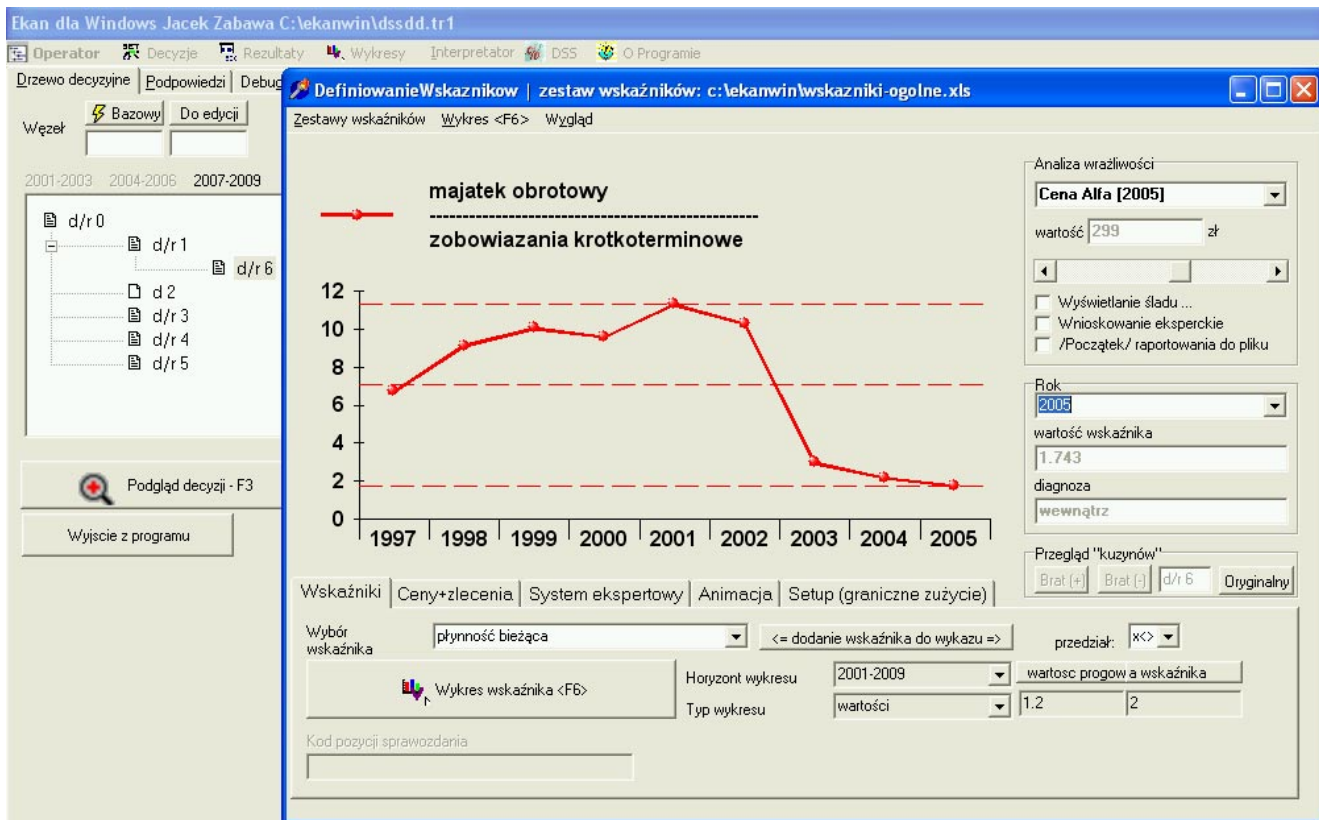
b) Zgrubne wyznaczenie granic przedziału do zbadania a następnie dynamiczne manipulacje pozycją suwaka co powoduje powstanie animacji na ekranie

c) Uruchomienie animacji w badanym zakresie przy przyjętym kroku zmian i próba uchwycenia wartości parametru dla którego występuje ekstremum funkcji kryterium [Rys. 6-2]

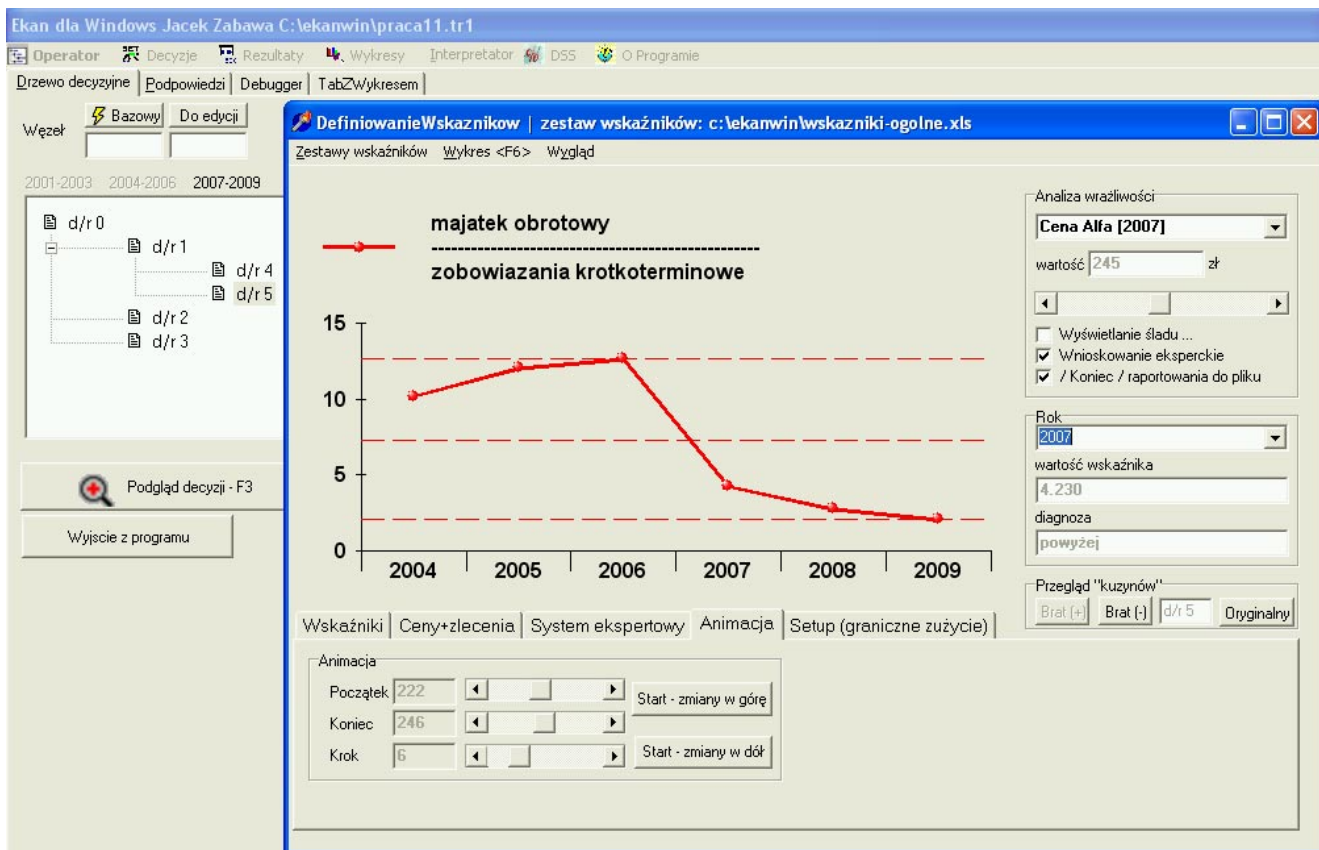
d) Uruchomienie animacji w możliwie szerokim zakresie a następnie analiza raportu w celu ustalenia prawdopodobnych granic przedziału w którym znajduje się ekstremum.

2. Wywoływanie powyżej opisanej procedury zgodnie z ustaloną strategią poszukiwań.

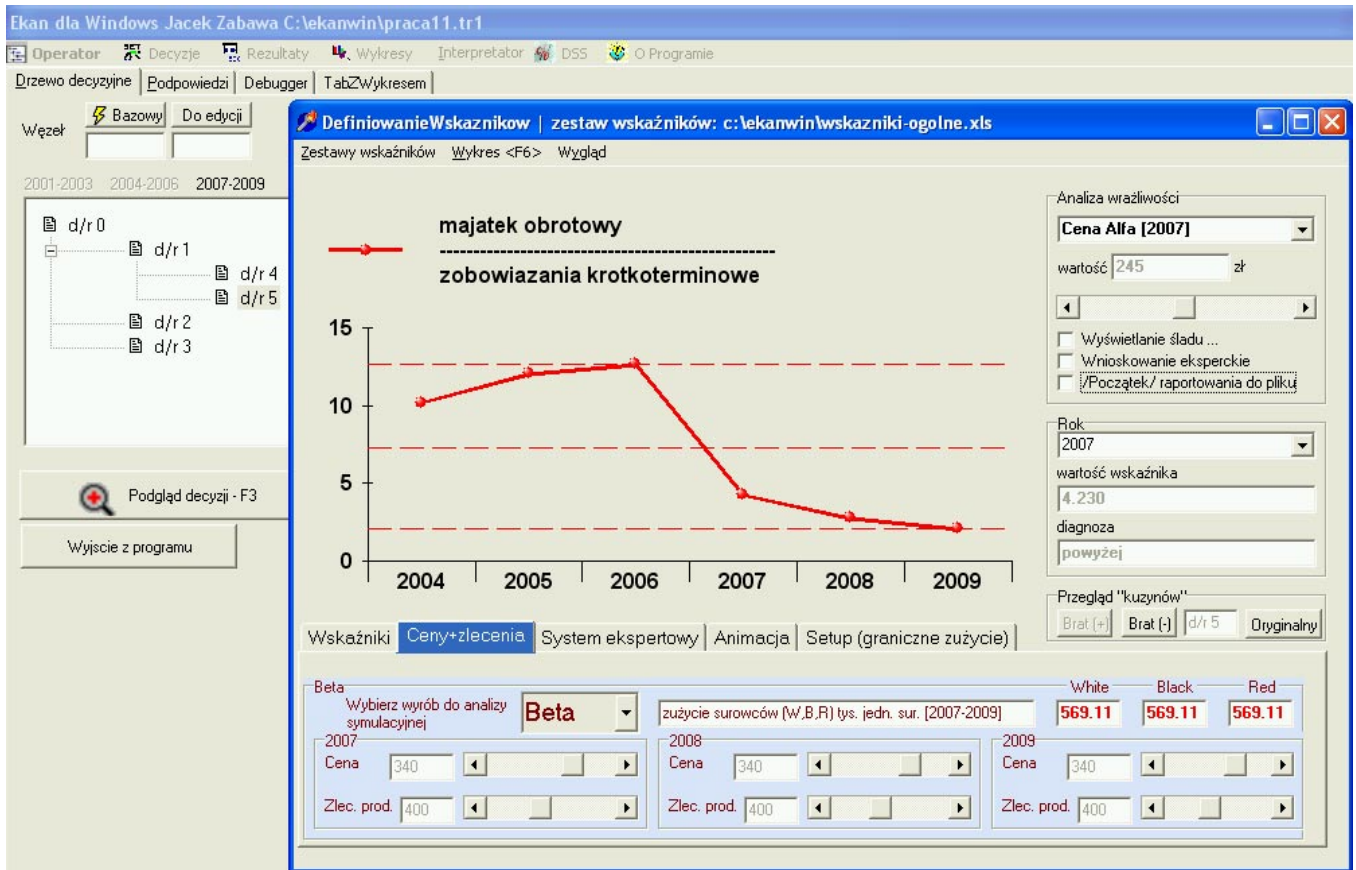
Strategię poszukiwań można opisać w następujący sposób: najpierw optymalizujemy zmienne dotyczące najbliższej przyszłości a potem dalszej (podejście Bellmanowskie, właściwe dla programowania dynamicznego). Można także optymalizować wartość kryterium kolejno dla poszczególnych wyrobów czy porównywać efekty optymalizacji dla poszczególnych projektów inwestycyjnych.



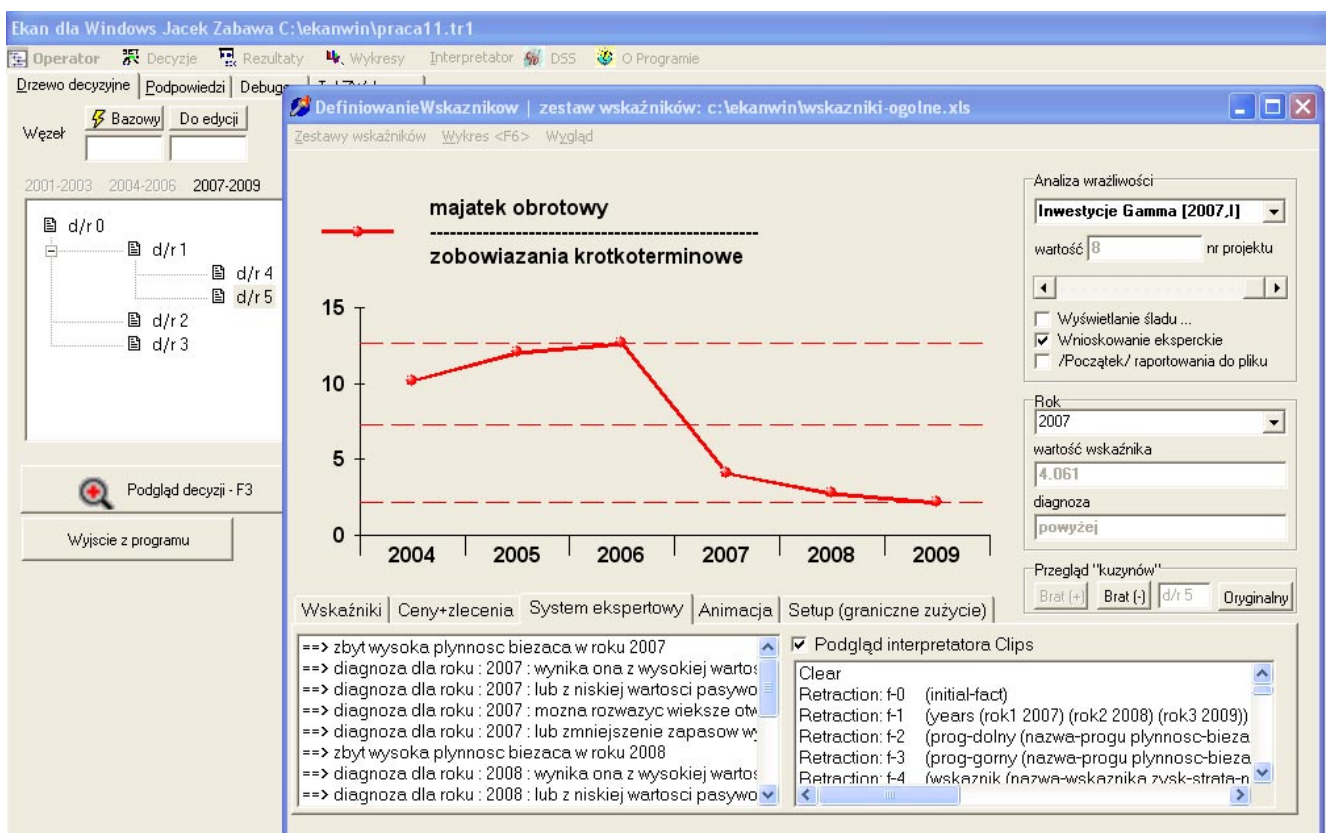
Rys. 6-1 Moduł wizualizacji. Tryb analizy wskaźnikowej - aktywne narzędzie wyświetlania wykresów wartości wskaźnika w pełnym zakresie lat, analiza wrażliwości, proste wnioskowanie dotyczące aktywnego wskaźnika na podstawie porównania z zakresem zadowalającym



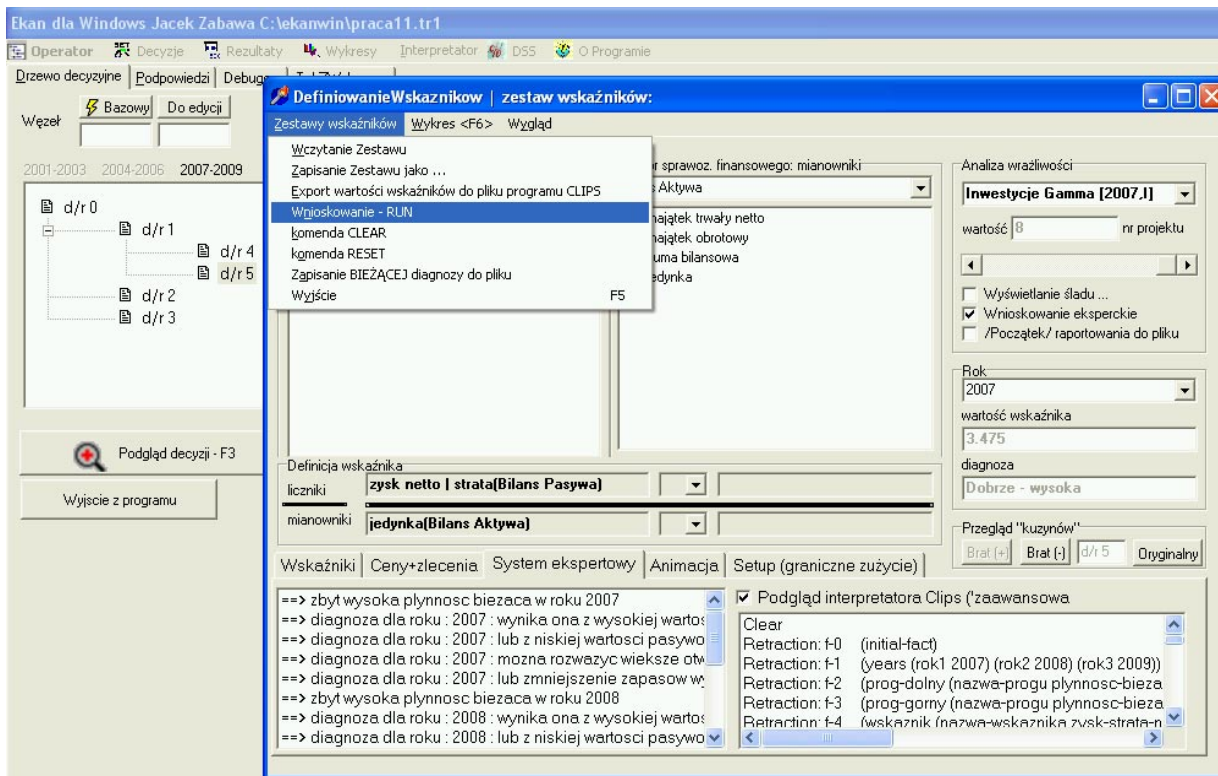
Rys. 6-2 Tryb analizy wskaźnikowej, aktywne narzędzie wyświetlania wykresów wartości wskaźnika w wybranym zakresie lat, analiza wrażliwości, proste wnioskowanie dotyczące aktywnego wskaźnika z podstawie porównania z zakresem zadowalającym, opcja animacji



Rys. 6-3 Moduł wizualizacji. Tryb analizy wskaźnikowej, pulpit sterujący z wieloma suwakami – umożliwiającymi badanie płynności w zależności od równoczesnych zmian sześciu zmiennych decyzyjnych oraz zużycia surowców



Rys. 6-4 Moduł wizualizacji. Tryb analizy wskaźnikowej, aktywne okno diagnoz otrzymanych z systemu eksperckiego, wyświetlany podgląd agendy interpretatora systemu szkieletowego CLIPS



Rys. 6-5 Moduł wizualizacji. Tryb analizy wskaźnikowej, przygotowanie do włączenia wnioskowania eksperckiego i konstrukcji wskaźnika ekonomicznego, aktywne okno diagnoz otrzymanych z systemu eksperckiego, wyświetlany podgląd agendy interpretatora systemu CLIPS

6.1.2 Funkcja raportowania systemu Ekanwin

Jedną z funkcji SWD Ekanwin jest możliwość **rejestracji wszelkich zmian** dokonywanych za pomocą graficznych elementów takich jak suwaki (paski przewijania) czy pola wyboru. Rejestrację zintegrowano w pełni z modułem eksperckim w celu tworzenia raportów tekstowych. W raporcie zawarte są także **informacje** o aktywowaniu konkretnych baz wiedzy (źródeł wiedzy) z których korzysta się podczas procesu wnioskowania. Rejestracja obejmuje także aktualnie interpretowane wskaźniki. Dzięki temu możliwe jest uzyskiwanie tekstowych raportów (w automatycznie tworzonych unikatowych plikach) dotyczących nie tylko statycznej analizy sytuacji symulowanego przedsiębiorstwa, ale także dokumentujących wszystkie zmiany w wynikach analiz będących efektem eksperymentów z wartościami zmiennych decyzyjnych. Dokumentowanie wszelkich postępowań analitycznych, decyzyjnych za pośrednictwem zapisów o dokonanych eksperymentach umożliwia także zapoznanie się ze stylem przygotowania podejmowanych decyzji prezentowanym przez konkretnego eksperymentatora a następnie jego kodyfikację. Być może pozwoli to także na przeprowadzenie badań nad przydatnością poszczególnych źródeł wiedzy (zestawów wskaźników ekonomicznych) w zależności od napotykaných problemów decyzyjnych. Poniżej przedstawiono przykład raportu.

Operator :C:\ekanwin\praca10.sir

Decyzja : 6

```
#####
#
#           Z M I E N N E   D E C Y Z Y J N E
#####
#           #           #           #           Dla roku:           #
#   Zmienna   #   Przedmiot   #   Jednostka   #           #           #           #           #
#   decyzyjna #   decyzji     #   miary     #           #####
#           #           #           #           #   2004 #   2005 #   2006 #
#####
#           #           #           #           #           #           #           #           #
#1. Zlecenia na produkcje #   Wyrób ALFA   #           #   400.0 #   400.0 #   400.0 #
#   wyrobów           #           BETA   #   1000j.wyr./rok #   400.0 #   400.0 #   400.0 #
#           #           GAMMA   #           #   400.0 #   400.0 #   400.0 #
#           #           #           #           #           #           #           #
#2. Ceny wyrobów gotowych #   Wyrób ALFA   #           #   372.0 #   250.0 #   250.0 #
#           #           BETA   #   zl/j.wyr. #   340.0 #   340.0 #   340.0 #
#           #           GAMMA   #           #   411.0 #   411.0 #   411.0 #
#           #           #           #           #           #           #           #
#3. Harmonogram zakupu   #   Surowiec WHITE #           #   400.0 #   350.0 #   350.0 #
#   surowców           #           BLACK #   1000j.sur./rok #   400.0 #   350.0 #   350.0 #
#           #           RED     #           #   400.0 #   350.0 #   350.0 #
#           #           #           #           #           #           #           #
#4. Udział dywidendy zwykł.#   Akcje zwykłe #   % zysku po opodatk. #   10.0 #   10.0 #   10.0 #
#           #           #           #           #           #           #           #
#5. Wskaźnik zadłużenia   #   Kredyt hipot. #   % wart. aktywów     #   50.0 #   50.0 #   50.0 #
#           #           #           #           #           #           #           #
#6. Emisja akcji         #   Akcje zwykłe #   mln zł             #   0.0 #   0.0 #   1.0 #
#           #           #           #           #           #           #           #
#7. Numery uruchamianych #   Linia wyrobu ALFA #           #   . . #   . . #   . . #
#   projektów inwest.   #           BETA   #           #   . . #   . . #   . . #
#           #           GAMMA #           #   . . #   . . #   . . #
#           #           #           #           #           #           #           #
#####
```

Korzystam z bazy wskaźników: C:\ekanwin\wskazniki-ogolne.xls

korzystam z bazy reguł: C:\ekanwin\reguły-zysk-strata netto.clp

==> niewystarczający zysk w roku 2004

==> ujemny zysk w roku 2004

==> dodatnia strata w roku 2004

==> wysoki zysk w roku 2005

==> wysoki zysk w roku 2006

==> wzrost zysku w latach 2004-2006

Zmieniono: Cena Alfa [2004] na: 382

==> niewystarczający zysk w roku 2004

==> ujemny zysk w roku 2004

==> dodatnia strata w roku 2004

==> wysoki zysk w roku 2005

- ==> wysoki zysk w roku 2006
- ==> wzrost zysku w latach 2004-2006

Zmieniono: Cena Alfa [2004] na: 359

- ==> niewystarczający zysk w roku 2004
- ==> ujemny zysk w roku 2004
- ==> dodatnia strata w roku 2004
- ==> wysoki zysk w roku 2005
- ==> wysoki zysk w roku 2006
- ==> wzrost zysku w latach 2004-2006

zmieniono bazę reguł na: C:\ekanwin\reguły-płynność bieżąca.clp

Zmieniono: Cena Alfa [2004] na: 369

- ==> zbyt wysoka płynność bieżąca w roku 2004
- ==> diagnoza dla roku : 2004 : wynika ona z wysokiej wartości aktywów obrotowych
- ==> diagnoza dla roku : 2004 : lub z niskiej wartości pasywów bieżących
- ==> diagnoza dla roku : 2004 : można rozważyć większe otwarcie na kredytowanie
- ==> diagnoza dla roku : 2004 : lub zmniejszenie zapasów wyrobów gotowych poprzez
obniżenie cen
- ==> zbyt wysoka płynność bieżąca w roku 2005
- ==> diagnoza dla roku : 2005 : wynika ona z wysokiej wartości aktywów obrotowych
- ==> diagnoza dla roku : 2005 : lub z niskiej wartości pasywów bieżących
- ==> diagnoza dla roku : 2005 : można rozważyć większe otwarcie na kredytowanie
- ==> diagnoza dla roku : 2005 : lub zmniejszenie zapasów wyrobów gotowych poprzez
obniżenie cen
- ==> zbyt wysoka płynność bieżąca w roku 2006
- ==> diagnoza dla roku : 2006 : wynika ona z wysokiej wartości aktywów obrotowych
- ==> diagnoza dla roku : 2006 : lub z niskiej wartości pasywów bieżących
- ==> diagnoza dla roku : 2006 : można rozważyć większe otwarcie na kredytowanie
- ==> diagnoza dla roku : 2006 : lub zmniejszenie zapasów wyrobów gotowych poprzez
obniżenie cen
- ==> zbyt wysoka płynność bieżąca we wszystkich latach 2004-2006

==> przelamanie trendu spadkowego płynności bieżącej w latach 2004-2006

Zmieniono: Cena Alfa [2004] na: 379

==> zbyt wysoka płynność bieżąca w roku 2004

==> diagnoza dla roku : 2004 : wynika ona z wysokiej wartości aktywów obrotowych

==> diagnoza dla roku : 2004 : lub z niskiej wartości pasywów bieżących

==> diagnoza dla roku : 2004 : można rozważyć większe otwarcie na kredytowanie

==> diagnoza dla roku : 2004 : lub zmniejszenie zapasów wyrobów gotowych poprzez
obniżenie cen

==> zbyt wysoka płynność bieżąca w roku 2005

==> diagnoza dla roku : 2005 : wynika ona z wysokiej wartości aktywów obrotowych

==> diagnoza dla roku : 2005 : lub z niskiej wartości pasywów bieżących

==> diagnoza dla roku : 2005 : można rozważyć większe otwarcie na kredytowanie

==> diagnoza dla roku : 2005 : lub zmniejszenie zapasów wyrobów gotowych poprzez
obniżenie cen

==> zbyt wysoka płynność bieżąca w roku 2006

==> diagnoza dla roku : 2006 : wynika ona z wysokiej wartości aktywów obrotowych

==> diagnoza dla roku : 2006 : lub z niskiej wartości pasywów bieżących

==> diagnoza dla roku : 2006 : można rozważyć większe otwarcie na kredytowanie

==> diagnoza dla roku : 2006 : lub zmniejszenie zapasów wyrobów gotowych poprzez
obniżenie cen

==> zbyt wysoka płynność bieżąca we wszystkich latach 2004-2006

==> przelamanie trendu spadkowego płynności bieżącej w latach 2004-2006

zmieniono bazę reguł na: C:\ekanwin\reguły-wskazniki-płynność-płynność bieżąca.clp

Zmieniono bazę wskaźników na: C:\ekanwin\wskazniki-płynność.xls

zmieniono bazę reguł na: C:\ekanwin\reguły-wskazniki-płynność-płynność bieżąca.clp

Zmieniono: Cena Alfa [2004] na: 389

==> zbyt wysoka płynność bieżąca w roku 2004

==> diagnoza dla roku : 2004 : wynika ona z wysokiej wartości aktywów obrotowych

- ==> diagnoza dla roku : 2004 : lub z niskiej wartosci pasywow biezacych
- ==> diagnoza dla roku : 2004 : mozna rozwazyc wieksze otwarcie na kredytowanie
- ==> diagnoza dla roku : 2004 : lub zmniejszenie zapasow wyrobow gotowych poprzez
obnizenie cen
- ==> zbyt wysoka plynnosc biezaca w roku 2005
- ==> diagnoza dla roku : 2005 : wynika ona z wysokiej wartosci aktywow obrotowych
- ==> diagnoza dla roku : 2005 : lub z niskiej wartosci pasywow biezacych
- ==> diagnoza dla roku : 2005 : mozna rozwazyc wieksze otwarcie na kredytowanie
- ==> diagnoza dla roku : 2005 : lub zmniejszenie zapasow wyrobow gotowych poprzez
obnizenie cen
- ==> zbyt wysoka plynnosc biezaca w roku 2006
- ==> diagnoza dla roku : 2006 : wynika ona z wysokiej wartosci aktywow obrotowych
- ==> diagnoza dla roku : 2006 : lub z niskiej wartosci pasywow biezacych
- ==> diagnoza dla roku : 2006 : mozna rozwazyc wieksze otwarcie na kredytowanie
- ==> diagnoza dla roku : 2006 : lub zmniejszenie zapasow wyrobow gotowych poprzez
obnizenie cen
- ==> zbyt wysoka plynnosc biezaca we wszystkich latach 2004-2006
- ==> przelamanie trendu spadkowego plynnosci biezacej w latach 2004-2006

6.2 Koncepcja wnioskowania i bazy wiedzy

Konstrukcja bazy wiedzy w systemie Ekanwin jest procesem wieloetapowym. Schemat działania modułu wnioskowania eksperckiego przedstawiono na rysunku [Rys. 6-6], natomiast przykładową diagnozę dotyczącą wybranego wskaźnika – na rysunku [Rys. 6-7]. Poniżej zamieszczono opis poszczególnych etapów.

1. **Wczytywany jest plik z definicjami wskaźników** (utworzony w Excelu); każdy z zestawów wskaźników stanowi odrębną całość i przechowywany jest w odrębnym arkuszu kalkulacyjnym (pliku MS Excel)
2. Tworzony jest **roboczy plik tekstowy** o takiej samej nazwie jak definicje wskaźników, ale z rozszerzeniem clp
3. W pliku roboczym zapisywana jest **informacja o trzech latach**, których dotyczy dany węzeł; odbywa się to dzięki odpowiedniemu kodowi w systemie Ekanwin
4. **Reguły wnioskowania parametryzuje się** za pomocą formatki ekranowej (formularza); w pliku roboczym zapisywana jest informacja o granicznym progu wartości wskaźnika; próg ten używany jest potem do orzeczenia o wartości wskaźnika. **Wartość progu granicznego** pochodzić może z odpowiedniego pola w formularzu. Wygląda to następująco.

(deffacts progi

"progi wskaźników ekonomicznych"

(prog (nazwa-progu rentownosc-sprzedazy) (wartosc 0.015)))

5. Kolejno **zapisywane są fakty** opisujące poszczególne wskaźniki zdefiniowane w pliku Excela (wraz z wartościami wskaźników):

(deffacts wskaźniki "wskaźniki ekonomiczne");

powstaje wtedy kompletny plik faktów z następującą przykładową zawartością:

(deffacts trojka-lat "numery lat"

(years (rok1 2000) (rok2 2001) (rok3 2002)))

(deffacts progi

"progi wskaźników ekonomicznych"

(prog (nazwa-progu rentownosc-sprzedazy) (wartosc 0.015)))

(deffacts wskaźniki "wskaźniki ekonomiczne"

(wskaźnik (nazwa-wskaźnika suma-bilansowa) (rok1 2000)(wartosc1 219.809) (rok2 2001)(wartosc2 233.132) (rok3 2002)(wartosc3 245.385))

(wskaźnik (nazwa-wskaźnika suma-bilansowa1) (rok1 2000)(wartosc1 219.809) (rok2 2001)(wartosc2 233.132) (rok3 2002)(wartosc3 245.385))

(wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok1 2000)(wartosc1 0.000) (rok2 2001)(wartosc2 0.000) (rok3 2002)(wartosc3 0.000))

(wskaznik (nazwa-wskaznika zysk-netto) (rok1 2000)(wartosc1 0.000) (rok2 2001)(wartosc2 0.000) (rok3 2002)(wartosc3 0.095))

(wskaznik (nazwa-wskaznika przychody-ze-sprzedazy) (rok1 2000)(wartosc1 246.418) (rok2 2001)(wartosc2 273.197) (rok3 2002)(wartosc3 254.675))

(wskaznik (nazwa-wskaznika koszty-operacyjne) (rok1 2000)(wartosc1 270.922) (rok2 2001)(wartosc2 245.116) (rok3 2002)(wartosc3 221.768))

(wskaznik (nazwa-wskaznika zobowiazania-krotkoterminowe) (rok1 2000)(wartosc1 23.332) (rok2 2001)(wartosc2 36.823) (rok3 2002)(wartosc3 48.990)))

6. Do interpretera CLIPSA ładowany jest plik z definicjami reguł, odpowiednio do wybranego wskaźnika, np.: 'reguły-+ComboBox1.Text+'.clp').

W przypadku wskaźnika rentowności sprzedaży plik definicji reguł będzie wyglądał następująco:

```
(deftemplate prog
```

```
"definicje progów wskaźnika ekonomicznego"
```

```
(slot nazwa-progu)
```

```
(slot wartosc (type NUMBER)))
```

```
(deftemplate years
```

```
"zdefiniowanie zakresu lat"
```

```
(slot rok1)
```

```
(slot rok2)
```

```
(slot rok3))
```

```
(deftemplate wskaznik
```

```
"definicje wskaźnika ekonomicznego"
```

```
(slot nazwa-wskaznika)
```

```
(slot rok1)
```

```
(slot wartosc1 (type NUMBER))
```

```
(slot rok2)
```

```
(slot wartosc2 (type NUMBER))
```

```
(slot rok3)
```

```
(slot wartosc3 (type NUMBER)))
```

```
(defrule dobra-rentownosc-w-roku-1
```

```
?wskaznik <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok1 ?rok1) (wartosc1 ?war))
```

```
?prog <- (prog (nazwa-progu rentownosc-sprzedazy) (wartosc ?progowa))
```

```
(test (>= ?war ?progowa))
```

```
=>
```

```
(assert (dobra-rentownosc-sprzedazy-w-roku ?rok1)))
```

```
(defrule niewystarczajaca-rentownosc-w-roku-1
```

```
?wskaznik <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok1 ?rok1) (wartosc1 ?war))
```

```
?prog <- (prog (nazwa-progu rentownosc-sprzedazy) (wartosc ?progowa))
```

```
(test (< ?war ?progowa))
```

```
=>
```

```
(assert (niewystarczajaca-rentownosc-sprzedazy-w-roku ?rok1)))
```



```
(defrule dobra-rentownosc-w-roku-2
  ?wskaznik <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok2 ?rok2) (wartosc2 ?war))
  ?prog <- (prog (nazwa-progu rentownosc-sprzedazy) (wartosc ?progowa))
  (test (>= ?war ?progowa))
=>
  (assert (dobra-rentownosc-sprzedazy-w-roku ?rok2)))
```

```
(defrule niewystarczajaca-rentownosc-w-roku-2
  ?wskaznik <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok2 ?rok2) (wartosc2 ?war))
  ?prog <- (prog (nazwa-progu rentownosc-sprzedazy) (wartosc ?progowa))
  (test (< ?war ?progowa))
=>
  (assert (niewystarczajaca-rentownosc-sprzedazy-w-roku ?rok2)))
```

```
(defrule dobra-rentownosc-w-roku-3
  ?wskaznik <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok3 ?rok3) (wartosc3 ?war))
  ?prog <- (prog (nazwa-progu rentownosc-sprzedazy) (wartosc ?progowa))
  (test (>= ?war ?progowa))
=>
  (assert (dobra-rentownosc-sprzedazy-w-roku ?rok3)))
```

```
(defrule niewystarczajaca-rentownosc-w-roku-3
  ?wskaznik <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok3 ?rok3) (wartosc3 ?war))
  ?prog <- (prog (nazwa-progu rentownosc-sprzedazy) (wartosc ?progowa))
  (test (< ?war ?progowa))
=>
  (assert (niewystarczajaca-rentownosc-sprzedazy-w-roku ?rok3)))
```

```
(defrule wzrost-rentownosci-sprzedazy
  ?tymczasowy1 <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok1 ?rok1) (wartosc1 ?war1))
  ?tymczasowy2 <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok2 ?rok2) (wartosc2 ?war2))
  ?tymczasowy3 <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok3 ?rok3) (wartosc3 ?war3))
  (test (> ?war2 ?war1))
  (test (> ?war3 ?war2))
=>
  (assert (wzrost-rentownosci-sprzedazy w-latach ?rok1 ?rok3)))
```

```
(defrule spadek-rentownosci-sprzedazy
  ?tymczasowy1 <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok1 ?rok1) (wartosc1 ?war1))
  ?tymczasowy2 <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok2 ?rok2) (wartosc2 ?war2))
  ?tymczasowy3 <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok3 ?rok3) (wartosc3 ?war3))
  (test (< ?war2 ?war1))
  (test (< ?war3 ?war2))
=>
  (assert (spadek-rentownosci-sprzedazy w-latach ?rok1 ?rok3)))
```

```
(defrule dolek-rentownosci-sprzedazy
  ?tymczasowy1 <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok1 ?rok1) (wartosc1 ?war1))
  ?tymczasowy2 <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok2 ?rok2) (wartosc2 ?war2))
  ?tymczasowy3 <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok3 ?rok3) (wartosc3 ?war3))
  (test (< ?war2 ?war1))
  (test (> ?war3 ?war2))
```

```

=>
(assert (przelamanie-trendu-spadkowego-rentownosci-sprzedazy w-latach ?rok1 ?rok3)))

(defrule pagorek-rentownosci-sprzedazy
  ?tymczasowy1 <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok1 ?rok1) (wartosc1 ?war1))
  ?tymczasowy2 <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok2 ?rok2) (wartosc2 ?war2))
  ?tymczasowy3 <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok3 ?rok3) (wartosc3 ?war3))
  (test (> ?war2 ?war1))
  (test (< ?war3 ?war2))
=>
(assert (utrata-trendu-rosnacego-rentownosci-sprzedazy w-latach ?rok1 ?rok3)))

(defrule dobra-rentownosc-sprzedazy-przez-caly-okres
  ?tymczasowy1 <- (years (rok1 ?rok1) (rok2 ?rok2) (rok3 ?rok3))
  ?tymczasowy2 <- (dobra-rentownosc-sprzedazy-w-roku ?rok1)
  ?tymczasowy3 <- (dobra-rentownosc-sprzedazy-w-roku ?rok2)
  ?tymczasowy4 <- (dobra-rentownosc-sprzedazy-w-roku ?rok3)
=>
(assert (dobra-rentownosc-sprzedazy-we-wszystkich-latach ?rok1 ?rok3)))

(defrule niska-rentownosc-sprzedazy-przez-caly-okres
  ?tymczasowy1 <- (years (rok1 ?rok1) (rok2 ?rok2) (rok3 ?rok3))
  ?tymczasowy2 <- (niewystarczajaca-rentownosc-sprzedazy-w-roku ?rok1)
  ?tymczasowy3 <- (niewystarczajaca-rentownosc-sprzedazy-w-roku ?rok2)
  ?tymczasowy4 <- (niewystarczajaca-rentownosc-sprzedazy-w-roku ?rok3)
=>
(assert (niska-rentownosc-sprzedazy-we-wszystkich-latach ?rok1 ?rok3)))

```

7. Do **obszaru roboczego** interpretera CLIPS wczytywane są kolejno: **plik z definicjami reguł** (przygotowany ręcznie, niewymagający wprowadzenia wartości liczbowych) oraz **plik z definicjami faktów** (utworzony w opisany powyżej sposób bezpośrednio przez Ekanwin, zawierający wartości liczbowe). Baza faktów i baza reguł połączone **zostają w jedną całość**

8. Po wykonaniu wnioskovania, w tablicy Clips1.Facts[j] znajdują się wszystkie fakty (występujące przed wnioskowaniem oraz po wnioskowaniu). Aby **przedstawić użytkownikowi w czytelnej formie fakty** (wyniki wnioskovania) dotyczące interesującego wskaźnika, konieczna jest **automatyczna interpretacja** (filtrowanie) faktów dostępnych w wewnętrznym, niezbyt czytelnym formacie CLIPSA (fakty w tym formacie można wyświetlić w prawym okienku „Podgląd interpretera Clips”). Podczas **przeglądania wszystkich faktów** wyszukiwane są fakty dotyczące rozważanego wskaźnika i wyprowadzane **w czytelnym formacie**.

Przykład:

fakt (dobra-rentownosc-sprzedazy-w-roku 2001) zamieniana jest na zdanie „dobra rentownosc sprzedazy w roku 2001”

9. Tworzony jest plik „clp.clp” **zawierający wynikową bazę wiedzy w języku CLIPS** (przed wnioskowaniem). Modyfikator „MAIN:” oznacza nazwę domyślnego modułu (CLIPS pozwala na operacje na podzielonej na moduły bazie wiedzy i dzielenie się definicjami za pomocą słów

kluczowych import i export). Zauważmy, że w **wynikowej bazie wiedzy** pojawiają się już wartości liczbowe, występujące w definicjach faktów (wartości wskaźników, numery lat których dotyczy rozważany okres trzyletni).

```
(deftemplate MAIN::prog "definicje progów wskaźnika ekonomicznego"
```

```
  (slot nazwa-progu)
```

```
  (slot wartosc (type NUMBER)))
```

```
(deftemplate MAIN::years "zdefiniowanie zakresu lat"
```

```
  (slot rok1)
```

```
  (slot rok2)
```

```
  (slot rok3))
```

```
(deftemplate MAIN::wskaznik "definicje wskaźnika ekonomicznego"
```

```
  (slot nazwa-wskaznika)
```

```
  (slot rok1)
```

```
  (slot wartosc1 (type NUMBER))
```

```
  (slot rok2)
```

```
  (slot wartosc2 (type NUMBER))
```

```
  (slot rok3)
```

```
  (slot wartosc3 (type NUMBER)))
```

```
(def facts MAIN::trojka-lat "numery lat"
```

```
  (years (rok1 2000) (rok2 2001) (rok3 2002)))
```

```
(def facts MAIN::progi "progi wskaźników ekonomicznych"
```

```
  (prog (nazwa-progu rentownosc-sprzedazy) (wartosc 0.01)))
```

```
(def facts MAIN::wskazniki "wskazniki ekonomiczne"
```

```
  (wskaznik (nazwa-wskaznika suma-bilansowa) (rok1 2000) (wartosc1 220.869) (rok2 2001) (wartosc2 219.792) (rok3 2002) (wartosc3 228.649))
```

```
  (wskaznik (nazwa-wskaznika suma-bilansowa1) (rok1 2000) (wartosc1 220.869) (rok2 2001) (wartosc2 219.792) (rok3 2002) (wartosc3 228.649))
```

```
  (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok1 2000) (wartosc1 0.0) (rok2 2001) (wartosc2 0.017) (rok3 2002) (wartosc3 0.019))
```

```
  (wskaznik (nazwa-wskaznika zysk-netto) (rok1 2000) (wartosc1 0.0) (rok2 2001) (wartosc2 4.584) (rok3 2002) (wartosc3 4.958))
```

```
  (wskaznik (nazwa-wskaznika przychody-ze-sprzedazy) (rok1 2000) (wartosc1 267.838) (rok2 2001) (wartosc2 273.197) (rok3 2002) (wartosc3 254.675))
```

```
  (wskaznik (nazwa-wskaznika koszty-operacyjne) (rok1 2000) (wartosc1 270.922) (rok2 2001) (wartosc2 245.116) (rok3 2002) (wartosc3 221.768))
```

```
  (wskaznik (nazwa-wskaznika zobowiazania-krotkoterminowe) (rok1 2000) (wartosc1 15.117) (rok2 2001) (wartosc2
```

13.378) (rok3 2002) (wartosc3 17.773)))

```
(defrule MAIN::dobra-rentownosc-w-roku-1
  ?wskaznik <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok1 ?rok1) (wartosc1 ?war))
  ?prog <- (prog (nazwa-progu rentownosc-sprzedazy) (wartosc ?progowa))
  (test (>= ?war ?progowa))
  =>
  (assert (dobra-rentownosc-sprzedazy-w-roku ?rok1)))
```

```
(defrule MAIN::niewystarczajaca-rentownosc-w-roku-1
  ?wskaznik <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok1 ?rok1) (wartosc1 ?war))
  ?prog <- (prog (nazwa-progu rentownosc-sprzedazy) (wartosc ?progowa))
  (test (< ?war ?progowa))
  =>
  (assert (niewystarczajaca-rentownosc-sprzedazy-w-roku ?rok1)))
```

```
(defrule MAIN::dobra-rentownosc-w-roku-2
  ?wskaznik <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok2 ?rok2) (wartosc2 ?war))
  ?prog <- (prog (nazwa-progu rentownosc-sprzedazy) (wartosc ?progowa))
  (test (>= ?war ?progowa))
  =>
  (assert (dobra-rentownosc-sprzedazy-w-roku ?rok2)))
```

```
(defrule MAIN::niewystarczajaca-rentownosc-w-roku-2
  ?wskaznik <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok2 ?rok2) (wartosc2 ?war))
  ?prog <- (prog (nazwa-progu rentownosc-sprzedazy) (wartosc ?progowa))
  (test (< ?war ?progowa))
  =>
  (assert (niewystarczajaca-rentownosc-sprzedazy-w-roku ?rok2)))
```

```
(defrule MAIN::dobra-rentownosc-w-roku-3
  ?wskaznik <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok3 ?rok3) (wartosc3 ?war))
  ?prog <- (prog (nazwa-progu rentownosc-sprzedazy) (wartosc ?progowa))
  (test (>= ?war ?progowa))
  =>
  (assert (dobra-rentownosc-sprzedazy-w-roku ?rok3)))
```

```
(defrule MAIN::niewystarczajaca-rentownosc-w-roku-3
  ?wskaznik <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok3 ?rok3) (wartosc3 ?war))
  ?prog <- (prog (nazwa-progu rentownosc-sprzedazy) (wartosc ?progowa))
  (test (< ?war ?progowa))
  =>
  (assert (niewystarczajaca-rentownosc-sprzedazy-w-roku ?rok3)))
```

```
(defrule MAIN::wzrost-rentownosci-sprzedazy
  ?tymczasowy1 <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok1 ?rok1) (wartosc1 ?war1))
  ?tymczasowy2 <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok2 ?rok2) (wartosc2 ?war2))
  ?tymczasowy3 <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok3 ?rok3) (wartosc3 ?war3))
  (test (> ?war2 ?war1))
  (test (> ?war3 ?war2))
  =>
  (assert (wzrost-rentownosci-sprzedazy w-latach ?rok1 ?rok3)))
```

```
(defrule MAIN::spadek-rentownosci-sprzedazy
  ?tymczasowy1 <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok1 ?rok1) (wartosc1 ?war1))
  ?tymczasowy2 <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok2 ?rok2) (wartosc2 ?war2))
  ?tymczasowy3 <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok3 ?rok3) (wartosc3 ?war3))
  (test (< ?war2 ?war1))
  (test (< ?war3 ?war2))
  =>
  (assert (spadek-rentownosci-sprzedazy w-latach ?rok1 ?rok3)))
```

```
(defrule MAIN::dolek-rentownosci-sprzedazy
  ?tymczasowy1 <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok1 ?rok1) (wartosc1 ?war1))
  ?tymczasowy2 <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok2 ?rok2) (wartosc2 ?war2))
  ?tymczasowy3 <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok3 ?rok3) (wartosc3 ?war3))
  (test (< ?war2 ?war1))
  (test (> ?war3 ?war2))
  =>
  (assert (przelamanie-trendu-spadkowego-rentownosci-sprzedazy w-latach ?rok1 ?rok3)))
```

```
(defrule MAIN::pagorek-rentownosci-sprzedazy
  ?tymczasowy1 <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok1 ?rok1) (wartosc1 ?war1))
  ?tymczasowy2 <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok2 ?rok2) (wartosc2 ?war2))
  ?tymczasowy3 <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok3 ?rok3) (wartosc3 ?war3))
  (test (> ?war2 ?war1))
  (test (< ?war3 ?war2))
  =>
  (assert (utrata-trendu-rosnacego-rentownosci-sprzedazy w-latach ?rok1 ?rok3)))
```

```
(defrule MAIN::dobra-rentownosc-sprzedazy-przez-caly-okres
  ?tymczasowy1 <- (years (rok1 ?rok1) (rok2 ?rok2) (rok3 ?rok3))
  ?tymczasowy2 <- (dobra-rentownosc-sprzedazy-w-roku ?rok1)
  ?tymczasowy3 <- (dobra-rentownosc-sprzedazy-w-roku ?rok2)
  ?tymczasowy4 <- (dobra-rentownosc-sprzedazy-w-roku ?rok3)
  =>
```

```
(assert (dobra-rentownosc-sprzedazy-we-wszystkich-latach ?rok1 ?rok3)))
```

```
(defrule MAIN::niska-rentownosc-sprzedazy-przez-caly-okres
```

```
?tymczasowy1 <- (years (rok1 ?rok1) (rok2 ?rok2) (rok3 ?rok3))
```

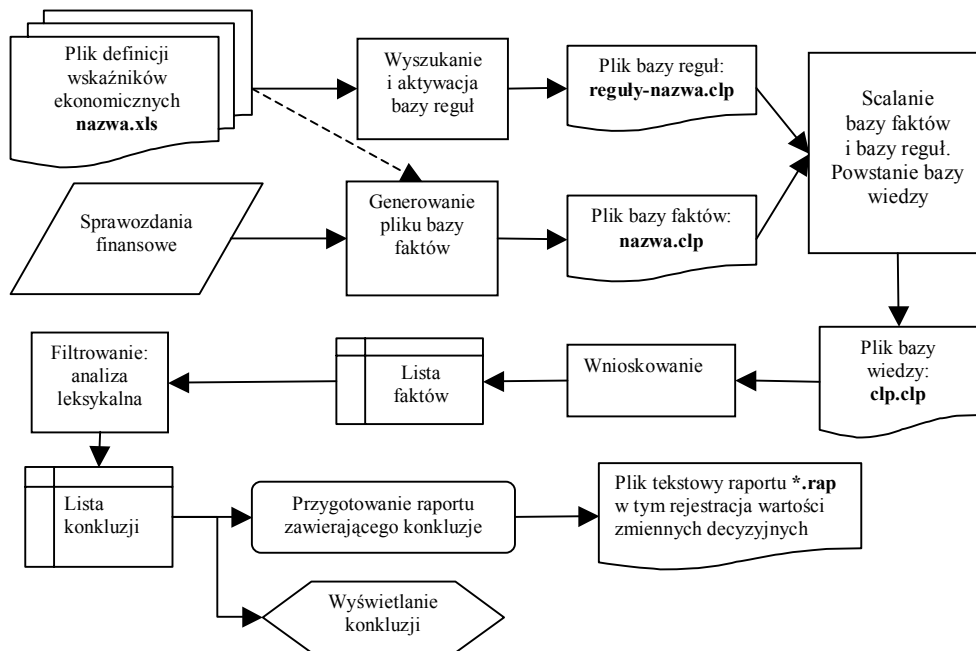
```
?tymczasowy2 <- (niewystarczajaca-rentownosc-sprzedazy-w-roku ?rok1)
```

```
?tymczasowy3 <- (niewystarczajaca-rentownosc-sprzedazy-w-roku ?rok2)
```

```
?tymczasowy4 <- (niewystarczajaca-rentownosc-sprzedazy-w-roku ?rok3)
```

```
=>
```

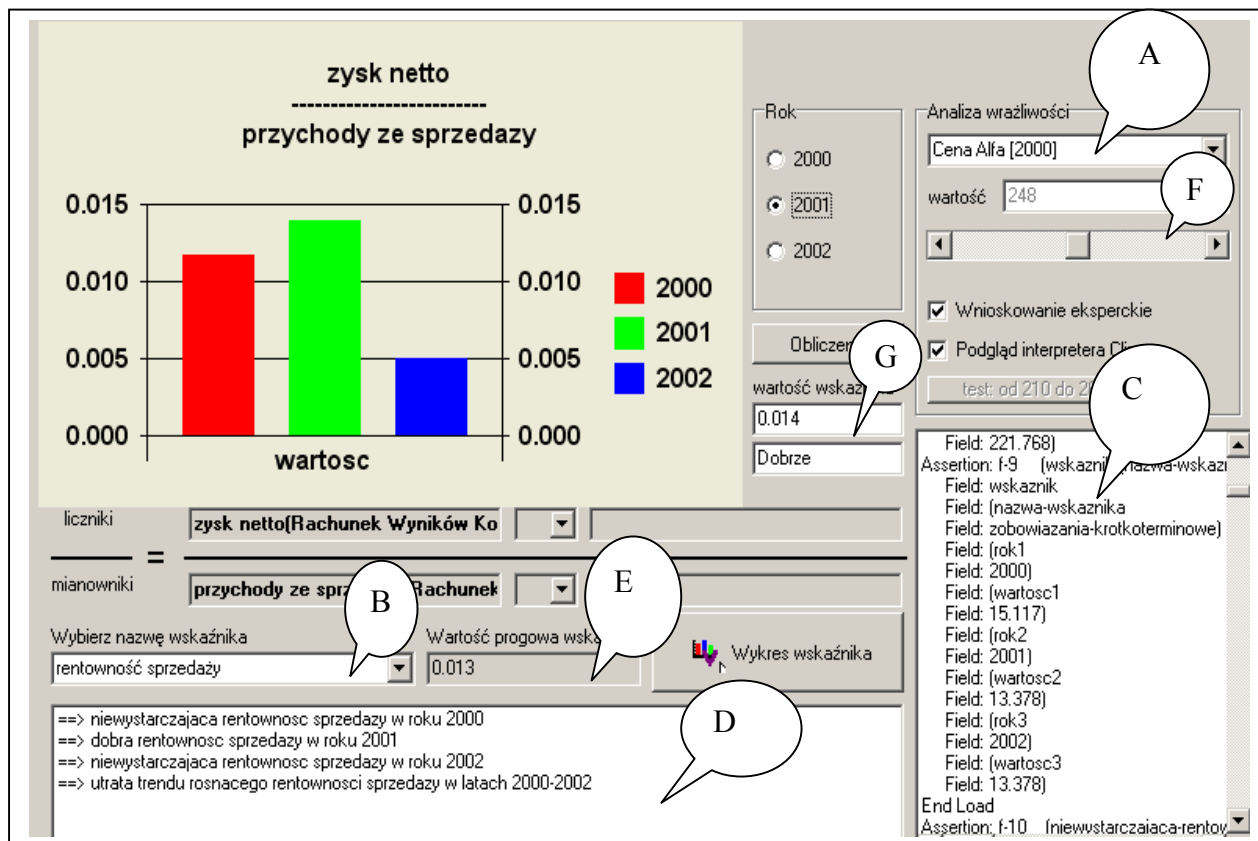
```
(assert (niska-rentownosc-sprzedazy-we-wszystkich-latach ?rok1 ?rok3)))
```



Rys. 6-6 Schemat przetwarzania i przepływu danych w procesie konstruowania bazy wiedzy. Współdziałanie ES shell - CLIPS i SWD Ekanwin. Źródło: [336]

- => dobra płynność bieżąca w roku 2004
- => diagnoza dla roku : 2004 : prawidłowe proporcje między aktywami obrotowymi a pasywami bieżącymi
- => dobra płynność bieżąca w roku 2005
- => diagnoza dla roku : 2005 : prawidłowe proporcje między aktywami obrotowymi a pasywami bieżącymi
- => zbyt wysoka płynność bieżąca w roku 2003
- => diagnoza dla roku : 2003 : wynika ona z wysokiej wartości aktywów obrotowych
- => diagnoza dla roku : 2003 : lub z niskiej wartości pasywów bieżących
- => diagnoza dla roku : 2003 : można rozważyć większe otwarcie na kredytowanie

Rys. 6-7 Przykład skróconego raportu zawierającego konkluzje na podstawie stanu bazy wiedzy po ukonkretnieniu i wnioskowaniu. Źródło: [336]



Rys. 6-8 Formatka „Definiowanie wskaźników”. Wyszczególnienie pól dialogowych, wyniki wnioskowania i analiza wrażliwości w jednej z poprzednich wersji SWD Ekanwin

Na rysunku [Rys. 6-8] wyszczególniono omawiane uprzednio elementy ekranowe związane z modułem systemu eksperckiego:

A: Wybór elementu macierzy zmiennych decyzyjnych; określenie zmiennej decyzyjnej której dotyczyć będzie analiza wrażliwości. Kolejność pozycji w liście rozwijanej zależy od pozycji w formularzu zmiennych decyzyjnych oraz roku dla którego podejmowane będą decyzje.

B: Wybór wskaźnika ekonomicznego – skupienie uwagi. Wybrany wskaźnik ekonomiczny determinuje zawartość bazy wiedzy (architektura tablicowa, respektująca kontekst). Na jego podstawie przeprowadzane będzie wnioskowanie.

C: Okienko zawierające kompletną listę faktów dostępnych w kolejnych sesjach wnioskowania w bazie wiedzy. Na początku listy znajdują się fakty wprowadzane do bazy wiedzy przez Ekanwin. Jak wspomniano wcześniej, tworzone są one na podstawie zawartości pliku definicji wskaźników ekonomicznych (MS Excel), który jest elementem architektury tablicowej zastosowanej w module systemu eksperckiego Ekanwina. Na podstawie sprawozdań finansowych (moduł modelu symulacyjnego) następuje ukonkretnienie faktów, czyli uzupełnienie ich definicji o wartości w rozważanym okresie trzyletnim. W dalszej części listy faktów pojawiają się fakty pochodzące z procesu wnioskowania czyli aktywacji odpowiednich reguł. Jeśli dana reguła pojawi się w ścieżce wnioskowania, to przeprowadzana jest weryfikacja jej przesłankowej części. Zawartość listy faktów wykorzystywana jest przez procedury filtrowania (interpretacji). Jej wyniki prezentowane są w okienku D.

D: Okienko prezentuje w czytelnej formie przefiltrowane fakty pochodzące z kompletnej listy faktów. W okienku znajdują się nowe fakty powstałe w wyniku wnioskowania, dotyczące wybranego wskaźnika ekonomicznego. W przeciwieństwie do okienka C nie są prezentowane fakty tworzone bezpośrednio w module modelu symulacyjnego Ekanwina, nie będące wynikiem wnioskowania a jedynie ukonkretnienia (trzyletniego) pozycji pliku definicji wskaźników ekonomicznych. Repertuar faktów, które mogą się pojawić w okienku D w wyniku procesu wnioskowania jest zdeterminowany przez (1) plik definicji wskaźników, podstawę architektury tablicowej modułu systemu eksperckiego, (2) ustalony przez listę wyboru B wskaźnik ekonomiczny, (3) zbiór reguł wnioskowania.

E: Pole „wartość progowa wskaźnika” służy wprowadzeniu wartości parametru wykorzystywane w regułach wnioskowania dotyczących danego wskaźnika. Domyślna wartość pola pochodzi z pliku zawierającego definicje wskaźników (MS Excel). Wartość znajdująca się w omawianym polu wykorzystywana jest ponadto w procedurze (chmurka G) szybkiego sprawdzania czy wartość badanego wskaźnika w danym roku jest większa („dobrze”) lub mniejsza („źle”) od założonego proggu. Tekstowa reprezentacja reguł i faktów w CLIPS pozwoliła na sparametryzowanie reguł bezpośrednio w programie Ekanwin. Użytkownik podaje wartość parametru w okienku edycyjnym (np. 0.08) a następnie modyfikowany jest dynamicznie fakt: prog (nazwa-progu rentownosc-sprzedazy) (wartosc 0.08) i zapisywany w pliku z wartościami wskaźników. W aktualnej wersji określa się do 2 wartości progowych oraz deklaruje typ przedziału akceptowanych wartości („<”, „>”, „<”, „>”).

F: Suwak (potencjometr) pełni kluczową rolę w analizie wrażliwości. Za jego pomocą dokonuje się modyfikacji badanej zmiennej, ustalonej za pomocą pola wyboru („chmurka” A). Wartość ustawiona za pośrednictwem suwaka jest wyświetlana w polu „wartość” znajdującym się między „chmurkami” A i F. Omawiany element ekranowy (potencjometr) działa podobnie jak standardowy suwak w systemie MS Windows, tzn. dostępne jest (1) zwiększanie i zmniejszanie o przyjętą wartość jednostkową (1 zł, 1% itd. wynikającą z miary dotyczącej badanej zmiennej), (2) szybkie zwiększanie i zmniejszanie wartości o wartość będącą wielokrotnością jednostki (np. 10 zł, 10% itd.), (3) szybkie ustawienie wartości równej ekstremom przyjętego zakresu wartości zmiennej, (4) szybkie przypisanie zmiennej dowolnej wartości z zastrzeżeniem, że należeć będzie do dozwolonego zakresu oraz wynikać będzie z przekształcenia ustawienia suwaka (za jego pomocą można ustawić jedynie wartości całkowite nieujemne w przyjętym zakresie). Poszczególne działania wykonać można w różnorodny sposób: za pomocą klawiatury, myszy z przyciskiem czy pokrętłem. Szczególna przydatność omawianego suwaka wynika z faktu, że każda zmiana jego pozycji (położenia jego punktu uwagi) powoduje zmianę parametrów macierzy zmiennych decyzyjnych aktywnego węzła oraz (1) wykonanie obliczeń (symulacji) na podstawie modelu przedsiębiorstwa zawartego w systemie Ekanwin, (2) przygotowanie sprawozdań finansowych dla aktywnego węzła za dany okres, (3) aktualizację wykresu (jeśli jest w danym momencie aktywny – widoczny) obserwowanego wskaźnika

ekonomicznego lub dowolnego elementu znajdującego się w sprawozdaniach finansowych oraz – jeśli jest to wymagane - (4) aktualizację bazy wiedzy i (5) wykonanie wnioskowania, czyli aktualizacji ocen będących interpretacjami nowych faktów dotyczących badanego wskaźnika. W ten sposób można przeprowadzać manipulacje wartościami poszczególnych zmiennych, zmieniać punkt zainteresowania (wskaźniki, zmienne decyzyjne), wykonywać złożone eksperymenty typu „co się stanie jeśli”, planować eksperymenty, rozpatrywać różnorodne metody optymalizacji. Równocześnie z jakąkolwiek zmianą ustawienia suwaka otrzymujemy uaktualnione, animowane wykresy, wyniki wnioskowania eksperckiego, opinie o wartościach badanych wskaźników. Można także zmieniać zestawy wskaźników, konfigurując poszczególne elementy architektury tablicowej.

G: Pole numeryczne „wartość wskaźnika” oraz towarzyszące mu pole tekstowe pozwalają na obserwację dokładniejszej niż prezentowanej na wykresie wartości danego wskaźnika (lub pozycji sprawozdania finansowego). Jeśli wartość wskaźnika znajdować będzie się w podanym zakresie, to wtedy w polu tekstowym zostanie to zakomunikowane przez napis „dobrze” natomiast jeśli nie, to otrzymamy napis „źle”. W ten sposób możemy bardziej świadomie interpretować i kontrolować wyniki wnioskowania eksperckiego oraz obserwować zachowanie wskaźnika w poszczególnych latach rozważanego okresu trzyletniego.

6.3 Moduł metod analitycznych. Obsługa wskaźników ekonomicznych

Baza wiedzy (moduł systemu eksperckiego) jak i **moduł** (komponent) **wizualizacji** korzystają ze sprawozdań finansowych pochodzących z modułu modelu symulacyjnego. Oba moduły korzystają także z **bazy danych**. W przypadku modułu systemu eksperckiego odbywa się to za pośrednictwem bazy wiedzy, w procedurach interpretacji plików zawierających definicje wskaźników. Moduł wizualizacji korzysta z nich intensywnie podczas przygotowania wykresów. Odbywa się to za pośrednictwem zbioru procedur tworzących **podkomponent** (składnik modułu), interpretujących definicje wskaźników, dokonujących rozłożenia na elementy atomowe (identyfikacja pozycji sprawozdania i jego nazwy, dekodowanie operatorów). Komponent (moduł) wizualizacji korzysta z wyników eksperymentów symulacyjnych przechowywanych w plikach dyskowych (sporządzanie sprawozdań finansowych to dopiero następna czynność wykonywana przez moduł modelu symulacyjnego), składających się na hierarchiczną bazę danych modelu symulacyjnego (o dedykowanej strukturze i formacie zapisu).

Rekordami pierwszej **bazy danych** są poszczególne węzły (sekwencje trzyletnie) drzewa eksperymentu symulacyjnego (dane wejściowe + wyniki). **Wyniki podzielić można** na służące sporządzeniu **sprawozdań** (założenie o zakończeniu ścieżki eksperymentu wraz z danym węzłem) oraz umożliwiające **kontynuowanie eksperymentu**, dodawanie nowych węzłów, jak również dane o niezakończonych czy nierozliczonych inwestycjach, pożyczkach, produkcji w toku, zapasach itd. zgodne z założeniem kontynuacji działania (*going concern*). Moduł wizualizacji ma za zadanie wspomaganie analizy porównawczej i wskaźnikowej.

Druga z baz danych w Ekanwinie **zawiera definicje wskaźników** [Tab. 6-1] i traktowana jest jako **źródło metawiedzy**. Poszczególne komórki arkusza bazy danych (implementacja w MS Excel) powinny zawierać nazwę wskaźnika, typ zalecanego zakresu wartości i wartości progowe, wskazanie na sprawozdanie źródłowe i nazwę pozycji w tym sprawozdaniu, rodzaj operacji matematycznej, specjalne modyfikatory związane z regułowym systemem eksperckim itd. Ze względu na rozpowszechnienie i dostępność narzędzi zarządzających bazą danych wybrano format arkusza MS Excel. Każdy z zestawów wskaźników stanowi odrębną całość i przechowywany jest w odrębnym arkuszu kalkulacyjnym. Opracowano specjalne **procedury przypisujące** wartość dowolnej pozycji w odpowiednim sprawozdaniu do jednej z zadeklarowanych zmiennych (X1 do X90). Zmienne te są jednak dostępne dla użytkownika także w postaci deklarowanych w programie (skrypcie) eksperymentu nazw mnemotechnicznych (jest to zalecany sposób). Procedury te zastosowano także w module systemu eksperckiego i planowania eksperymentu (omówionym dalej).

Tab. 6-1 Fragment arkusza kalkulacyjnego (relacyjnej bazy danych) z zestawem definicji wskaźników i analizowanych wielkości

Nazwa wskaźnika	Sprawozdanie źródłowe licznika	Licznik	Sprawozdanie źródłowe mianownika	Mianownik
suma bilansowa	Bilans Aktywa	suma bilansowa	Bilans Aktywa	Jedynka
rentowność sprzedaży	Rachunek Wyników Koszty i Straty	zysk netto	Rachunek Wyników Przychody i Zyski	przychody ze sprzedaży
zysk netto	Rachunek Wyników Koszty i Straty	zysk netto	Bilans Aktywa	jedynka
sprzedaż	Rachunek Wyników Przychody i Zyski	przychody ze sprzedaży	Bilans Aktywa	jedynka
koszty operacyjne	Rachunek Wyników Koszty i Straty	koszty operacyjne	Bilans Aktywa	jedynka
źródła pieniądza	Płynność Finansowa Przedsiębiorstwa	źródła pieniądza	Bilans Aktywa	jedynka
kapitał własny	Bilans Pasywa	kapitał własny	Bilans Aktywa	jedynka

Istnieją dwa sposoby budowania zestawu wskaźników. **Pierwszy** polega na udostępnieniu analitykowi statycznego, ustrukturalizowanego, niezmiennego zestawu wskaźników. Unika się wtedy pominięcia, choćby przypadkowego, istotnych – wg projektanta systemu – powiązań między danymi ekonomicznymi. Przykładem takiego rozwiązania jest tradycyjny dialog (drzewo pytań) w systemach eksperckich typu *shell*. **Drugi** sposób polega na samodzielnym, prowadzonym przez analityka, budowaniu całego zestawu od postaw. Rozwiązanie takie zgodne jest ze zdaniem Cempela [36], że precyzyjne modele matematyczne, **przyczynowo-skutkowe** można zbudować **jedynie dla układów względnie prostych**, a bardziej złożone systemy wymagają bazy danych umożliwiającej **wnioskowanie statystyczne** oraz przetwarzanie informacji i poszukiwanie wiedzy za pomocą nowoczesnych **technologii informatycznych**.

Wydaje się jednak, że należy przygotować zestaw wskaźników niezbędnych do przeprowadzenia analizy wskaźnikowej przedsiębiorstwa – tzw. **zestaw podręcznikowy**. Moduł systemu eksperckiego może wykorzystywać wskaźniki zawarte w tym zbiorze tworząc rdzeń, czyli zestaw najważniejszych twierdzeń (reguł) uznanych za przydatne w ocenie wyników decyzji kierownictwa i działalności przedsiębiorstwa. **Zbiór reguł wnioskowania**, uznanych przez specjalistów, a zatem takich które nie budzą wątpliwości, powinien w pewnym zakresie wyznaczać właśnie **rdzeń (niezbędny zestaw) wskaźników ekonomicznych**, których nie można pominąć podczas konstruowania SWD. Zbiór ten będzie mógł być ewolucyjnie rozwijany, w toku eksperymentów poznawczych prowadzonych przez projektanta-analityka. Powyżej opisane podejście zastosowano w projektach znanych narzędzi analitycznych. Przykładami są wykorzystujące symulację typu Monte Carlo – systemy Analizator Menedżera (firmy Interlan) i SIAF (firmy AITech).

Jak już wspomniano, **moduł metod analitycznych** korzysta z wyliczonych **wartości wskaźników ekonomicznych** oraz wielkości bezpośrednio dostępnych w **sprawozdaniach finansowych** (zysk, suma bilansowa, wpływy z działalności itp.)

Wskaźniki mogą być definiowane jako **agregaty** zawierające:

- Dane pochodzące z pojedynczego sprawozdania ekonomicznego (wskaźniki strukturalne, ale także np. wskaźniki płynności)

- Dane pochodzące z kilku sprawozdań (różniących się strukturalnie, ale dotyczących tego samego podmiotu). Na nich oparte są wskaźniki obrotowości
- Dane pochodzące z kolejnych okresów zawierające się w tej samej ścieżce decyzyjnej (tzw. wskaźniki dynamiki). Uwzględniono następujące sposoby obliczenia dynamiki: $\frac{\text{wartość[rok+1]}}{\text{wartość[rok]}}$, $\frac{\text{wartość[rok+1]} - \text{wartość[rok]}}{\text{wartość[rok]}}$, $\frac{\text{wartość[rok+1]} - \text{wartość[rok]}}{\text{wartość[rok+1]}}$, $\text{wartość[rok+1]} - \text{wartość[rok]}$
- Dane będące wynikiem zastosowania poszczególnych wariantów decyzyjnych (stosowane w analizie porównawczej)
- Dane wejściowe (zmiennie niezależne) dotyczące podjętych decyzji i dane wyjściowe czyli wyniki (zestawienia porównawcze, wykorzystywane podczas oceny wariantów decyzyjnych).

Repertuar **przekształceń dostępnych danych** może być uzupełniony o **operacje logiczne, arytmetyczne** itp. Warto zauważyć że w konkretnych czynnościach analitycznych niezbędne są najczęściej ograniczone wielkościami zestawy wskaźników. Dzieje się tak dlatego że **zbyt wielka liczba wskaźników** może **utrudnić analizę**. Zopounidis i inni [342] przypominają, że decyzje finansowe podejmuje się na podstawie **wielu czynników ilościowych** (strukturalnych i funkcjonalnych) oraz **jakościowych**, często wzajemnie powiązanych, **co powoduje trudności** w pozyskaniu właściwych informacji finansowych. Aby spełnić powyżej opisane wymagania w niniejszej pracy **zaproponowano** wykorzystanie **modyfikowalnych zestawów wskaźników**. Posiadają one **postać wielowymiarowych arkuszy** (tworzących zbiór plików) z zapisanymi formułami kalkulacyjnymi o **odpowiednio ustalonej strukturze** wspierającą definiowanie wskaźników, ich agregatów i deklarowanie zalecanych zakresów wartości.

6.4 Moduł systemu eksperckiego – charakterystyka języka reprezentacji wiedzy

6.4.1 Środowisko CLIPS jako przykład systemu shell

Hybrydowy SWD Ekanwin wyposażono w **moduł systemu eksperckiego**. Platformą programistyczną Ekanwina jest środowisko Delphi firmy Borland. Umożliwia on korzystanie z **zewnętrznych bibliotek procedur**. Do przeprowadzenia eksperymentów badawczych zastosowano **system szkieletowy CLIPS**. Jest to, jak pisze Chen [40] **język programowania** wyposażony w możliwość korzystania z różnorodnych paradygmatów, zorientowany obiektowo, akceptujący reprezentację wiedzy zarówno w postaci **reguł wnioskowania** jak i **programowania proceduralnego** (sekwencyjne funkcje). Giarratano i Riley [93] określają *CLIPS* jako „język przetwarzania wiedzy opartej na regułach, działający w trybie *forward chaining*. Został zaprojektowany w NASA aby zastosować go w zadaniach wymagających dobrej przenośności, taniego, cechującego się łatwą integracją z innymi systemami”. **Wszystkie wersje** środowiska *CLIPS* (jest np. wersja z obsługą logiki rozmytej) posiadają następujące elementy:

1. Listę faktów (*factlist*) – pełniącą rolę globalnej pamięci (zbiornik danych o dostępie swobodnym)
2. Bazę wiedzy (*knowledge base*) – zawierającą pełny zestaw reguł
3. Mechanizm wnioskujący (*inference engine*) – sterujący procesem poszukiwania rozwiązań oraz przeprowadzający wnioskowanie.

Jest to zatem rozwiązanie standardowe, najczęściej spotykane w SE: wiedza została podzielona na fakty i reguły, które przechowywane są osobno. W literaturze podaje się [40] trzy techniki **akwizycji (pozyskiwania) wiedzy** dla potrzeb środowiska *CLIPS*:

1. Ręczną edycję reguł przez inżyniera wiedzy
2. Preprocesor uczenia maszynowego zgodnego teorią zbiorów przybliżonych (*rough sets theory*)
3. Wbudowanie w wewnętrzną strukturę *CLIPS*-a ułatwień procesu pozyskiwania wiedzy (dostępny jest kod źródłowy *CLIPS* w języku C; sam język *CLIPS* jest dialektem języka LISP).

W **hybrydowym SWD Ekanwin** skorzystano z odmiennego podejścia, jednocześnie wykorzystując **technikę tablicowej reprezentacji wiedzy**. Podejście to polega na **kontekstowym generowaniu fragmentów** (główne założenie architektury tablicowej: *blackboard*) **bazy wiedzy** (zestawu faktów i reguł wnioskowania) w **module odpowiadającym za sporządzanie sprawozdań finansowych**. Kontekst oznacza tu obiekt zainteresowania użytkownika tzn. **zestaw wskaźników ekonomicznych**, dla których Ekanwin ma wspierać analizę wrażliwości, tworzenie raportów, wnioskowanie eksperckie, wykonywanie wykresów.

Technika tablicowej reprezentacji wiedzy, której podstawy zostały omówione przez Mookerjee i Chaturvedi [205] wspomaga **wybór wariantów** modeli decyzyjnych zgodnie z preferencjami użytkownika. Tablicowa reprezentacja wiedzy znalazła także zastosowanie w pakiecie

Aitech DSS [5]. Wykorzystywana jest na etapie wyboru rozwiązań w celu zintegrowania wyników optymalizacji z założeniami i preferencjami formułowanymi przez konkretnego decydena.

6.4.2 Język reprezentacji wiedzy CLIPS

CLIPS jest środowiskiem programistycznym działającym w trybie interpretera. Podstawą funkcjonowania *CLIPS*-a jest wnioskowanie typu *forward-chaining*, reprezentacja wiedzy w postaci reguł wnioskowania (*production system*) i algorytm dopasowywania wzorców typu *Rete*. Baza wiedzy w *CLIPS*-ie zbudowana jest z **faktów** reprezentowanych w postaci zestawu kolejno następujących po sobie pól oraz **reguł** będących złożeniem zestawu warunków (w części lewej) i zestawu działań (w części prawej), które są wykonywane gdy wszystkie przesłanki (warunki) w **LHS** (*left hand side*) reguły są spełnione. Jak piszą Lee i Wu [165] **wadą CLIPS-a** może być wykorzystanie jako strategii aktywowania reguł techniki przeszukiwania *depth-first*. Polega ona na aktywacji reguły dopasowanej (*rule matching*) **do najnowszego faktu** jako pierwszej. Grozi to wpadnięciem w **nieskończoną pętlę**.

W języku *Clips* deklarowanie podstawowych elementów bazy wiedzy przedstawia się następująco [165]:

Przykłady faktów: (lipiec)
 (miesiac lipiec)
 (dane-osobowe „Jan Kowalski” mezczyzna)

Przykłady reguł: (defrule dziadek
 (?X ojciec ?Y)
 (?Y rodzic ?Z)
 =>
 (assert (?X dziadek ?Z)))

Reguła ta odwzorowuje następującą zależność:

jeśli ?X jest ojcem ?Y a ?Y rodzicem ?Z oznacza to że ?X jest dziadkiem ?Z

Fragment (?X ojciec ?Y) (?Y rodzic ?Z) to tzw. *left-hand side (LHS)*, występujący w każdej regule. Jest to **lista przesłanek**, które (wszystkie) muszą być spełnione aby został utworzony nowy fakt (wniosek). **Listę działań** które mają być wykonane jeśli będzie spełniony *LHS* nazywa się *right-hand side (RHS)*. W przykładzie lista działań zawiera pojedyncze działanie; jest to dodanie (asercja)

faktu stwierdzającego pokrewieństwo między dziadkiem a wnukiem. W języku CLIPS wygląda to następująco:

```
(assert (?X dziadek ?Z).
```

Jeżeli zatem w bazie wiedzy (liście faktów) znajdować się będą: (Jan ojciec Karolina) i (Karolina rodzic Maria) to nastąpi **aktywacja reguły** (przeniesienie do tzw. agendy – listy reguł) i **dopisanie faktu** (Jan dziadek Maria). Proces wnioskowania zakończy się, gdy nie będzie już żadnych reguł w agendzie.

Deklarowanie klasy obiektu w CLIPS wygląda następująco:

```
(deftemplate prog "definicje progów wskaźnika ekonomicznego"  
  (slot nazwa-progu)  
  (slot wartosc (type NUMBER)))
```

Powyższa klasa - prog służy do przechowywania wartości progowej wskaźnika ekonomicznego. Obiekty tej klasy wykorzystywane są w regułach wnioskowania.

Reguła dodająca nowy fakt do listy faktów wygląda następująco:

```
(defrule dobra-rentownosc-w-roku-1  
  ?wskaznik <- (wskaznik (nazwa-wskaznika rentownosc-sprzedazy) (rok1 ?rok1) (wartosc1 ?war))  
  ?prog <- (prog (nazwa-progu rentownosc-sprzedazy) (wartosc ?progowa))  
  (test (>= ?war ?progowa))  
  =>  
  (assert (dobra-rentownosc-sprzedazy-w-roku ?rok1)))
```

Powyższa reguła dodaje fakt np. dobra-rentownosc-sprzedazy-w-roku 2000 jeżeli rentowność sprzedaży w roku 2000 jest wyższa niż zadeklarowana wartość progowa.

Reguła sprawdzająca czy rentowność sprzedaży w całym okresie jest zadowalająca:

```
(defrule dobra-rentownosc-sprzedazy-przez-caly-okres  
  (dobra-rentownosc-sprzedazy-w-roku 1997)  
  (dobra-rentownosc-sprzedazy-w-roku 1998)  
  (dobra-rentownosc-sprzedazy-w-roku 1999)  
  =>  
  (assert (dobra-rentownosc-sprzedazy-przez-caly-okres)))
```

Deklaracja faktu w systemie CLIPS wygląda następująco:

```
(deffacts progi "progi wskaźników ekonomicznych"  
  (prog (nazwa-progu rentownosc-sprzedazy) (wartosc 0.05))).
```

W powyższym przykładzie przedstawiono deklaracje progów wskaźnika rentowności sprzedaży na poziomie 5%. Wykorzystany on zostanie w regule „dobra-rentownosc-sprzedazy-w-roku”.

**7 Koncepcja, projekt i zastosowanie
języka sterowania eksperymentem
LEKS**

7.1 System wspomaganie decyzji – Ekanwin – moduł planowania i sterowania eksperymentem symulacyjnym – koncepcja projektowania

7.1.1 Koncepcja języka LEKS

Najbardziej złożony moduł (komponent) systemu wspomaganie decyzji EKANWIN ma na celu dostarczenie **narzędzia planowania i sterowania eksperymentem** umożliwiające pełną kontrolę i automatyzację prowadzonych eksperymentów. Moduł funkcjonuje jako **warstwa oprogramowania między modulem modelu symulacyjnego a interfejsem użytkownika**. Jest elementem architektury typu **klient-serwer**. Założenia modelu klient-serwer [108] w kontekście hybrydowego środowiska systemu wspomaganie decyzji dotyczą ewoluowania formuły programu z paradygmatu sterowania za pomocą systemu menu, przycisków ekranowych, manipulatorów typu mysz, kontroli sprawozdań finansowych i wprowadzania danych w kierunku **konfigurowanego przetwarzania wsadowego**. Moduł powinien umożliwić co najmniej prowadzenie działań optymalizacyjnych (przy wybranej algorytmicznej metodzie optymalizacji).

Przed modulem postawiono **następujące zadania** dotyczące realizowanych funkcji:

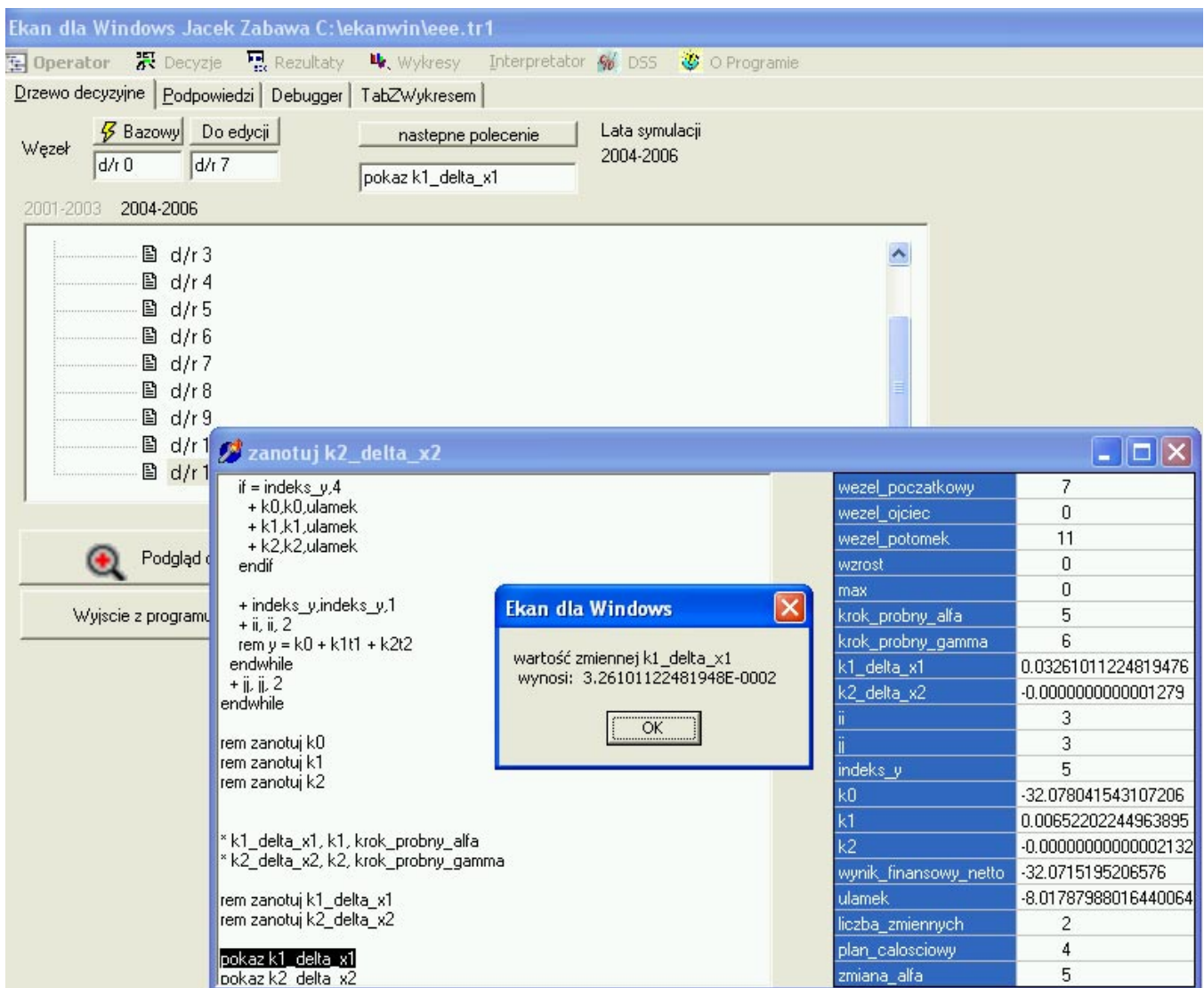
- a) Interpretacja poleceń formułowanym w algorytmicznym języku (LEKS) zbliżonym do języka Pascal
- b) Praca w trybie wsadowym; programy napisane w języku LEKS zapisywane w plikach tekstowych
- c) Posługiwanie się zmiennymi programowymi
- d) Wykonywanie podstawowych działań arytmetycznych, także na zmiennych
- e) Dostępność instrukcji warunkowych i pętli
- f) Możliwość rozdzielania poszczególnych zadań na rozłączne pliki (korzystanie ze zmiennych globalnych)
- g) Dostępność instrukcji: zatrzymania, pełnej prędkości wykonywania programu, pracy krokowej
- h) Multimedialny interfejs (wiele kanałów komunikacji): zasadnicza część informacji (program) przetwarzanych przez interpreter zawarta jest w plikach tekstowych, informacje organizacyjne takie jak wyświetlanie wartości zmiennych decyzyjnych i komunikaty realizowane są przez moduł interfejsu. Rozruch (start) programu i poszczególne instrukcje w trybie pracy krokowej uruchamiane są za pomocą przycisków ekranowych lub systemu menu
- i) Realizacja poleceń dotyczących konfigurowania eksperymentu symulacyjnego (wymienione poniżej) przy założeniu, że pojedyncza instrukcja odpowiada pojedynczemu zadaniu
- j) Wymagania dotyczące interfejsu prezentacji.

Punkty i oraz j wymagają uszczegółowienia. Funkcje udostępniane w tym kontekście przez język LEKS dotyczą:

- Działania na **operatorach** (plikach z drzewami eksperymentów, nazywanych także drzewami decyzyjnymi), takie jak odczytywanie operatorów z dysku, kasowanie i zapisywanie oraz

tworzeniem nowych operatorów

- Działania na węzłach (sekwencjach decyzyjnych) takie jak: dodawanie węzła, zmiana węzła aktywnego, kasowanie węzła, edycja węzła (wprowadzanie wartości zmiennych decyzyjnych), kopiowanie węzła (jw.), ustalenia danego węzła jako bazowego (jeśli jest to węzeł z dostępnymi wynikami eksperymentów)
- Wprowadzania wartości zmiennych decyzyjnych do wnętrza węzłów drzewa decyzyjnego (wypełnianie formularza zmiennych decyzyjnych, za wyjątkiem katalogu inwestycji)
- Wykonywania eksperymentu czyli uruchamianie obliczeń symulacyjnych
- Odczytu wyników eksperymentu, przyjmowanie przez zmienne programowe wartości poszczególnych pozycji sprawozdań finansowych.



Rys. 7-1 Interfejs użytkownika w trakcie wykonywania programu (skryptu) w języku LEKS. Zaznaczono aktualnie wykonywaną instrukcję (komunikat ekranowy dotyczący wartości zmiennej) oraz aktualne wartości zmiennych programowych

Od interfejsu prezentacji oczekuje się implementacji następujących funkcji:

- Prezentacji wartości zmiennych programowych w specjalnym formularzu [Rys. 7-1]

- Prezentacji dynamiki wykonywania programu: wizualizacja aktualnie wykonywanej instrukcji na tle sąsiednich instrukcji (poniżej i powyżej)
- Przedstawienia graficznej postaci drzewa decyzyjnego (własności dynamiczne) analogicznie jak w ręcznym prowadzeniu eksperymentów.

Jak już wspomniano wcześniej, programy zapisane w języku LEKS zapisywane są w plikach tekstowych składających się z **zestawów następujących po sobie rozkazów** odwzorowujących **czynności analityczno-decyzyjne** użytkownika (kierownictwa firmy).

Zestaw rozkazów (instrukcji) języka LEKS obejmuje:

- Rozkazy sterujące
- Pętle (*while...endwhile*) i sprawdzanie warunków (*if...then*), operacje na zmiennych (przypisanie i odczytywanie wartości)
- Podstawowe działania arytmetyczne (dodawanie, odejmowanie, mnożenie, dzielenie)
- Obsługę pracy krokowej z podglądem wartości zmiennych i wyświetlaniem komunikatów
- Wszystkie operacje wykonywane przez użytkownika korzystającego z systemu Ekanwin (tworzenie pliku operatora itd.).

Zadaniem języka LEKS jest umożliwienie **opracowania zewnętrznych programów** wykonujących takie same **czynności operatorskie**, jakie mógłby podjąć **analityk** (tzn. modyfikować zmienne decyzyjne i odczytywać wyniki symulacji). Odbywa się to za pośrednictwem specjalnie zaprojektowanego środowiska **interpretatora planu eksperymentu**, które udostępnia wszystkie podstawowe czynności decyzyjno-analityczne.

Poszczególne czynności reprezentowane są przez odpowiednie **instrukcje programowe**. Można wśród nich wyróżnić instrukcje:

1. Służące nadawaniu wartości zmiennych decyzyjnych (operujące na macierzy zmiennych decyzyjnych w trybie zapisu)
2. Służące odczytywaniu wartości pozycji poszczególnych sprawozdań finansowych oraz ustalonych uprzednio (lub domyślnych) wartości komórek macierzy zmiennych decyzyjnych
3. Instrukcje operujące na poziomie topologii drzewa eksperymentów symulacyjnych (dodawanie, kasowanie, kopiowanie, aktywacja węzłów oraz identyfikacja relacji między węzłami).

Oprócz **instrukcji odczytujących** i **zapisujących** informacje pochodzące z modelu przedsiębiorstwa zaimplementowanego w systemie Ek_An, interpretator wyposażono w instrukcje sterujące procesem wykonywania programu (eksperymentów) oraz dostępu do zmiennych programowych. Ze względu na **krokowy** (sekwencyjny) **charakter pracy interpretatora** języka LEKS zawarte są w nim instrukcje deklarowania zmiennych programowych oraz wykonywania podstawowych działań arytmetycznych. Zmienne programowe mogą być wykorzystywane podczas odwoływania się do instrukcji odczytu i zapisu parametrów, zmiennych wejściowych i wyjściowych

modelu przedsiębiorstwa. Za sterowanie wykonywaniem zdefiniowanych sekwencyjnych zbiorów instrukcji odpowiadają instrukcje w rodzaju *if ... then .. endif* oraz *while .. endwhile*. Pierwsza służy do realizacji wyboru wariantów dalszego działania na podstawie logicznej wartości warunku, natomiast druga odpowiada za realizację struktury pętli. Zestaw instrukcji dopełniają wywołania zewnętrznych procedur złożonych z sekwencji instrukcji wykonywanych wielokrotnie lub stanowiących logicznie wyodrębnioną całość (np. optymalizacja ceny danego wyrobu w konkretnym roku).

7.1.2 Perspektywa zastosowań języka LEKS

Istota prowadzenia **działań optymalizacyjnych** za pomocą programu (skryptu) napisanego w języku LEKS polega na wykonaniu następujących czynności:

- Akceptacji danej decyzji (pozostanie węzła w drzewie)
- Modyfikacji decyzji
- Skopiowaniu zawartości formularza zmiennych decyzyjnych do nowego węzła (traktowanie jako decyzji wzorcowej, przeznaczonej do dalszych badań)
- Usunięciu danej decyzji po uznaniu jej za nieakceptowalną.

Konstrukcja **interpretatora LEKS** zintegrowanego z platformą Ekanwin umożliwia **zmierzenie się** z następującymi problemami badawczymi:

- Badaniem wpływu wielkości zamówień produkcyjnych na poziom zapasów z uwzględnieniem przebiegu zamówień na poszczególne wyroby w poszczególnych latach
- Wyznaczaniem optymalnej trajektorii cen w danym okresie dla wielokryterialnej funkcji celu w postaci „X * zysk netto+Y * sprzedaż”
- Badaniem wpływu obserwowanej wysokości rentowności danej linii produkcyjnej na jej priorytet przy poszukiwaniu optymalnej struktury cen
- Wyborem źródeł finansowania inwestycji: środki własne czy kredyt inwestycyjny
- Wyborem wielkości i źródeł pozyskiwania kapitału: emisja akcji (uwarunkowana dochodowością przedsiębiorstwa) czy raczej kredyt hipoteczny (uwarunkowany jego majątkiem)
- Efektywnością (np. liczba iteracji, wielkość ryzyka) optymalizacji cen wyrobów lub zamówień surowców dla funkcji celu uwzględniającej wielkość zysku
- Odkrywaniem relacji między z jednej strony zamówieniami surowców, cenami, inwestycjami i zadłużeniem a płynnością z drugiej strony
- Formułowaniem heurystycznych zasad poszukiwania optymalnych rozwiązań: kolejność wyboru optymalizowanych wielkości zmiennych niezależnych (decyzyjnych) np. produkcja, ceny, inwestycje, optymalizowanie „wprzód” lub „w tył”
- Weryfikacją reguł decyzyjnych (konstruowanych na potrzeby modułu systemu eksperckiego)
- Implementacją metod optymalizacyjnych (na podstawie ogólnie dostępnych algorytmów).

Język LEKS zaimplementowany w hybrydowym SWD Ekanwin umożliwia **opracowanie biblioteki procedur** wspomagających poszukiwanie suboptymalnych decyzji. Poszczególne działy (zestawy plików) obejmują pewien zakres metodologii (wyróżnionych metod optymalizacyjnych) lub zakres przedmiotowy (horyzont optymalizacji, wycinek macierzy zmiennych decyzyjnych, dopuszczalny zakres wartości zmiennych decyzyjnych (dziedzinę funkcji celu)). Glover i inni [96] postulują rozwiązywanie wielu problemów dzięki połączeniu symulacji i optymalizacji np.:

- Ustaleniu najlepszej konfiguracji obrabiarek w procesie produkcyjnym
- Optymalizacji planowania finansowego: wyznaczenie najlepszego portfolio
- Planowaniu w zarządzaniu personelem: ustaleniu parametrów decyzji skutkującej najlepszym wykorzystaniem zatrudnionych.

7.1.3 Wprowadzenie do badań wykonywanych za pomocą języka LEKS

Wiele **czynności analitycznych** można zapisać w sposób **algorytmiczny** wykorzystując interpretator **autorskiego języka planowania eksperymentu** LEKS. Język LEKS pozwala na prowadzenie badań polegających na wykonaniu **serii eksperymentów o strukturze zdefiniowanej z góry** lub **modyfikowanej w trakcie eksperymentów** na podstawie obserwowanych wyników. Można zatem otrzymać **zbiór wartości zmiennej zależnej** w funkcji dziedziny zmiennej niezależnej (np. zmiennych decyzyjnych), przygotować **dane dla wykresów zależności między zmiennymi** modelu, **poszukiwać rozwiązań optymalnych** poprzez implementację odpowiednich algorytmów, **prowadzić eksperymenty** wg określonego planu.

W przypadku programów poszukiwania **rozwiązań optymalnych** można respektować także określone ograniczenia nałożone na zmienne, jeśli takie występują w modelu sytuacji problemowej. Można je uwzględnić za pomocą np. **funkcji kary** czyli w postaci wyrażenia matematycznego reprezentującego np. naruszenie zużycia zasobów (nawet jeśli jest niedostępne explicite w żadnym ze sprawozdań) lub **przeglądu znalezionych rozwiązań** pod kątem respektowania ograniczeń.

Opracowanie i implementacja **języka sterowania eksperymentem** umożliwiły prowadzenie badań o wyszczególnionych wyżej cechach. Moduł języka LEKS został **zintegrowany z symulatorem** przedsiębiorstwa, otrzymał **specjalizowany interfejs** za pomocą którego komunikuje się z analitykiem, umożliwia **pełne objęcie kontroli** nad prowadzonymi **eksperymentami**, tzn. **zastąpienie użytkownika**, jeśli chodzi o **czynności wyznaczania i wprowadzania wartości zmiennych decyzyjnych**, analizy danych wejściowych i wyjściowych (efektów podejmowanych decyzji) i **podejmowanie decyzji** o prowadzonych czynnościach analityczno-decyzyjnych.

Tworzenie zestawu instrukcji (programu) w języku sterowania eksperymentem wspomagane jest przez moduł definiowania wskaźników w którym odwołanie do pozycji danego sprawozdania finansowego reprezentowane jest w postaci graficzno-tekstowej (zagnieżdżona struktura sprawozdań)

oraz tekstowej, przedstawiającej odwoływanie się do danej pozycji w formacie języka LEKS. W ten sposób zminimalizowane zostaje niebezpieczeństwo błędów kodowania, występujące szczególnie w przypadku sprawozdań finansowych o złożonej strukturze, takich jak bilans.

W dalszej części pracy zaprezentowano zastosowania języka LEKS w poglądowych badaniach zmienności funkcji niektórych zmiennych decyzyjnych a następnie w zadaniach optymalizacji wyników symulacyjnego modelu przedsiębiorstwa przemysłowego na przykładzie systemu EKANWIN. W tym celu zaprezentowano programy (wraz z wynikami) implementujące wybrane algorytmy: połowienia przedziałów, metodę Hooke'a-Jeevesa, algorytm genetyczny. Przedstawiono także programy w języku LEKS wspomagające optymalizacyjną analizę czynnikową (na przykładzie techniki planowania eksperymentu wg techniki Boxa-Wilsona w tym planu dwupoziomowego).

7.1.4 Elementy języka LEKS

Poniżej wymieniono pogrupowane na podstawie kryterium realizowanych funkcji dostępne w interpretatorze języka LEKS instrukcje (rozказы). Bardziej formalny opis języka zamieszczono w Dodatku.

Instrukcje **organizacyjne** są następujące:

1. **rem komentarz**; wszystkie znaki występujące po słowie REM nie są analizowane przez interpreter
2. **message komunikat**; wyświetlenie komunikatu w okienku dialogowym (wyskakującym na pulpit)
3. **pokaz Xnumer_zmiennej**; wyświetlenie wartości wyspecyfikowanej zmiennej w okienku dialogowym (wyskakującym na pulpit). Przykład: pokaz X30
4. **pokaz nazwa_zmiennej**; wyświetlenie wartości wyspecyfikowanej zmiennej w okienku dialogowym (wyskakującym na pulpit). Przykład: pokaz zysk_netto

Instrukcje dotyczące **obsługi plików** (operatorów, drzew decyzyjnych rozumianych jako **całości**):

1. **wczytaj plik** nazwa_pliku; załadowanie do pamięci roboczej Ekanwina pliku operatora zawierającego drzewo eksperymentów symulacyjnych. Aktywacja zmiennych i nadanie im wartości początkowych. Prezentacja struktury drzewa w odpowiednim okienku. Warunek: plik operatora musi istnieć w wybranym katalogu
2. **utwórz plik** nazwa_pliku; utworzenie w pamięci roboczej Ekanwina najprostszego drzewa eksperymentów symulacyjnych, zawierającego węzeł-korzeń z trzyletnią sekwencją inicjującą. Aktywacja zmiennych i nadanie im wartości początkowych. Prezentacja struktury drzewa w odpowiednim okienku. Zapisanie drzewa eksperymentu (operatora) do pliku o podanej nazwie.

Uwaga: wszystkie operacje wykonywane na drzewie eksperymentu symulacyjnego niezwłocznie znajdują odzwierciedlenie w pliku operatora. Dlatego też nie ma potrzeby wprowadzenia osobnego

polecenia zapisywania operatora w pliku dyskowym.

Zestaw instrukcji służących do **przemieszczania się po poszczególnych węzłach** drzewa decyzyjnego:

1. **ustaw węzeł** numer_wezla; aktywacja wybranej trzyletniej sekwencji eksperymentu symulacyjnego (węzła drzewa eksperymentu). Od tej chwili instrukcje operujące na danych wejściowych (zmiennych decyzyjnych) i wyjściowych (sprawozdania finansowe) dotyczyć będą wybranego węzła
2. **dodaj węzeł** numer_wezla_ojca, numer_wezla_kopiowanego, numer_dodanego_wezla; Utworzenie w drzewie decyzyjnym nowego węzła. Określić należy węzeł bazowy (czyli ten który reprezentuje stan eksperymentu na koniec poprzedniej sekwencji trzyletniej) oraz węzeł kopiowany (czyli ten który będzie traktowany jako wzorzec wartości występujących w formularzu zmiennych decyzyjnych). Oczywiście możliwa jest późniejsza modyfikacja wartości zmiennych decyzyjnych. Eksperyment reprezentowany przez dodawany węzeł będzie kontynuacją eksperymentu prowadzonego zgodnie z wartościami przyjętymi w węzle bazowym (rozszerzona zostanie ta ścieżka eksperymentu symulacyjnego). Zmienna podstawiona w miejscu parametru „numer_dodanego_wezla” zawierać będzie numer (identyfikator) automatycznie nadany nowemu węzłowi. Uwaga: podanie zmiennej przechowującej numer nowego węzła jest opcjonalny
3. **kasuj węzeł** numer_wezla; wyszczególniony węzeł zostanie usunięty z drzewa decyzyjnego
4. **symuluj węzeł** numer_wezla; jeśli sekwencja decyzyjna reprezentowana przez węzeł nie była jeszcze wykorzystana do przeprowadzenia eksperymentu (nie zostały przeprowadzone obliczenia), to wykonana zostanie symulacja i będzie można odczytać zawartość odpowiednich sprawozdań finansowych
5. **odczyt węzeł** numer_wezla; numer porządkowy (identyfikator) aktywnego węzła (sekwencji decyzyjnej) zostaje przypisany do wybranej zmiennej
6. **odczyt ojciec** numer_wezla_źródłowego, numer_wezla_ojca; numer porządkowy (identyfikator) bezpośredniego przodka aktywnego węzła (sekwencji decyzyjnej) zostaje przypisany do wybranej zmiennej.

Zestaw instrukcji służących do **wypełnienia formularza** (macierzy) **zmiennych decyzyjnych**.

Wartość **każdego elementu** macierzy jest jednak **ustawiana osobno**:

1. **zmiana zlecenia** alfa|beta|gamma rok1|rok2|rok3 wartosc; ustawienie wartości fragmentu macierzy zmiennych decyzyjnych, dotyczącego wielkości zleceń produkcyjnych. Podmacierz zawiera 9 elementów (3 wyroby x 3 lata)
2. **zmiana cena** alfa|beta|gamma rok1|rok2|rok3 wartosc; ustawienie wartości fragmentu macierzy zmiennych decyzyjnych, dotyczącego wysokości cen sprzedawanych wyrobów. Podmacierz zawiera 9

elementów (3 wyroby x 3 lata)

3. **zmiana zamówienia surowców** white|black|red rok1|rok2|rok3 wartosc; ustawienie wartości fragmentu macierzy zmiennych decyzyjnych, dotyczącego wysokości cen sprzedawanych wyrobów. Podmacierz zawiera 9 elementów (3 surowce x 3 lata)

4. **zmiana dywidenda** rok1|rok2|rok3 wartosc; ustawienie wartości fragmentu macierzy zmiennych decyzyjnych, dotyczącego wysokości dywidendy. Podmacierz zawiera 3 elementy (3 lata)

5. **zmiana zadłużenie** rok1|rok2|rok3 wartosc; ustawienie wartości fragmentu macierzy zmiennych decyzyjnych, dotyczącego stopy zadłużenia. Podmacierz zawiera 3 elementy (3 lata)

6. **zmiana emisja akcji** rok1|rok2|rok3 wartosc; ustawienie wartości fragmentu macierzy zmiennych decyzyjnych, dotyczącego wielkości emisji akcji. Podmacierz zawiera 3 elementy (3 lata).

Zestaw instrukcji służących do **odczytania wartości poszczególnych macierzy zmiennych wyjściowych** (sprawozdań finansowych). Wartość każdego elementu interesującej macierzy jest jednak odczytywana osobno i podstawiana do wyspecyfikowanej zmiennej. Formaty sprawozdań mają strukturę właściwą dla wykorzystywanego modelu przedsiębiorstwa:

1. **odczyt zmienne decyzyjne** numer_wiersza, strona, rok, Xnumer_zmiennej; uwaga: jest to wyjątek, gdyż chodzi tu o kontrolne odczytywanie wartości elementów macierzy zmiennych decyzyjnych

2. **odczyt wynik syntetyczny** numer_wiersza, numer_kolumny, Xnumer_zmiennej; odczyt sprawozdania „wynik syntetyczny”

3. **odczyt wskaźniki płynności** numer_wiersza, numer_kolumny, Xnumer_zmiennej; odczyt sprawozdania „wskaźniki płynności”

4. **odczyt rachunek wyników** numer_wiersza, strona,rok, Xnumer_zmiennej; odczyt sprawozdania „rachunek wyników”

5. **odczyt bilans** numer_wiersza, strona,rok, Xnumer_zmiennej; odczyt sprawozdania „Bilans”

6. **odczyt zmienne decyzyjne** numer_wiersza, strona,rok, Xnumer_zmiennej

7. **odczyt wskaźniki rentowności** numer_wiersza, strona, rok, Xnumer_zmiennej; odczyt sprawozdania „wskaźniki rentowności”

8. **odczyt wskaźniki zadłużenia** numer_wiersza, strona, rok, Xnumer_zmiennej; odczyt sprawozdania „wskaźniki zadłużenia”

9. **odczyt wskaźniki obrotowości** numer_wiersza, strona, rok, Xnumer_zmiennej; odczyt sprawozdania „wskaźniki obrotowości”.

Zestaw instrukcji służących do **sprawdzania logicznej prawdziwości wyspecyfikowanych warunków** w postaci *jeśli warunek to akcja*. W zależności od ich **wartości logicznej** podejmowane są odpowiednie działania. Za pomocą **instrukcji warunkowych** reprezentowane jest podejmowanie

decyzji. Zapisujemy to następująco:

if (modyfikator warunku) (zmiennie wykorzystywane w specyfikacji warunku) ...ciąg instrukcji wykonywanych jeśli warunek jest prawdziwy... endif

W ogólnosci: **if greater_** x?? x??

1. **if greater_** X1 X2; Warunek ma wartość logiczną „prawda” jeśli $X1 > X2$
2. **if great_eq** X1 X2; Warunek ma wartość logiczną „prawda” jeśli $X1 \geq X2$
3. **if less___** X1 X2; Warunek ma wartość logiczną „prawda” jeśli $X1 < X2$
4. **if less_eq_** X1 X2; Warunek ma wartość logiczną „prawda” jeśli $X1 \leq X2$
5. **if equal___** X1 X2; Warunek ma wartość logiczną „prawda” jeśli $X1 = X2$
6. **if notequal** X1 X2; Warunek ma wartość logiczną „prawda” jeśli $X1 \neq X2$

Zestaw instrukcji służących do przypisania danej zmiennej wartości oraz wykonania elementarnych operacji arytmetycznych. Zapisujemy to następująco:

W ogólnosci: + X??, X??, X??

1. **przypisz** X2, wartość_liczbowa (inny wariant: przypisz X2,X1); przykład: przypisz X2,100 spowoduje nadanie zmiennej X2 wartości 100. Natomiast przypisz X2, X1 spowoduje skopiowanie wartości zmiennej X1 do zmiennej X2
2. * X3, X2, X1 (inny wariant: * X3, wartość_liczbowa_1, wartość_liczbowa_2); efektem powyższej instrukcji będzie przyjęcie przez zmienną X3 wartości równej iloczynowi X2 i X1
3. / X3, X2, X1 (inny wariant: / X3, wartość_liczbowa_1, wartość_liczbowa_2); efektem powyższej instrukcji będzie przyjęcie przez zmienną X3 wartości równej ilorazowi X2 i X1
4. + X3, X2, X1 (inny wariant: + X3, wartość_liczbowa_1, wartość_liczbowa_2); efektem powyższej instrukcji będzie przyjęcie przez zmienną X3 wartości równej sumie X2 i X1
5. - X3, X2, X1 (inny wariant: + X3, wartość_liczbowa_1, wartość_liczbowa_2); efektem powyższej instrukcji będzie przyjęcie przez zmienną X3 wartości równej sumie X2 i X1
6. **clear**; przypisanie wszystkim zmiennym globalnym (x10..x100) programu wartości zero.

Pozostaje jeszcze bardzo przydatna, **elementarna instrukcja pętli**:

while operator $x \geq y$, $x \leq y$

(...)

endwhile

Przykład:

while $x \geq$ zmienna1, zmienna2

(zestaw instrukcji)

endwhile

Uwaga: *zestaw instrukcji* będzie wykonywany dopóty, dopóki $x1$ będzie większe lub równe $x2$

7.2 Przykłady badań wykonywanych za pomocą języka LEKS wspomaganym interfejsem użytkownika

7.2.1 Zastosowanie metody bisekcji

Spróbujemy teraz zastosować metody optymalizacji funkcji jednej zmiennej w zadaniu optymalizacji funkcji wielu zmiennych w następujący sposób. Zamiast rozpatrywać całościowo podzbiory macierzy zmiennych decyzyjnych, zbadamy jak na otrzymane rozwiązanie wpływa rozbieżność wielowymiarowych problemów optymalizacyjnych na szereg zastosowań prostych, wydajnych metod optymalizacji funkcji jednej zmiennej. Wśród numerycznych metod optymalizacji funkcji jednej zmiennej wymienić należy metodę bisekcji i metodę złotego podziału.

Sformułowanie ramowego problemu badawczego (algorytm bisekcji):

Przeprowadzić badania dotyczące **planowania eksperymentu** mającego na celu suboptymalizację zmiennej wyjściowej środowiska hybrydowego systemu wspomaganego decyzji, sformułowanej za pomocą narzędzia definiowania wskaźników ekonomicznych (interfejs użytkownika). Zmienna wejściowa powinna być wskaźnikiem ekonomicznym - agregatem wybranych pozycji sprawozdań finansowych lub wprost pozycją wybranego sprawozdania finansowego mającą istotne znaczenie w procesie oceny funkcjonowania przedsiębiorstwa. **Wybrać należy zmienne wejściowe** które będą modyfikowane, określić ich **dopuszczalne zakresy**, dokonać próby sformułowania **zasad projektowania eksperymentu**, **zidentyfikować elementarne struktury**, których wyodrębnienie jest niezbędne dla prowadzenia eksperymentów.

Konkretyzacja problemu badawczego:

Celem operacyjnym jest osiągnięcie **jak największej wartości zysku netto** na koniec badanego okresu. Jednocześnie powinno zapewnić się jak najwyższą wartość zysku netto w poszczególnych latach. Wartości komórek macierzy zmiennych decyzyjnych oparte będą na macierzy pochodzącej z poprzedniego okresu (bazowej). W celu optymalizacji modyfikowany będzie podzbiór macierzy zmiennych decyzyjnych, zawierający ceny wyrobów gotowych. Zakres dopuszczalnych zmian cen zawiera wartości pochodzące z poprzedniego okresu, tzn. że wartości lewych krańców przedziałów są mniejsze a wartości prawych krańców przedziałów są większe niż odpowiednie wartości bazowe.

Założenia:

Niech a – początek badanego przedziału, b – koniec badanego przedziału, $f(x)$ – wartość optymalizowanej funkcji, d – szerokość badanego przedziału

Algorytm bisekcji przedstawić można za pomocą następujących etapów:

1. Jeśli $f(b) > f(a)$ to podstawiamy $c = (a+b) / 2$ i następnie $a = c$
2. Jeśli $f(b) < f(a)$ to podstawiamy $c = (a+b) / 2$ i następnie $b = c$
3. Jeśli $f(b) = f(a)$ lub $d = |b-a|$ jest wystarczająco mały to kończymy
4. wracamy do 1

Metoda powinna znaleźć przynajmniej jedno maksimum funkcji (w końcowym przedziale). Po n krokach szerokość przedziału zawierającego maksimum wynosi $d_n = d_0 / 2^n$

Podobną metodą jest metoda złotego podziału (*sectio aurea*). Polega ona m.in. na zastosowaniu proporcji harmonicznego podziału odcinka. Proporcja harmoniczna ma miejsce gdy części odcinka oznaczone jako a i b pozostają do siebie w stosunku $a : b = b : (a - b)$

Implementacja metody bisekcji dla pojedynczej zmiennej (cena wyrobu ALFA w pierwszym roku sekwencji decyzyjnej, optymalizowany jest zysk netto wykazany na koniec sekwencji) w języku LEKS ma zaprezentowaną poniżej postać.

Założenia:

- obszar poszukiwań rozwiązania: [200,290] wyrażony jest w złotych
- znany węzeł (sekwencja) bazowy
- znany węzeł (sekwencja) wzorcowy (początkowy)
- priorytetem jest jak najmniejsza liczba węzłów zaangażowanych w obliczenia (kasujemy węzeł gdy tylko przestaje być potrzebny)
- wartość liczbowa wykorzystywana w *pierwszym warunku* (rozmiar badanego przedziału) zatrzymania algorytmu wynosi 0.00005
- wartość liczbowa wykorzystywana w *drugim warunku* (liczby iteracji) zatrzymania algorytmu wynosi 20. Gdy *warunek pierwszy* jest spełniony to dopuszczalna liczba iteracji spada do zera
- po zakończeniu obliczeń kasowany jest węzeł wzorcowy (początkowy).

Metodę poszukiwania rozwiązania optymalnego (maksimum) przez **połowienie przedziałów** przedstawić można w postaci następującego algorytmu [Rys. 7-2]:

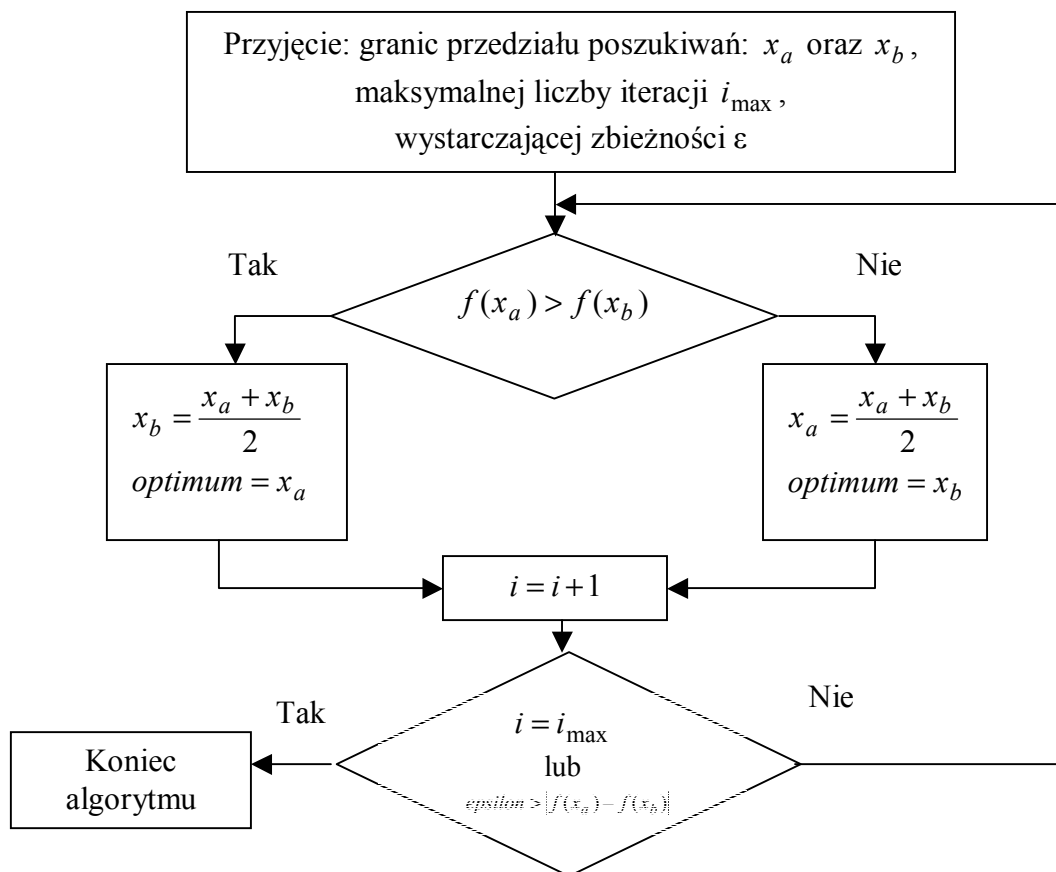
Określić początkowe granice przedziału w którym poszukiwane będzie maksimum, czyli argument funkcji celu: x_a i x_b , warunek zakończenia algorytmu: liczba iteracji oraz zbieżność (ϵ)

Wyznaczyć wartość funkcji celu $f(x_a)$ i $f(x_b)$

Jeśli $f(x_a) > f(x_b)$ to optimum poszukiwane będzie w przedziale $(x_a, (x_b + x_a) / 2)$. Nowa prawa granica przedziału poszukiwań będzie równa $x_b = (x_b + x_a) / 2$ a robocze optimum stanowi argument x_a

Jeśli $f(x_a) \leq f(x_b)$ to optimum poszukiwane będzie w przedziale $((x_b + x_a)/2, x_b)$. Nowa lewa granica przedziału poszukiwań będzie równa $x_a = (x_b + x_a)/2$ a robocze optimum stanowi argument x_b .
 Jeśli liczba iteracji przekracza założoną to zakończyć algorytm z wartością roboczą optimum.
 Jeśli obserwuje się zbieżność rozwiązań czyli $\epsilon > |f(x_b) - f(x_a)|$ to zakończyć algorytm z wartością roboczą optimum.

Zwiększyć licznik iteracji i przejść do kroku nr 2



Rys. 7-2 Schemat algorytmu połowienia przedziałów

Metoda połowienia przedziałów przedstawiona zostanie na przykładzie zadania optymalizacji wartości podzbioru macierzy zmiennych decyzyjnych. Optymalizacji podlegać będą ceny wyrobów gotowych, zlecenia produkcyjne oraz zamówienia surowców niezbędnych w procesie produkcyjnym. Zadanie rozwiązane zostanie w trzech etapach. Wymiary podzbioru macierzy podlegającego optymalizacji to $[3(\text{identyfikatory wyrobów i surowców}) \times 3(\text{liczba przedmiotów optymalizacji}) \times 3(\text{lata})]$ czyli $[9 \times 3]$. Zadanie optymalizacji podzielono na trzy etapy (kolejność przedmiotowa: ceny, zlecenia produkcyjne, zamówienia surowców). Funkcją celu będzie zysk netto. Optymalizacja przeprowadzana jest kolejno w poszczególnych komórkach macierzy. Dlatego też metoda rozwiązania zawierać musi opis strategii (konfiguracji) dotyczący założonej kolejności poszczególnych operacji w czasie optymalizacji.

Elementarną strukturą składającą się na złożone struktury tworzące plan eksperymentu jest procedura optymalizacji w której dochodzi do modyfikacji wartości pojedynczego elementu macierzy

zmiennych decyzyjnych. Poszczególne struktury elementarne wykorzystywane są w formułowaniu heurystycznych zasad poszukiwania optymalnych rozwiązań obejmujących kolejność wyboru optymalizowanych wielkości zmiennych niezależnych (decyzyjnych) np. produkcja, ceny, inwestycje oraz okresów (fragmentów sekwencji) w obrębie których prowadzi się optymalizację. W tym przypadku jest to cena danego wyrobu stała dla jednego z trzech lat formułujących sekwencję decyzyjną.

Elementarną strukturą składającą się na złożone struktury tworzące **plan eksperymentu** jest **procedura optymalizacji w której dochodzi do modyfikacji wartości pojedynczego elementu macierzy zmiennych decyzyjnych**. Poszczególne struktury elementarne wykorzystywane są w formułowaniu heurystycznych zasad poszukiwania optymalnych rozwiązań obejmujących kolejność wyboru optymalizowanych wielkości zmiennych niezależnych (decyzyjnych) np. produkcja, ceny, inwestycje oraz okresów (fragmentów sekwencji) w obrębie których prowadzi się optymalizację. W tym przypadku jest to cena danego wyrobu stała dla jednego z trzech lat formułujących sekwencję decyzyjną.

Planując badania warto także określić sposób wyboru optymalizowanych w następnym kroku elementarnych struktur:

- a) wybór z góry określony, tzn. postępuje się zgodnie z ustanowioną strategią (kolejnością) na którą nie mają wpływu wyniki eksperymentów
- b) wybór zdeterminowany wynikami poprzednich eksperymentów (przykładem jest trój etapowa metoda połowienia przedziałów).

Dla planów eksperymentów wyróżnić można następujące grupy strategii:

- a) **losowy wybór następnej optymalizowanej struktury elementarnej** (zapobiegając jednocześnie powtarzaniu już optymalizowanej struktury) aż do wykonania optymalizacji wszystkich przewidzianych struktur elementarnych
- b) **zdeterminowane strategie**; do ich zilustrowania zaproponowano następującą **notację graficzną** z poniższymi oznaczeniami:

- „ - ” brak optymalizacji,
- „ + ” optymalizowano;

od lewej do prawej: kolejne lata; od góry do dołu: kolejne wyroby – Alfa, Beta, Gamma, tzn.:

$$\begin{bmatrix} Alfa - rok1 & Alfa - rok2 & Alfa - rok3 \\ Beta - rok1 & Beta - rok2 & Beta - rok3 \\ Gamma - rok1 & Gamma - rok2 & Gamma - rok3 \end{bmatrix}, \text{ gdzie rok3} = \text{rok2}+1 = \text{rok1}+2$$

strategia „najpierw ostatnie lata, kolejno wyroby np.: Alfa, Beta, Gamma”

$$\begin{bmatrix} - & - & - \\ - & - & - \\ - & - & - \end{bmatrix} \rightarrow \begin{bmatrix} - & - & + \\ - & - & - \\ - & - & - \end{bmatrix} \rightarrow \begin{bmatrix} - & - & + \\ - & - & + \\ - & - & - \end{bmatrix} \rightarrow \begin{bmatrix} - & - & + \\ - & - & + \\ - & - & + \end{bmatrix} \rightarrow \dots \rightarrow \begin{bmatrix} - & + & + \\ - & + & + \\ - & + & + \end{bmatrix} \rightarrow \dots \rightarrow \begin{bmatrix} + & + & + \\ + & + & + \\ + & + & + \end{bmatrix}$$

strategia „najpierw wyrób np. Alfa, rozpoczynając od ostatnich lat”

$$\begin{bmatrix} - & - & - \\ - & - & - \\ - & - & - \end{bmatrix} \rightarrow \begin{bmatrix} - & - & + \\ - & - & - \\ - & - & - \end{bmatrix} \rightarrow \begin{bmatrix} - & + & + \\ - & - & - \\ - & - & - \end{bmatrix} \rightarrow \begin{bmatrix} + & + & + \\ - & - & - \\ - & - & - \end{bmatrix} \rightarrow \dots \rightarrow \begin{bmatrix} + & + & + \\ + & + & + \\ - & - & - \end{bmatrix} \rightarrow \dots \rightarrow \begin{bmatrix} + & + & + \\ + & + & + \\ + & + & + \end{bmatrix}$$

strategia „najpierw pierwsze lata, kolejno wyroby np.: Alfa, Beta, Gamma”

$$\begin{bmatrix} - & - & - \\ - & - & - \\ - & - & - \end{bmatrix} \rightarrow \begin{bmatrix} + & - & - \\ - & - & - \\ - & - & - \end{bmatrix} \rightarrow \begin{bmatrix} + & - & - \\ + & - & - \\ - & - & - \end{bmatrix} \rightarrow \begin{bmatrix} + & - & - \\ + & - & - \\ + & - & - \end{bmatrix} \rightarrow \dots \rightarrow \begin{bmatrix} + & + & - \\ + & + & - \\ + & + & - \end{bmatrix} \rightarrow \dots \rightarrow \begin{bmatrix} + & + & + \\ + & + & + \\ + & + & + \end{bmatrix}$$

strategia „najpierw wyrób np. Alfa, rozpoczynając od ostatnich lat”

$$\begin{bmatrix} - & - & - \\ - & - & - \\ - & - & - \end{bmatrix} \rightarrow \begin{bmatrix} + & - & - \\ - & - & - \\ - & - & - \end{bmatrix} \rightarrow \begin{bmatrix} + & + & - \\ - & - & - \\ - & - & - \end{bmatrix} \rightarrow \begin{bmatrix} + & + & + \\ - & - & - \\ - & - & - \end{bmatrix} \rightarrow \dots \rightarrow \begin{bmatrix} + & + & + \\ + & + & + \\ - & - & - \end{bmatrix} \rightarrow \dots \rightarrow \begin{bmatrix} + & + & + \\ + & + & + \\ + & + & + \end{bmatrix}$$

c) w przypadku **niezadowolających wyników** uzyskany w wariantach a) i b) **zaproponować** można postępowanie oparte na **następujących strategiach**:

1. **Rozszerzenie obszaru modyfikowanych zmiennych decyzyjnych** i optymalizacja zamówień produkcyjnych dla wyrobów lub lat w których nie obserwowano poprawy wskaźników ekonomicznych. Celem może być zwiększenie zyskowności przedsięwzięć dzięki dostosowaniu wielkości produkcji oraz zmniejszeniu nadmiernych zapasów lub likwidacji wąskich gardeł w sprzedaży. Następnie ewentualnie ponowna optymalizacja cen.

2. **Rozszerzenie obszaru modyfikowanych zmiennych decyzyjnych** i optymalizacja zamówień surowców produkcyjnych mających decydujące znaczenie w produkcji wyrobów w których nie obserwowano poprawy wskaźników ekonomicznych. Celem może być zmniejszenie nadmiernych zapasów lub likwidacji wąskich gardeł w procesie produkcji (gdy surowców jest zbyt mało). Następnie ewentualnie ponowna optymalizacja cen.

3. Prowadzenie w **pierwszej kolejności optymalizacji cen wyrobów** dla których obserwuje się **problemy z zaopatrzeniem w surowce** lub **niezadowolające** (zbyt duże lub zbyt małe) **wielkości zapasów**.

4. **Radykalne zmiany warunków eksperymentu** w celu przesunięcia przeszukiwanego

hipersześcianu zmiennych decyzyjnych. Kierunek i natężenie zmian (np. procesów zaopatrzeniowych, podziału zysku, rozpoczęcia i sposobu finansowania inwestycji) można ustalić **po przeprowadzeniu szeregu eksperymentów.** Eksperymenty te polegają na **interaktywnej modyfikacji wartości odpowiednich zmiennych** za pośrednictwem **graficznego interfejsu użytkownika.** W związku z tym wyposażono go w takie aktywne elementy ekranowe jak listy wyboru (zmiennej decyzyjnej), potencjometry (suwaki) za pomocą których dokonuje się skokowej lub ciągłej modyfikacji zmiennych. Równocześnie z modyfikowaniem wartości zmiennych prezentowana jest animacja graficzna obserwowanego wskaźnika i – jeśli dla danego wskaźnika zdefiniowano zestaw reguł eksperckich – wnioskowanie. Proces wnioskowania bazowanego na regułach wykorzystuje (opisaną w innym miejscu) **architekturę tablicową systemu eksperckiego** oraz **parametryzację reguł.** **Architektura tablicowa** polega na konstruowaniu bazy wiedzy na bieżąco, w zależności od zainteresowań użytkownika natomiast parametryzacja reguł polega na modyfikacji diagnostycznego działania reguł: zdefiniowanych w plikach wskaźników wartości progowych (granicznych) lub wprowadzanych w trakcie prowadzenia eksperymentów (formularzy wywoływanych za pomocą aktywnych klawiszy ekranowych). Omawiane w bieżącym punkcie postępowanie jest **szczególnie zalecane** w przypadku optymalizacji ścieżek decyzyjnych o znacznej długości, przedstawiających **wielosekwencyjny i wieloletni horyzont przedsięwzięcia.**

Dla potrzeb badań zakładamy, że sposób wyboru optymalizowanych w następnym kroku elementarnych struktur jest **zdeteminowany**, tzn. postępuje się zgodnie z ustanowioną strategią (kolejnością) na którą nie mają wpływu wyniki eksperymentów.

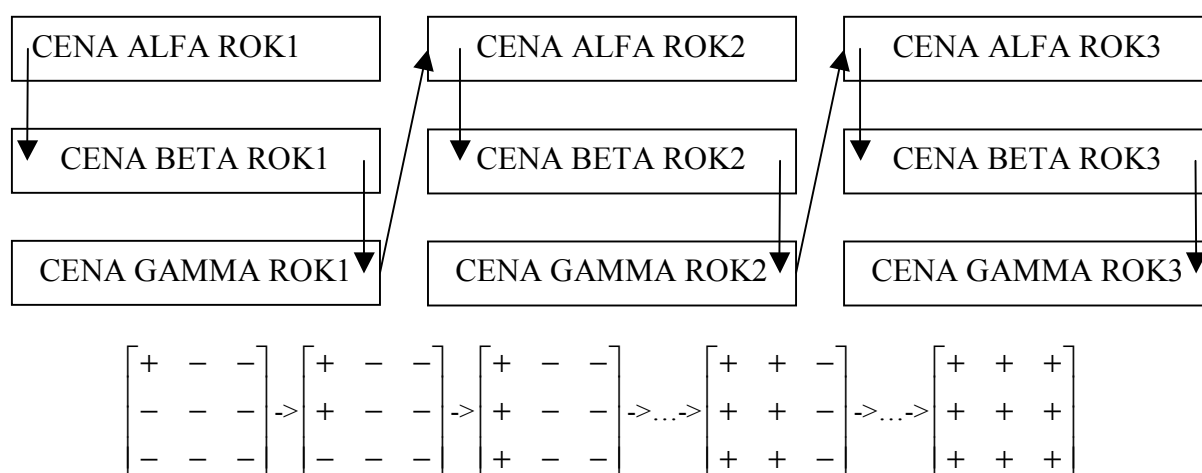
Opis przebiegu eksperymentu

Zadanie polega na takim doborze cen wyrobów gotowych aby zysk netto był jak największy. Optymalizacja dotyczy zysku netto na koniec ostatniego roku trzyletniej sekwencji decyzyjnej. Informacja o wartości zysku netto pochodzi z bilansu. Wartość zysku netto odczytuje się np. za pomocą instrukcji "odczyt bilans 5,2,3,wartosc_lewego" gdzie "5" oznacza numer wiersza sprawozdania, "2" jest liczbą początkową strony bilansu ("1" - aktywa, "2" - pasywa), "3" numerem okresu (roku) w trzyletniej sekwencji natomiast "wartosc_lewego" jest identyfikatorem (nazwą) uprzednio zadeklarowanej zmiennej, której przypisana zostanie wartość zysku netto.

Rozwiązanie problemu doboru optymalnej ceny (przy założeniu o pojedynczym maksimum funkcji celu) wymaga rozwiązania następujących problemów cząstkowych:

1. Wybór punktu startowego (początkowej wartości ceny zbytu wyrobu). Można wyróżnić następujące alternatywy: wartość ceny ustalona w programie - za pomocą instrukcji przypisania wartości albo przypisywana na podstawie zawartości (parametrów zmiennych decyzyjnych) aktywnego węzła decyzyjnego

- Wybór konfiguracji (strategii określanej kolejnością) optymalizowanych lat i wyrobów [Rys. 7-3]. W przypadku konfiguracji lat można wyróżnić postępowanie zgodne z następstwem czasowym (kolejno optymalizacji podlegają pierwszy, drugi i trzeci rok) lub przeciwstawne następstwu czasowemu (kolejno optymalizacji podlegają trzeci, drugi i pierwszy rok). W przypadku konfiguracji wyrobów można wyróżnić postępowanie w którym nadrzędną rolę przypisano wyrobom (przykładem może być konfiguracja: Alfa-1, Alfa-2, Alfa-3, Beta-1, Beta-2, Beta-3 itd.) (A1A2A3B1B2B3 itd.) oraz z nadrzędną rolą lat (przykładem może być konfiguracja: Alfa-1, Beta-1, Gamma-1, Alfa-2, Beta-2, Gamma-2 itd.)
- Dobór zakresu (dziedziny) wartości zmiennych decyzyjnych. Można zaproponować wyróżnienie trzech przypadków: dziedziny ustalonej odgórnie (za pomocą instrukcji przypisania wartości), dziedziny wartości zmiennych spełniających wzajemnie określone relacje oraz dziedziny zależnej od parametrów punktu startowego.
- Dobór warunku zatrzymania algorytmu. Wyróżnić tu można przede wszystkim kryterium liczby iteracji oraz zadanej dokładności przybliżenia (obliczeń) rozwiązania.



Rys. 7-3 Strategia „najpierw pierwsze lata, kolejno wyroby np.: Alfa, Beta, Gamma”. Dwie techniki reprezentacji graficznej

Poniżej przedstawiono tekst źródłowy planu eksperymentu w języku LEKS, dotyczącego metody bisekcji.

```
clear
raport
run
```

```
deklaracja wezel_poczatkowy
REM cd deklaracji
```

```
przypisz epsilon,0.00005
```

przypisz max_licznik,5

REM uruchamianie w trybie ciągłym
REM przygotowanie eksperymentu
REM aktualnie aktywny węzeł
REM staje się szablonem działań optymalizacyjnych
REM czyli jest optymalizowany
REM zmiany dotyczą ceny poszczególnych wyrobów
REM w odpowiednich latach
REM optymalizujemy zysk netto

REM wezel_początkowy przypisany zostanie
REM numer aktywnego (zaznaczonego) węzła
odczyt węzeł wezel_początkowy

REM zmiennej wezel_ojciec przypisany zostanie
REM numer rodzica węzła o numerze wezel_początkowy
odczyt ojciec wezel_początkowy, wezel_ojciec

REM *****

REM zmiennej wezel_potomek przypisany zostanie
REM numer nowego węzła dodanego,
REM którego rodzicem jest wezel_ojciec
REM a wzorcem: wezel_początkowy (czyli bliźniaczego względem wezel_początkowy)
dodaj węzeł wezel_ojciec,wezel_początkowy,wezel_potomek
REM węzłem aktywnym zostanie węzeł
REM o numerze zawartym w zmiennej wezel_potomek
ustaw węzeł wezel_potomek

REM -----

REM A1B1G1 A2B2G2

REM optymalizujemy zysk netto w ostatnim roku sekwencji
REM zmieniając w poszczególnych latach (1->3)
REM kolejno ceny wyrobów (alfa, beta, gamma)
REM inaczej:
REM -----
REM kolejność działań optymalizacyjnych
REM czasowa: rok1 rok2 rok3 (- - - - >)
REM przedmiotowa: ceny wyrobu alfa, ceny wyrobu beta, ceny wyrobu gamma
REM zaczynamy od ceny wyrobu alfa w roku rok1
REM następnie cena wyrobu beta w roku rok1 itd.

call alfarok1ceny.roz pocz,200,kon,290
call betarok1ceny.roz pocz,320,kon,400
call gammarok1ceny.roz pocz,380,kon,490
call alfarok2ceny.roz pocz,200,kon,290
call betarok2ceny.roz pocz,320,kon,400
call gammarok2ceny.roz pocz,380,kon,490
call alfarok3ceny.roz pocz,200,kon,290
call betarok3ceny.roz pocz,320,kon,400
call gammarok3ceny.roz pocz,380,kon,490

call notowanie.roz

REM -----

dodaj węzeł wezel_ojciec,wezel_początkowy,wezel_potomek
ustaw węzeł wezel_potomek

REM -----

REM A1G1B1 A2G2B2

REM optymalizujemy zysk netto w ostatnim roku sekwencji
REM zmieniając w poszczególnych latach (1->3)
REM kolejno ceny wyrobów (alfa, gamma, beta)
REM inaczej:
REM -----
REM kolejność działań optymalizacyjnych

REM czasowa: rok1 rok2 rok3 (----->)
REM przedmiotowa: ceny wyrobu alfa, ceny wyrobu gamma, ceny wyrobu beta
REM zaczynamy od ceny wyrobu alfa w roku rok1
REM następnie cena wyrobu gamma w roku rok1 itd.

call alfarok1ceny.roz pocz,200,kon,290
call gammarok1ceny.roz pocz,380,kon,490
call betarok1ceny.roz pocz,320,kon,400
call alfarok2ceny.roz pocz,200,kon,290
call gammarok2ceny.roz pocz,380,kon,490
call betarok2ceny.roz pocz,320,kon,400
call alfarok3ceny.roz pocz,200,kon,290
call gammarok3ceny.roz pocz,380,kon,490
call betarok3ceny.roz pocz,320,kon,400

call notowanie.roz

REM -----
dodaj węzeł wezel_ojciec,wezel_poczatkowy,wezel_potomek
ustaw węzeł wezel_potomek
REM -----
REM B1A1G1 B2A2G2
REM optymalizujemy zysk netto w ostatnim roku sekwencji
REM zmieniając w poszczególnych latach (1->3)
REM kolejno ceny wyrobów (beta, alfa, gamma)
REM inaczej:
REM -----
REM kolejność działań optymalizacyjnych
REM czasowa: rok1 rok2 rok3 (----->)
REM przedmiotowa: ceny wyrobu beta, ceny wyrobu alfa, ceny wyrobu gamma
REM zaczynamy od ceny wyrobu beta w roku rok1
REM następnie cena wyrobu alfa w roku rok1 itd.

call betarok1ceny.roz pocz,320,kon,400
call alfarok1ceny.roz pocz,200,kon,290
call gammarok1ceny.roz pocz,380,kon,490
call betarok2ceny.roz pocz,320,kon,400
call alfarok2ceny.roz pocz,200,kon,290
call gammarok2ceny.roz pocz,380,kon,490
call betarok3ceny.roz pocz,320,kon,400
call alfarok3ceny.roz pocz,200,kon,290
call gammarok3ceny.roz pocz,380,kon,490

call notowanie.roz

REM -----
dodaj węzeł wezel_ojciec,wezel_poczatkowy,wezel_potomek
ustaw węzeł wezel_potomek
REM -----
REM B1G1A1 B2G2A2
REM optymalizujemy zysk netto w ostatnim roku sekwencji
REM zmieniając w poszczególnych latach (1->3)
REM kolejno ceny wyrobów (beta, gamma, alfa)
REM inaczej:
REM -----
REM kolejność działań optymalizacyjnych
REM czasowa: rok1 rok2 rok3 (----->)
REM przedmiotowa: ceny wyrobu beta, ceny wyrobu gamma, ceny wyrobu alfa
REM zaczynamy od ceny wyrobu beta w roku rok1
REM następnie cena wyrobu gamma w roku rok1 itd.

call betarok1ceny.roz pocz,320,kon,400
call gammarok1ceny.roz pocz,380,kon,490
call alfarok1ceny.roz pocz,200,kon,290
call betarok2ceny.roz pocz,320,kon,400

```
call gammarok2ceny.roz pocz,380,kon,490
call alfarok2ceny.roz pocz,200,kon,290
call betarok3ceny.roz pocz,320,kon,400
call gammarok3ceny.roz pocz,380,kon,490
call alfarok3ceny.roz pocz,200,kon,290
```

call notowanie.roz

```
REM -----
dodaj węzeł wezeł_ojciec,wezeł_początkowy,wezeł_potomek
ustaw węzeł wezeł_potomek
REM -----
REM G1A1B1 G2A2B2
REM optymalizujemy zysk netto w ostatnim roku sekwencji
REM zmieniając w poszczególnych latach (1->3)
REM kolejno ceny wyrobów (gamma, alfa, beta)
REM inaczej:
REM -----
REM kolejność działań optymalizacyjnych
REM czasowa: rok1 rok2 rok3 (- - - - >)
REM przedmiotowa: ceny wyrobu gamma, ceny wyrobu alfa, ceny wyrobu beta
REM zaczynamy od ceny wyrobu gamma w roku rok1
REM następnie cena wyrobu alfa w roku rok1 itd.
```

```
call gammarok1ceny.roz pocz,380,kon,490
call alfarok1ceny.roz pocz,200,kon,290
call betarok1ceny.roz pocz,320,kon,400
call gammarok2ceny.roz pocz,380,kon,490
call alfarok2ceny.roz pocz,200,kon,290
call betarok2ceny.roz pocz,320,kon,400
call gammarok3ceny.roz pocz,380,kon,490
call alfarok3ceny.roz pocz,200,kon,290
call betarok3ceny.roz pocz,320,kon,400
```

call notowanie.roz

REM itd.

stop

W powyższym programie poszukiwania rozwiązania optymalnego zawężone zostały do następujących przedziałów:

$200 < \text{Cena Alfa} < 290$

$320 < \text{Cena Beta} < 400$

$380 < \text{Cena Gamma} < 490$

i w takim też zakresie rozwiązania będą poszukiwane w ramach podprocedur.

Procedura rozwiązywania problemu elementarnego (dobór optymalnej ceny wyrobu alfa w drugim roku trzyletniej sekwencji) jest następująca („cenyalfarok2.roz”):

```
run
REM *****
REM wg ceny alfa rok2 od 200 do 290
REM *****
```

```
dodaj węzeł wezeł_ojciec,wezeł_potomek,aktywny_wezeł_lewy
ustaw węzeł aktywny_wezeł_lewy
dodaj węzeł wezeł_ojciec,wezeł_potomek,aktywny_wezeł_prawy
ustaw węzeł aktywny_wezeł_prawy
```

```
przypisz lewy_kraniec ,pocz
przypisz prawy_kraniec ,kon
```

przypisz licznik ,0
REM licznik

while > max_licznik,licznik

ustaw węzeł aktywny_wezel_lewy
zmiana cena alfa rok2 lewy_kraniec
symuluj węzeł aktywny_wezel_lewy
odczyt bilans 5,2,3,wartosc_lewego

ustaw węzeł aktywny_wezel_prawy
zmiana cena alfa rok2 prawy_kraniec
symuluj węzeł aktywny_wezel_prawy
odczyt bilans 5,2,3,wartosc_prawego

if > wartosc_lewego,wartosc_prawego
rem message dla lewego wiekszy
- dodatnia_roznica, wartosc_lewego, wartosc_prawego
+ optymalny_argument, lewy_kraniec, 0

+ nowy_kraniec, prawy_kraniec, lewy_kraniec
/ prawy_kraniec, nowy_kraniec, 2
kasuj węzeł aktywny_wezel_prawy
dodaj węzeł wezel_ojciec,wezel_potomek,aktywny_wezel_prawy
kasuj węzeł aktywny_wezel_lewy
dodaj węzeł wezel_ojciec,wezel_potomek,aktywny_wezel_lewy
endif

if >= wartosc_prawego,wartosc_lewego
REM message dla prawego wiekszy
- dodatnia_roznica, wartosc_prawego, wartosc_lewego
+ optymalny_argument, prawy_kraniec, 0
+ nowy_kraniec, prawy_kraniec, lewy_kraniec
/ lewy_kraniec, nowy_kraniec, 2
kasuj węzeł aktywny_wezel_lewy
dodaj węzeł wezel_ojciec,wezel_potomek,aktywny_wezel_lewy
kasuj węzeł aktywny_wezel_prawy
dodaj węzeł wezel_ojciec,wezel_potomek,aktywny_wezel_prawy
endif

+ licznik, licznik, 1

if > epsilon,dodatnia_roznica
rem pokaz licznik
przypisz licznik, max_licznik
endif

endwhile

rem
rem licznik zawiera liczbe iteracji
rem

kasuj węzeł aktywny_wezel_lewy
kasuj węzeł aktywny_wezel_prawy
dodaj węzeł wezel_ojciec,wezel_potomek,aktywny_wezel_lewy
ustaw węzeł aktywny_wezel_lewy

rem optymalny_argument zawiera wartosc ceny alfa w roku drugim, dla ktorej wartosc zysku jest najwieksza

zmiana cena alfa rok2 optymalny_argument
symuluj węzeł aktywny_wezel_lewy
odczyt bilans 5,2,3,optymalna_wartosc

przypisz ost, aktywny_wezel_lewy
kasuj węzeł wezel_potomek
+ wezel_potomek, aktywny_wezel_lewy, 0

Procedura odpowiedzialna za raportowanie wartości zmiennych do pliku tekstowego jest następująca („notowanie.roz”):

```
run
zanotuj aktywny_wezel_lewy
zanotuj optymalna_wartosc
ustaw węzeł ost
odczyt zmienne decyzyjne 4, 1, optymalny_argument
zanotuj optymalny_argument
odczyt zmienne decyzyjne 4, 2, optymalny_argument
zanotuj optymalny_argument
odczyt zmienne decyzyjne 4, 3, optymalny_argument
zanotuj optymalny_argument
odczyt zmienne decyzyjne 5, 1, optymalny_argument
zanotuj optymalny_argument
odczyt zmienne decyzyjne 5, 2, optymalny_argument
zanotuj optymalny_argument
odczyt zmienne decyzyjne 5, 3, optymalny_argument
zanotuj optymalny_argument
odczyt zmienne decyzyjne 6, 1, optymalny_argument
zanotuj optymalny_argument
odczyt zmienne decyzyjne 6, 2, optymalny_argument
zanotuj optymalny_argument
odczyt zmienne decyzyjne 6, 3, optymalny_argument
zanotuj optymalny_argument
```

7.2.2 Metoda bisekcji – wyniki eksperymentów i wnioski

Wyniki eksperymentów przedstawiono w tabeli [Tab. 7-1] i na wykresach [Rys. 7-4], [Rys. 7-5].

Wskaźnik jakości w_j obliczany jest w następujący sposób: $w_j = \frac{\text{wynik finansowy}}{\text{wynik finansowy zestawu bazowego}}$

Tab. 7-1 Metoda bisekcji - wyniki pierwszej serii eksperymentów

Konfiguracja eksperymentu (strategia)	Optymalne ceny wyrobu ALFA [zł]	Optymalne ceny wyrobu BETA [zł]	Optymalne ceny wyrobu GAMMA [zł]	Wynik finansowy [mln zł]	Wskaźnik jakości w_j	Lokata
Zestaw bazowy				1,2	=1	
A1->A2->A3	A1 A2 A3 ceny			średnia	=15,27	<-średnia
A1A2A3 B1B2B3	273,1 245 245	360 390 375	462,5 490 435	16,9	=14,08	11
A1A2A3 G1G2G3	273,1 245 245	360 360 380	462,5 490 435,	20	=16,67	Wariant B
B1B2B3 A1A2A3	245 267,5 256,2	400 360 370	462,5 490 435,	17,1	=14,25	10
B1B2B3 G1G2G3	267,5 245 256,2	400 360 370	435 490 448,	18,3	=15,25	7
G1G2G3 A1A2A3	267,5 245 256,2	360 360 380	490 483,1 421,2	18,8	=15,67	Wariant C
G1G2G3 B1B2B3	267,5 245 256,2	360 360 380	490 483,1 421,2	18,8	=15,67	jw.
A1->B1->G1	A1 B1 G1 ceny				=15,2	<-średnia
A1B1G1 A2B2G2	273,1 245 250,6	360 400 375	462,5 490 435	15,8	=13,17	24
A1G1B1 A2G2B2	273,1 245 250,6	360 360 380	462,5 490 435	20,9	=17,42	Wariant A
B1A1G1 B2A2G2	245 273,1 250,6	400 360 375	435 490 448,8	17,3	=14,42	9
B1A1G1 B2A2G2	273,1 245 250,6	400 360 375	435 490 448,8	18	=15	8
G1A1B1 G2A2B2	273,1 245 250,6	360 360 380	490 483,1 435	18,7	=15,58	5-6
G1B1A1 G2B2A2	273,1 245,250,6	360 360 380	490 483,1 435	18,7	=15,58	5-6
A3->A2->A1	A3 A2 A1 ceny				=13,53	<-średnia
A3A1A1 B3B2B1	245 245 267,5	360 360 380	435 490 476	16,1	=13,42	17-23
A3A2A1 G3G2G1	245 245 267,5	360 380 380	490 435 476	16,2	=13,5	14-16
B3B2B1 A3A2A1	245 245 267,5	360 360 380	435 490 476	16,1	=13,42	17-23
B3B2B1 G3G2G1	245 267,5 267,5	360 360 380	490 380 476	16,6	=13,83	12-13
G3G2G1 A3A2A1	245 245 267,5	360 380 380	490 435 476	16,2	=13,5	14-16
G3G2G1 B3B2B1	245 245 267,5	360 390 380	490 435 476	16,2	=13,5	14-16
A3->B3->G3	A3 B3 G3 ceny				=13,49	<-średnia
A3B3G3 A2B2G2	245 245 267,5	360 360 380	490 380 476	16,1	=13,42	17-23
A3G3B3 A2G2B2	245 245 267,5	360 360 380	490 380 476	16,1	=13,42	17-23
B3A3G3 B2A2G2	245 245 267,5	360 360 380	490 380 476	16,1	=13,42	17-23
B3A3G3 B2A2G2	245 245 267,5	360 360 380	435 490 476	16,1	=13,42	17-23
G3A3B3 G2A2B2	284,4 222,5 267,5	360 360 380	435 490 476,3	16,6	=13,83	12-13
G3B3A3 G2B2A2	245 245 267,5	360 360 380	435 490 476,3	16,1	=13,42	17-23

Konfiguracja eksperymentu (grupa strategii)	Średni wskaźnik jakości	Odchylenie standardowe wskaźnika jakości
A1->A2->A3	15,27	0,973
A1->B1->G1	15,2	1,413
A3->A2->A1	13,53	0,153
A3->B3->G3	13,49	0,167

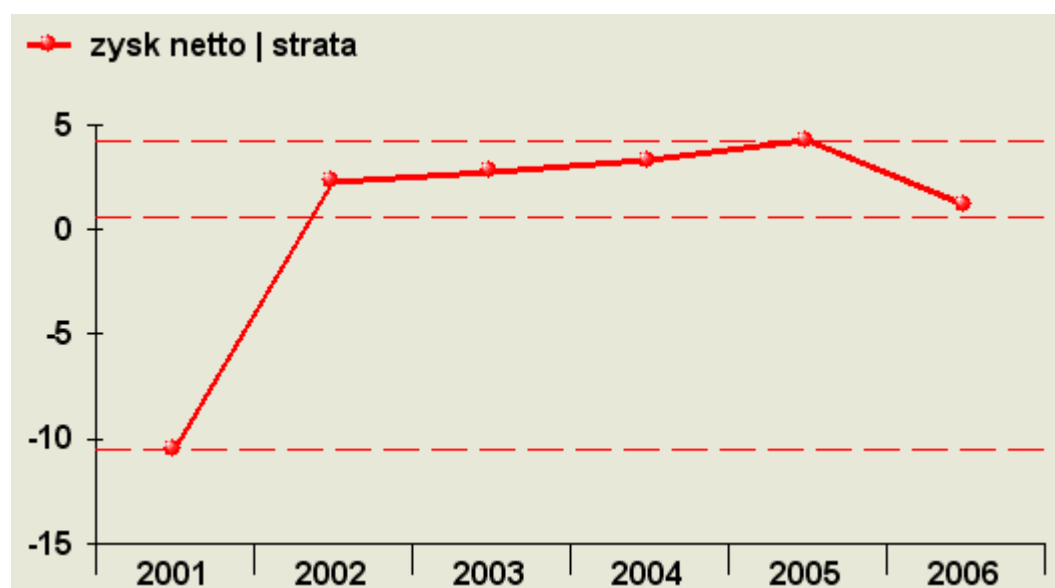
Jak widać z powyższego zestawienia (po wypełnieniu) najlepsze miejsce w rankingu zajęły kolejno:

1. strategia 'A1G1B1 A2G2B2 A3G3B3'

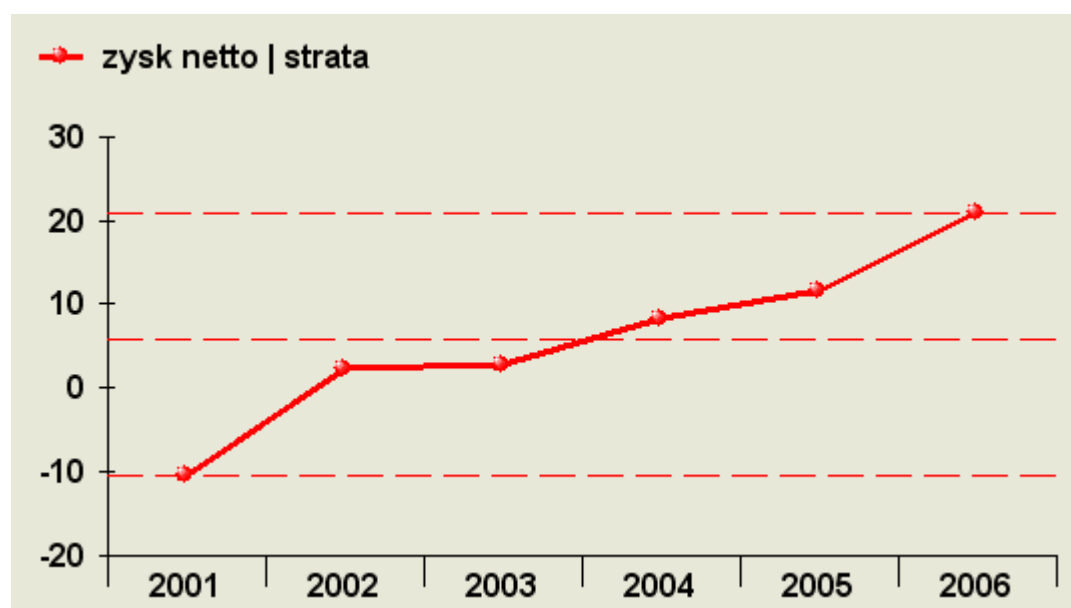
2. strategia 'A1A2A3 G1G2G3 B1B2B3'

3-4: strategia 'G1G2G3 A1A2A3 B1B2B3' i 'G1G2G3 B1B2B3 A1A2A3'.

Średnio najlepsze były strategie typu „A1-->A2-->A3” i obserwowano w ich przypadku odchylenie standardowe o wartości 0,973 którą można uważać za znacząco mniejsze od następnej w kolejności grupy strategii. Wszystkie rozwiązania były wg zastosowanego kryterium znacząco (kilkanaście razy) lepsze niż rozwiązanie bazowe. Co więcej, algorytm umożliwił osiągnięcie wyższej wartości zysku netto także w latach poprzedzających zakończenie eksperymentu [Rys. 7-4], [Rys. 7-5]. Świadczy to o zdolności algorytmu połowienia przedziałów do wyszukiwania korzystnych rozwiązań.



Rys. 7-4 Przebieg wartości wyniku finansowego netto dla parametrów początkowych (domyślnych)



Rys. 7-5 Przebieg wartości wyniku finansowego netto dla parametrów optymalnych

Cztery najlepsze strategie wykorzystujemy jako rozwiązania bazowe A, B, C, D (służące skonstruowaniu szablonu) dla dalszych poszukiwań suboptymalnych rozwiązań [Tab. 7-2]. Zmiany będą obecnie dotyczyły zamówień produkcyjnych. Zastosujemy strategię, która sprawdziła się już w trakcie poszukiwań optymalnych cen, tzn. najlepszą „średnio” strategię „y”, niezależnie od strategii zastosowanej do uzyskania danego rozwiązania. Zakres zmienności zamówień przyjęto jako 0, 120% wartości bazowych (pochodzących z sekwencji D/R 0).

Tab. 7-2 Metoda bisekcji - wyniki drugiej serii eksperymentów

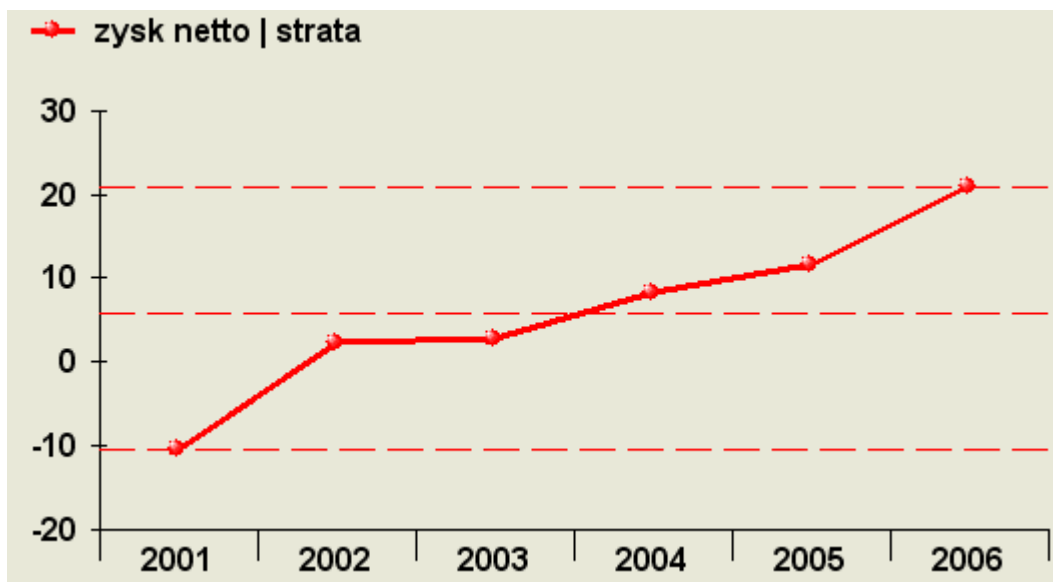
Konfiguracja eksperymentu (strategia)	Optymalne zamówienie produkcyjne Alfa [tys. j. wyr.]	Optymalne zamówienie produkcyjne Beta [tys. j. wyr.]	Optymalne zamówienie produkcyjne Gamma [tys. j. wyr.]	Wynik finansowy [mln zł]	Wskaźnik jakości wj	Lokata
Zestaw bazowy				1,2	=1	
Wariant A						
A1A2A3 B1B2B3	480 480 480	480 480 480	480 480 480	20,9	=17,42	1-6
A1A2A3 G1G2G3	480 480 480	480 480 480	480 480 480	20,9	=17,42	
B1B2B3 A1A2A3	480 480 480	480 480 480	480 480 480	20,9	=17,42	
B1B2B3 G1G2G3	480 480 480	480 480 480	480 480 480	20,9	=17,42	
G1G2G3 A1A2A3	480 480 480	480 480 480	480 480 480	20,9	=17,42	
G1G2G3 B1B2B3	480 480 480	480 480 480	480 480 480	20,9	=17,42	
Wariant B						
A1A2A3 B1B2B3	480 480 480	480 480 480	480 480 480	20	=16,67	7-12
A1A2A3 G1G2G3	480 480 480	480 480 480	480 480 480	20	=16,67	
B1B2B3 A1A2A3	480 480 480	480 480 480	480 480 480	20	=16,67	
B1B2B3 G1G2G3	480 480 480	480 480 480	480 480 480	20	=16,67	
G1G2G3 A1A2A3	480 480 480	480 480 480	480 480 480	20	=16,67	
G1G2G3 B1B2B3	480 480 480	480 480 480	480 480 480	20	=16,67	
Wariant C=D						
A1A2A3 B1B2B3	480 480 480	480 480 240	480 480 240	18,8	=15,67	13-18
A1A2A3 G1G2G3	480 480 480	480 480 240	480 480 240	18,8	=15,67	
B1B2B3 A1A2A3	480 480 480	480 480 240	480 480 240	18,8	=15,67	
B1B2B3 G1G2G3	480 480 480	480 480 240	480 480 240	18,8	=15,67	
G1G2G3 A1A2A3	480 480 480	480 480 240	480 480 240	18,8	=15,67	
G1G2G3 B1B2B3	480 480 480	480 480 240	480 480 240	18,8	=15,67	

Jak widać z powyższego zestawienia najlepsze miejsce w rankingu prowadzenie utrzymał Wariant A a kolejne miejsca zajęły Wariant B i Wariant C. Oczywiście jest, że wszystkie rozwiązania były wg zastosowanego kryterium lepsze niż rozwiązanie bazowe. Niestety algorytm nie pozwolił na osiągnięcie wyższej wartości zysku netto niż w odpowiednich rozwiązaniach bazowych. Jednak w Wariancie C wykryto możliwość osiągnięcia zysku netto na tym samym poziomie przy mniejszych zleceniach produkcyjnych wyrobu Gamma. Biorąc pod uwagę powyższe oraz poprzednie zestawienie można orzec, że Wariant A (bez zmian dotyczących zleceń produkcyjnych) jest jak na razie najkorzystniejszy.

Trzy najlepsze rozwiązania wykorzystujemy jako nowe rozwiązania bazowe A, B, C (służące skonstruowaniu szablonu) dla dalszych poszukiwań suboptymalnych rozwiązań. Zastosujemy [Tab. 7-3] ponownie najlepszy (średnio) typ strategii „y”. Zmiany będą obecnie dotyczyły zamówień surowców. Zakres zmienności zamówień przyjęto jako 0, 120% (czyli [0, 480 tys. j. wyr./rok]) wartości bazowych (pochodzących z sekwencji D/R 0).

Tab. 7-3 Metoda bisekcji - wyniki trzeciej serii eksperymentów

Konfiguracja eksperymentu (strategia)	Optymalne zamówienie surowca White [tys. j. sur.]	Optymalne zamówienie surowca Black [tys. j. sur.]	Optymalne zamówienie surowca Red [tys. j. sur.]	Wynik finansowy [mln zł]	Wskaźnik jakości wj	Lokata
Zestaw bazowy				1,2	=1	
Wariant A						
A1A2A3 B1B2B3	360 300 480	480 240 480	360 360 480	21	=17,5	1-6
A1A2A3 G1G2G3	360 300 480	480 480 480	360 360 480	21	=17,5	
B1B2B3 A1A2A3	360 360 480	270 360 480	360 360 480	21	=17,5	
B1B2B3 G1G2G3	360 360 480	270 360 480	360 360 480	21	=17,5	
G1G2G3 A1A2A3	360 360 480	480 240 480	360 360 480	21	=17,5	
G1G2G3 B1B2B3	360 360 480	360 360 480	360 360 480	21	=17,5	
Wariant B						
A1A2A3 B1B2B3	360 300 480	480 240 480	360 360 480	20,1	=16,75	13-18
A1A2A3 G1G2G3	360 300 480	480 480 480	360 360 480	20,1	=16,75	
B1B2B3 A1A2A3	360 360 480	270 360 480	360 360 480	20,1	=16,75	
B1B2B3 G1G2G3	360 360 480	270 360 480	360 360 480	20,1	=16,75	
G1G2G3 A1A2A3	360 360 480	480 240 480	360 360 480	20,1	=16,75	
G1G2G3 B1B2B3	360 360 480	360 360 480	360 360 480	20,1	=16,75	
Wariant C						
A1A2A3 B1B2B3	240 360 480	360 360 480	360 360 480	20,7	=17,25	9-12
A1A2A3 G1G2G3	240 360 480	480 480 480	360 360 480	20,7	=17,25	
B1B2B3 A1A2A3	300 360 480	240 240 240	360 360 480	20,7	=17,25	
B1B2B3 G1G2G3	360 360 480	240 240 240	360 360 480	20,7	=17,25	
G1G2G3 A1A2A3	360 360 480	480 480 480	300 360 30	20,8	=17,33	7-8
G1G2G3 B1B2B3	360 360 480	480 240 480	300 360 30	20,8	=17,33	



Rys. 7-6 Wykres wyniku finansowego netto w latach 2001-2006 dla najlepszego ze znalezionych rozwiązań

Stwierdzamy [Tab. 7-3] że najlepsze miejsce w rankingu zajęły kolejno: Wariant A (jak zwykle) następnie dwie odmiany (G1G2G3 A1A2A3, G1G2G3 B1B2B3) Wariantu C, potem pozostałe odmiany Wariantu C a końcowe miejsce zajął Wariant B. Oczywiście jest, że wszystkie rozwiązania były wg zastosowanego kryterium lepsze niż rozwiązanie bazowe. W niektórych przypadkach udało się poprawić wyniki w porównaniu z zadaniem optymalizacji cen. Co więcej, algorytm umożliwił osiągnięcie wyższej wartości zysku netto niż w odpowiednich rozwiązaniach początkowych także w latach poprzedzających zakończenie eksperymentu [Rys. 7-6]. Ponownie świadczy to o zdolności algorytmu połowienia przedziałów do wyszukiwania korzystnych rozwiązań.

Podsumowanie. Metoda połowienia przedziałów niekiedy okazywała się zawodna w rozbudowanych przypadkach. Jednak nigdy nie spowodowała pogorszenia wyników. Słabości metody wynikają m.in. z konieczności umiejętnego dobrania granic badanego przedziału, bezkontekstowości (brak preferowania rozwiązań angażujących mniejsze środki).

Na podstawie szeregu eksperymentów (nie zamieszczonych w niniejszej pracy) można stwierdzić, że **mechaniczne (automatyczne) postępowanie optymalizacyjne nie nadaje się zbytnio do optymalizacji dłuższych ścieżek decyzyjnych** w drzewie eksperymentów (czyli wielokrotnych sekwencji decyzyjnych). **Powody są następujące:** praktycznie nie da się stosować strategii „wstecznej” (optymalizacja najpierw końcowych lat) oraz istnieje **niebezpieczeństwo utknięcia w lokalnych optimach**, z których nie ma możliwości wydostania się **bez wykonania jakościowego skoku**. W drugim przypadku **niezbędne byłoby losowe generowanie zmian parametrów** (zmiennych decyzyjnych), analogiczne do rozwiązań stosowanych w algorytmach genetycznych. **Przeciwko** mechanicznemu przeniesieniu technik właściwych dla prognozowania krótkookresowego świadczą także wskazówki metodologiczne podawane np. przez Heizera i Rendera [114] stanowiące że **projektując decyzje i prognozując następstwa długoterminowych przedsięwzięć** należy wykorzystywać odmienne techniki. **Techniki ilościowe** lepiej sprawdzają się w rozwiązywaniu

problemów krótkoterminowych gdy tymczasem w zagadnieniach długoterminowych, wieloletnich warto rozważać wykorzystanie technik jakościowych lub „w mniejszym stopniu” ilościowych. W przypadku zagadnień długoterminowych nie stoi na przeszkodzie, **aby zastosować dekompozycje zadania** na krótsze okresy, pamiętać jednak należy o ocenie (diagnozie) efektów decyzji na koniec każdego okresu i ewentualnej rewizji prognoz i projektów dotyczących dalszego postępowania. Dlatego też postuluje się w tym miejscu skorzystanie z możliwości interaktywnego graficznego interfejsu użytkownika obrazującego wykresy zmiennej wyjściowej (np. zysku netto). Jak już wspomniano wcześniej, zastosowano w nim **sprzężenie aktywnych elementów ekranowych** (takich jak **listy wyboru i suwaki**) z **badanymi zmiennymi wejściowymi** (elementami macierzy zmiennych decyzyjnych) oraz **wyjścia hybrydowego SWD** w postaci **wykresów zmiennej zależnej**, parametryzowanych reguł eksperckich, informacji pochodzących ze sprawozdań finansowych.

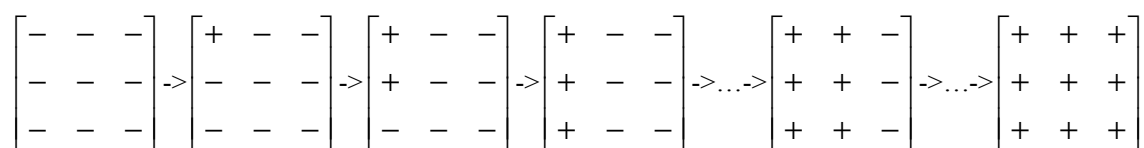
7.2.3 Wsparcie metody bisekcji interaktywnym interfejsem użytkownika

Poniżej przedstawiono **porównanie efektów dwu podejść** w przypadku dziewięcioletniego okresu optymalizacji (dwu następujących po sobie sekwencji decyzyjnych):

1. **W pełni automatycznej** optymalizacji podmacierzy zmiennych decyzyjnych (cen wyrobów)
2. **Automatycznej modyfikacji** (optymalizacji) połączonej z **wykorzystaniem interaktywnego interfejsu** użytkownika, za pomocą którego dokonywano zasadniczych zmian w zakresie zamówień produkcyjnych i zaopatrzenia (zamówień surowców).

Drugie podejście polega na **szybkiej identyfikacji** tych **zmiennych decyzyjnych**, których (już systematyczne, wykonywane automatycznie zgodnie z wybraną techniką optymalizacji) modyfikacje wartości (strojenie) **mogą zapewnić dalszą poprawę** wyników finansowych przedsiębiorstwa.

1. Zastosowano **strategię** typu „najpierw pierwsze lata, kolejno wyroby (we wszystkich sekwencjach tej ścieżki)” o reprezentacji kodowej (A1B1G1 A2B2G2 ...) i **reprezentacji graficznej**:



Tab. 7-4 Wyniki i wnioski dot. badanej strategii (A1B1G1 A2B2G2 ...)

Lata	Uwagi	Wykres zysku netto
------	-------	--------------------

2000-2002	Automatyczna optymalizacja przynosi dobre wyniki	
2003-2005	Automatyczna optymalizacja przynosi dobre wyniki	
2006-2008	Automatyczna optymalizacja nie sprawdziła się. Coraz większe straty wynikające z bezustannie rosnących zapasów. Właściwe ceny to zbyt mało.	

2. Zastosowano zmodyfikowaną **strategię** typu „najpierw pierwsze lata, kolejno wyroby np.: Alfa, Beta, Gamma”, jednak na koniec każdej z sekwencji dokonywano rewizji takich wielkości i wskaźników jak: zapasy surowców i wyrobów gotowych łącznie i w podziale na poszczególne wyroby, wielkość sprzedaży poszczególnych wyrobów, wskaźniki obrotowości. Następnie za pomocą interfejsu użytkownika (suwaków i pól wyboru elementów macierzy) **dokonywano odpowiednich modyfikacji w obszarze macierzy poza cenami** obserwując wynikowy wykres [Tab. 7-5] optymalizowanej zmiennej wyjściowej oraz odpowiedzi systemu eksperckiego.

Tab. 7-5 Wyniki i wnioski dot. badanej zmodyfikowanej strategii

Lata	Uwagi	Wykres
2000-2002	Automatyczna optymalizacja przynosi dobre wyniki	

2003-2005	Biorąc pod uwagę problemy opisane powyżej, zmieniono dodatkowo (zmniejszono) zamówienia surowców. Efektem jest utrzymany trend rosnący zysku	<table border="1"> <thead> <tr> <th>Year</th> <th>Net Profit</th> </tr> </thead> <tbody> <tr> <td>2003</td> <td>10</td> </tr> <tr> <td>2004</td> <td>14</td> </tr> <tr> <td>2005</td> <td>18</td> </tr> </tbody> </table>	Year	Net Profit	2003	10	2004	14	2005	18
Year	Net Profit									
2003	10									
2004	14									
2005	18									
2006-2008	Zastosowano automatyczną optymalizację. Niestety pojawił się trend spadkowy	<table border="1"> <thead> <tr> <th>Year</th> <th>Net Profit</th> </tr> </thead> <tbody> <tr> <td>2006</td> <td>12</td> </tr> <tr> <td>2007</td> <td>10</td> </tr> <tr> <td>2008</td> <td>-2</td> </tr> </tbody> </table>	Year	Net Profit	2006	12	2007	10	2008	-2
Year	Net Profit									
2006	12									
2007	10									
2008	-2									
2006-2008	Zastosowano ponownie modyfikację zamówień surowców. Uzyskano stosunkowo duże i stabilne wartości zysku netto	<table border="1"> <thead> <tr> <th>Year</th> <th>Net Profit</th> </tr> </thead> <tbody> <tr> <td>2006</td> <td>12</td> </tr> <tr> <td>2007</td> <td>13</td> </tr> <tr> <td>2008</td> <td>12</td> </tr> </tbody> </table>	Year	Net Profit	2006	12	2007	13	2008	12
Year	Net Profit									
2006	12									
2007	13									
2008	12									

Jak widać z powyższego przykładu, zadowalające wyniki osiągnięto korzystając zarówno z automatycznej optymalizacji ustalonego fragmentu macierzy zmiennych decyzyjnych (cen wyrobów), jak i zmiany, w razie potrzeby, wartości pozostałych komórek macierzy. Wykorzystano w tym celu informacje zawarte w sprawozdaniach (raporty produkcyjne), diagnozy pochodzące z parametryzowanych reguł systemu eksperckiego oraz (przede wszystkim) możliwości interfejsu użytkownika – suwaki i pola wyboru sprzęgające dane liczbowe (zmiennie decyzyjne), model symulacyjny, system ekspercki oraz wykresy badanej wielkości ekonomicznej.

**8 Zastosowanie języka sterowania
eksperymentem LEKS.
Analiza czynnikowa**

8.1 Rola eksperymentu w badaniach naukowych

Metoda naukowa badania obiektu (hipotetyczno-dedukcyjna) składa się z co najmniej pięciu elementów: obserwacji, hipotezy/prognozy, eksperymentu, wniosków, powtarzalności [324]. Poniżej przedstawiono opis składników metody naukowej w środowisku symulacyjnego komputerowego modelu przedsiębiorstwa.

Obserwacje realizowane są dzięki **rejestrowaniu** (zazwyczaj w funkcji czasu) wartości poszczególnych zmiennych decyzyjnych, pozycji sprawozdań finansowych i wskaźników ekonomicznych wyrażanych w odpowiednich jednostkach (pieniężnych, naturalnych, procentowych).

Hipotezy i prognozy służą wyjaśnieniu rejestrowanych obserwacji. Wyjaśnienie tego co działo się w przeszłości będzie pomocne podczas przewidywania oraz kształtowania przyszłości. Konstruowane hipotezy dotyczą identyfikacji (opracowania opisu matematycznego) badanego obiektu i optymalizacji (poprawienia) jego działania.

Eksperyment służy badaniu prawdziwości hipotez; wg Mańczaka [185] **eksperyment** (jako pojęcie złożone) jest **serią doświadczeń** „umożliwiająca identyfikację lub optymalizację rozważanego obiektu”. Właściwy **opis eksperymentu** podnosi jego wartość ze względu na możliwość jego (wielokrotnego) powtarzania oraz zmniejszenia wpływu wszelkich czynników, które mogłyby zakłócić eksperyment. Z drugiej strony koszty związane z prowadzeniem eksperymentu wymuszają aby serie doświadczeń były „możliwie krótkie” [185]. Pojawia się zatem **problem planowania eksperymentu**.

Wyciąganie wniosków (inaczej: opracowywanie wyników doświadczeń) polega na konfrontowaniu hipotez/prognoz z otrzymanymi wynikami eksperymentu a następnie ich wartościowaniu. Zarówno pozytywny, jak i negatywny wynik konfrontacji powinien być odpowiednio wyjaśniony. Rozważyć należy czy na wynik miały wpływ **ewentualne błędy** w konstrukcji planu eksperymentu, sposób pozyskiwania, jak i charakter (stochastyczny, poddany zakłóceniom) obserwacji oraz metoda wnioskowania (w tym osoba eksperymentatora).

Ustalenie **powtarzalności wyników** eksperymentu ważne jest z dwu powodów. Po pierwsze, jeśli wynik eksperymentu **nie jest zgodny z przewidywaniami** (hipotezą/prognozą) to **rewizji** może wymagać **teoria** dostarczająca metod procesowi wnioskowania i wyjaśniająca obserwacje. Po drugie, powtarzalność (pozytywna) **zgodności wyników eksperymentu** z teorią umożliwia ich **prezentację** przed innymi eksperymentatorami i **decydentami**. Występują **dwa rodzaje** zgodności wyników z teorią: (1) dla **tych samych wartości** zmiennych wejściowych (parametrów eksperymentu) uzyskuje się **takie same** wartości zmiennych wyjściowych (wyników eksperymentu) oraz (2) dla tych samych wartości zmiennych wejściowych uzyskuje się wartości zmiennych wyjściowych **będących w określonym związku** (relacji) z wartościami zmiennych wejściowych. Zadaniem teorii jest wyjaśnienie tego związku; “pierwotnym celem badania naukowego jest zwykle pokazanie statystycznej istotności efektu działania określonego czynnika na badaną zmienną zależną” [282].

8.2 Planowanie doświadczeń. Systematyka planów eksperymentu

Pojęcie **planowania eksperymentu** zostało wprowadzone w pracy Fishera [86] w kontekście analizy wariancyjnej. W opozycji do analizy wariancji jednym z celów planowania eksperymentu jest **pomijanie czynników nieistotnych** z punktu widzenia **identyfikacji badanych czynników**. Cechą wspólną czynności planowania eksperymentu i analizy wariancyjnej jest dążenie do uzyskania opisu działania obiektu za pomocą zależności statystycznych czyli **analizy regresji**. Eksperyment prowadzony na rzeczywistym systemie lub jego modelu (symulacyjnym) pozwala na odkrycie które z czynników (zmiennych wejściowych) mają **wpływ na zmienne zależne** i w konsekwencji na ocenę działania systemu.

Efektem planowania eksperymentu jest **plan eksperymentu**. Plan eksperymentu jest **zbiorem wartości zmiennych wejściowych** (wejść) określającym parametry konkretnych doświadczeń (niekoniecznie prowadzonych w tym samym czasie). Zestaw doświadczeń tworzy **serię**. W poszczególnych doświadczeniach **zmieniać mogą się wartości zmiennych** natomiast **zbiór zmiennych** nie ulega zmianom. Ze względu na postulat powtarzalności **seria doświadczeń** o tych samych wartościach wejściowych ma zwykle długość większą niż jeden. **Pojedyncze doświadczenie** umożliwić może ustalenie parametrów funkcji regresji **jedynie przy pewności dotyczącej opisu obiektu**. Powtarzanie (realizacja serii) doświadczeń przy tych samych wartościach zmiennych wejściowych stosuje się w celu oszacowania parametrów opisu badanego obiektu przy założeniu o obecności czynników nieuwzględnionych w matematycznym opisie (modelu) obiektu.

Przed przystąpieniem do przeprowadzenia eksperymentu proponuje się [50] wykonanie następujących czynności:

- a) Podjęcia decyzji o **charakterze eksperymentu** (czynny albo bierny, ciągły albo dyskretny)
- b) Ustalenia **parametrów** eksperymentu (dokładność prognoz, okres kwantowania, czas trwania eksperymentu)
- c) Ogólnego, wstępnego prognozowania wyników które otrzymamy w przyszłości; nazywane jest to **likwidacją anomalii** [137] w wynikach eksperymentu i filtracją wyników eksperymentu
- d) **Planowania** eksperymentu (dla eksperymentu czynnego): **dobór macierzy** eksperymentu
- e) Podjęcia decyzji o **technice eksperymentu** (dobór przyrządów, układów pomiarowych, języka sterowania eksperymentem, interfejs między systemem lub modelem a systemem komputerowym sterującym zmiennymi wejściowymi i zbierającym wyniki).

Typ planu eksperymentu czynnego określany jest przez strukturę planu, złożoność modeli opracowanych dzięki zastosowaniu planu oraz proces jego konstruowania. **Typologię planów eksperymentu** zbudować można na podstawie następujących kryteriów: zbioru wartości zmiennych wejściowych, zastosowanej techniki rozbudowy planu, charakteru ograniczeń, liczby doświadczeń.

Według kryterium zbioru wartości zmiennych wejściowych wyróżniamy:

- a) **Plany dwupoziomowe**, oznaczane jako 2^S . Nadają się do wyznaczania współczynników modeli z elementami liniowymi oraz elementami interakcyjnymi (oprócz kwadratowych). Wykorzystywane są podczas optymalizacji. Ich cechą charakterystyczną jest dwupoziomowe kodowanie wartości zmiennych wejściowych. Wykorzystuje się w tym celu wartości -1,1. W celu ułatwienia obliczeń podczas analizy czynnikowej przeprowadza się zamianę zmiennych i standaryzację wartości
- b) **Plany trójpoziomowe**, oznaczane jako 3^S . Ich zaletą w porównaniu do planów dwupoziomowych jest **możliwość rozdzielenia wpływu członów kwadratowych od wpływu składowej stałej**. Lepiej sprawdzają się także w badaniu obszaru zawierającego ekstrema co jest przydatne podczas optymalizacji lokalnej. Podobnie jak plany dwupoziomowe korzystają z techniki standaryzacji zmiennych. W przypadku wyznaczania współczynników modeli liniowokwadratowych stosuje się dodatkowo planowanie kompozycyjne
- c) **Plany wielopoziomowe**. Szczególnie istotne jest planowanie pięciopoziomowe, będące **rozbudowanym planowaniem dwupoziomowym**. Nadają się one do opracowania modeli liniowokwadratowych w których liczba wejść przekracza $S=3$. Techniki rozbudowy obejmują planowanie kompozycyjne, ortogonalne i rotatabilne.

Według kryterium zastosowanej techniki rozbudowy planu wyróżniamy:

- a) **Planowanie kompozycyjne**; jego istotą jest rozbudowa planu dwu- lub trójpoziomowego (całkowitego lub ułamkowego) o tzw. **doświadczenia gwiazdne** (badana zmienna przyjmuje wartości symetryczne względem osi (zera) przy pozostałych zmiennych równych 0) **oraz centralne** (wszystkie zmienne równe zero). Cechą szczególną planowania kompozycyjnego jest wykorzystanie doświadczeń prowadzonych **podczas ustalania parametrów modelu stopnia I** do opracowania modelu stopnia II
- b) **Planowanie ortogonalne**; jego zaletą jest **uproszczenie obliczeń** podczas obliczania współczynników modelu regresji. O ortogonalności planu można mówić po diagonalizacji macierzy kowariancji $(X^T X)^{-1}$ poprzez wybór macierzy wejść X. **Ortogonalność** oznacza [178], że wszystkie iloczyny skalarne wektorów wierszowych macierzy eksperymentu są równe zero, tzn.
$$\sum_{n=1}^N x_n^i x_n^j = 0$$
dla $i \neq j$ gdzie $i, j = 1, 2, 3, \dots, (s+1)$. Ortogonalność zapewnia także diagonalność macierzy $(X^T X)^{-1}$ tzn. że jest ona macierzą diagonalną postaci $(1/N) I$ (I – macierz jednostkowa), co znacznie upraszcza obliczenia w analizie czynnikowej
- c) **Planowanie rotatabilne**; jego zaletą jest **stabilizacja dokładności wyznaczonego modelu** oraz analogiczna własność oszacowania funkcji regresji w otoczeniu punktu centralnego. Cechą macierzy wejść X dla planu rotatabilnego jest niezależność macierzy kowariancyjnej $(X^T X)^{-1}$ od obrotu współrzędnych. Wyznaczony w ten sposób model charakteryzuje się **wariancją zależną tylko od odległości od punktu centralnego** serii doświadczeń.

Według **kryterium charakteru ograniczeń** wyróżniamy:

a) **Ogólne plany czynnikowe**, w których zmienne wejściowe przyjmować mogą **dowolne wartości** a do wyznaczenia nieznannej charakterystyki statycznej obiektu wykorzystuje się analizę regresji. Nazwa „czynnikowe” pochodzi z określenia zbioru zmiennych wejściowych jako czynników doświadczenia a zbioru wartości zmiennych wejściowych jako przestrzeni czynnikowej [178].

b) **Plany sympleksowe** stosowane są gdy na wartości zmiennych narzucono określony typ ograniczeń [185]. Interpretacją powyższych ograniczeń są mieszaniny zależne od składu. Narzucone na zmienne ograniczenia określają tzw. sympleks na (S-1) wymiarowej hiperpłaszczyźnie. Celem planowania sympleksowego jest wyznaczenie współczynników wielomianów zredukowanych. Wśród planów sympleksowych wyróżnia się (podobnie jak dla innych planów) plany całkowite i ułamkowe.

Według **kryterium liczby doświadczeń** wyróżniamy:

a) **Plany całkowite** tzn. zawierające wszystkie możliwe kombinacje wartości zmiennych wejściowych. Plany całkowite **spełniają wiele szczególnych warunków**, np. jeśli jednocześnie są planami dwupoziomowymi, to zachodzi w nich symetria doświadczeń względem punktu centralnego (środką) eksperymentu, ortogonalność, równość sum kwadratów we wszystkich wierszach macierzy eksperymentu [185].

b) **Plany ułamkowe** (lub szerzej – niepełne); nie zawierają wszystkich możliwych kombinacji. Plany ułamkowe powinny spełniać wszystkie korzystne własności planów całkowitych. W tym celu jednak **konieczny jest staranny dobór** zarówno liczby doświadczeń (tzw. plany połówkowe, ćwiartkowe, itd.) jak i kombinacji wartości zmiennych wejściowych. Podczas tworzenia planu ułamkowego stosuje się kodowanie poszczególnych doświadczeń. **Koncepcję kodowania poziomów** poszczególnych czynników (zmiennych wejściowych) opracował Yates [331]. Polega ona na oznaczaniu kombinacji najniższych poziomów wartości wszystkich zmiennych liczbą 1, a pozostałych poziomów ciągiem liter (przydzielanych alfabetycznie) z tym, że dana litera występuje w takiej kombinacji jeżeli odpowiadająca jej zmienna przyjmuje wartość maksymalną. Przy tworzeniu odpowiednich planów niepełnych/ułamkowych bierze się pod uwagę wynikową dokładność aproksymacji (prowadzi to wg Kacprzyńskiego [137] do zagęszczenia punktów pomiarów przy brzegu badanego obszaru) ale zazwyczaj punkty pomiarowe umieszcza się w sposób jak najbardziej regularny w dziedzinie wartości wejściowych (zakodowanych). Prawidłowo przyjęte współrzędne punktów pomiarowych (wartości wejściowych) w hipersześcianie pozwalają na „wyznaczenie ocen bądź wszystkich efektów głównych działania czynników lub też efektów współdziałania czynników” [136]. Celem tworzenia **planów ułamkowych** jak potwierdza Mańczak [185] jest **zmniejszenie liczby wykonywanych doświadczeń**.

8.3 Planowanie dwupoziomowe – przykłady zastosowań

8.3.1 Założenia metody Boxa-Wilsons dla potrzeb dwupoziomowego planu eksperymentu

Jednym z problemów badawczych które można rozwiązać posługując się techniką planowania eksperymentu jest **poszukiwanie odpowiednich wartości zmiennych wejściowych** (decyzyjnych) w celu **optymalizacji** wskaźników syntetycznie oceniających efekty działania przedsiębiorstwa. Przykładem może być dobór cen zbytu wyrobów gotowych w celu maksymalizacji wyniku finansowego netto. W tym celu wykorzystano język sterowania eksperymentu LEKS zintegrowany z hybrydowym systemem wspomagania decyzji EKANWIN.

Technikę planowania dwupoziomowego, w której zastosowano metodę Boxa-Wilsons można przedstawić w następujący sposób [25]:

Założenia:

a) niech badany obiekt będzie opisany przez funkcję:

$y = f(x_1, x_2, x_3, \dots, x_S, z)$ - ciągłą, nieliniową, posiadającą jedno ekstremum, gdzie

$x_1, x_2, x_3, \dots, x_S$ są wejściami obiektu,

z jest niemierzalnym zakłóceniem

b) można wyznaczyć aproksymację funkcji opisującej obiekt w otoczeniu punktu

postaci $x_1^0, x_2^0, x_3^0, \dots, x_S^0$ za pomocą liniowej funkcji regresji (hiperpowierzchni stopnia drugiego) o następującej postaci:

$$\hat{y} = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + \dots + b_Sx_S$$

Struktura planu eksperymentu:

Plan eksperymentu polega na ustaleniu wartości wejść $x_{n1}, x_{n2}, x_{n3}, \dots, x_{nS}$ przy serii N doświadczeń $n = 1, 2, 3, \dots, N$

Zapis przebiegu eksperymentu zaprezentować można w formie tablicy:

x_{11}	x_{12}	...	x_{1S}	y_1
x_{21}	x_{22}	...	x_{2S}	y_2
...
x_{N1}	x_{N2}	...	x_{NS}	y_N

Po wyliczeniu kierunku gradientu opisanego przez współczynniki $b_0, b_1, b_2, \dots, b_S$ funkcji regresji rozpoczyna się poszukiwania punktu ekstremalnego na tym kierunku.

W planie dwupoziomowym każde z wejść x_s może przyjmować wartość na jednym z dwu poziomów:

$$x_s: x_s^0 - \Delta x_s, x_s^0 + \Delta x_s \text{ dla } s = 1, 2, 3, \dots, S$$

8.3.2 Zastosowanie procedury optymalizacji wg metody Boxa-Wilsona

W procedurze poszukiwania ekstremum za pomocą metody Boxa-Wilsona można wyróżnić dwa główne etapy:

- a) Przeprowadzenie **małej liczby doświadczeń** które powinny umożliwić ustalenie kierunku gradientu. Na tym kierunku oczekuje się punktu ekstremalnego. Celem etapu jest wyznaczenie parametrów hiperpłaszczyzny stopnia pierwszego, opisującej lokalny charakter zmienności funkcji. Opcjonalnie powtarza się serię doświadczeń w celu znalezienia nowego optimum oraz nowego kierunku gradientu, dopóki liniowy opis nie gwarantuje poprawy wartości ekstremum
- b) Wykonanie **dodatkowych eksperymentów** w celu dokładniejszego znalezienia ekstremum. Wykorzystuje się w tym celu opis matematyczny w postaci hiperpłaszczyzny drugiego stopnia.

Uwaga: teksty źródłowe programów w języku LEKS rozwiązujących omawiane zadania zamieszczono na dysku CD w odpowiednim katalogu.

Przypadek 1. Optymalizacja z wykorzystaniem metody Boxa-Wilsona dla **planu dwupoziomowego**, eksperymentu całkowitego, o dwu wejściach ($S=2$)

Założenia:

- a) postać funkcji celu

$$y = f(x^1, x^2), \text{ gdzie:}$$

y - wynik finansowy netto w pierwszym roku sekwencji trzyletniej [zł], funkcja ciągła z pojedynczym ekstremum [Rys. 8-1]

x^1 - cena zbytu wyrobu gotowego ALFA [zł / jednostkę wyrobu]

x^2 - cena zbytu wyrobu gotowego GAMMA [zł / jednostkę wyrobu]

- b) punkty startowe (początkowe wartości zmiennych wejściowych)

cena wyrobu ALFA x_0^1 - 250 [zł/jednostkę wyrobu]

cena wyrobu GAMMA x_0^2 - 411 [zł/jednostkę wyrobu]

Wokół punktów startowych x_0 aproksymuje się charakterystykę ekstremalną $y = f(x)$ za pomocą hiperpowierzchni stopnia pierwszego $y = b_0 + b_1x^1 + b_2x^2$

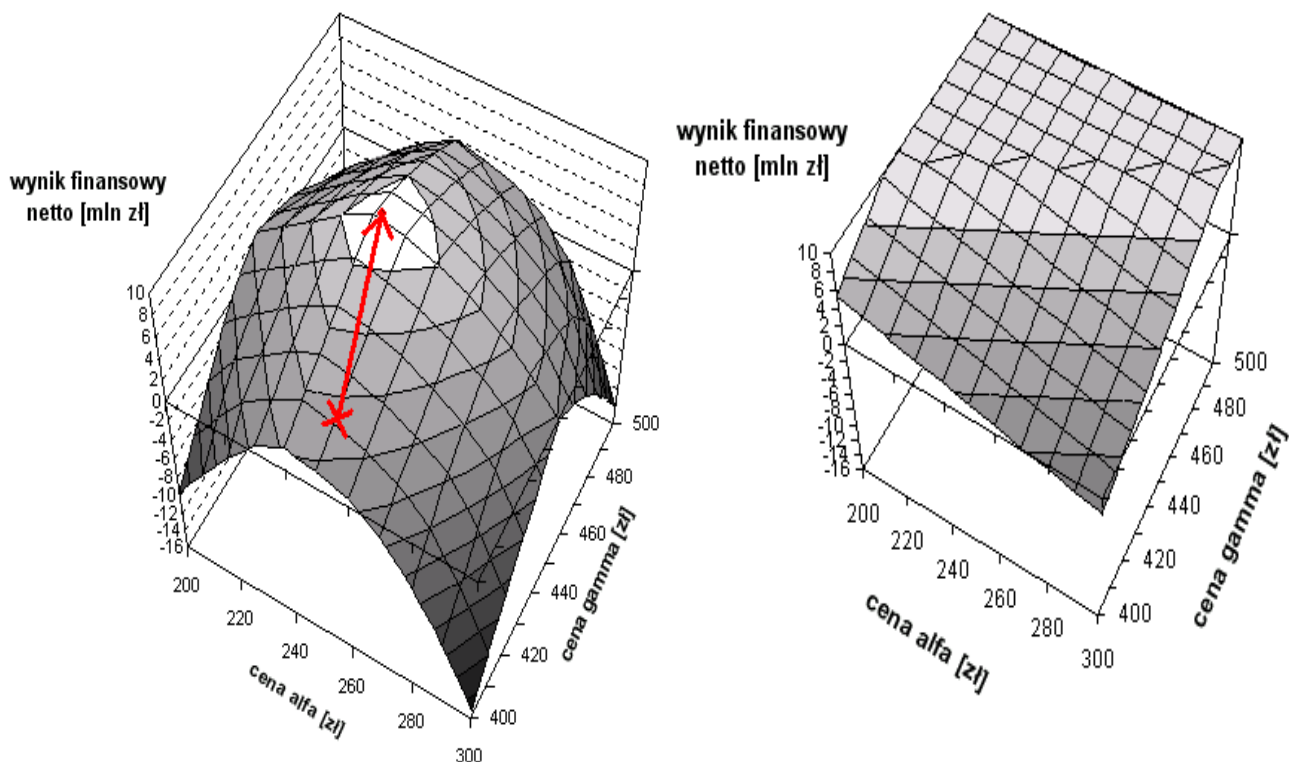
c) liniowe ograniczenia dotyczące zmiennych wejściowych

wyrób ALFA: $0 \leq x^1 \leq 500$ [zł/jednostkę wyrobu]

wyrób GAMMA: $0 \leq x^2 \leq 600$ [zł/jednostkę wyrobu]

d) arbitralnie przyjmujemy kroki próbne $\Delta x^1 = 5$ oraz $\Delta x^2 = 6$ [zł/jednostkę wyrobu]

Postępujemy zgodnie z algorytmem, budując plan eksperymentu o $2^S = 2^2 = 4$ doświadczeniach. Punkty reprezentujące kombinacje wartości zmiennych wejściowych w poszczególnych doświadczeniach otaczają z czterech stron punkt startowy x_0 .



Rys. 8-1 Wykres funkcji celu (fragment dziedziny) z zaznaczonym punktem startowym oraz punktem ekstremalnym znalezionym w efekcie rozwiązania problemu „przypadek 1” oraz hiperpłaszczyzna wyznaczona w trakcie procedury

e) wyniki eksperymentu zapisujemy w postaci macierzy wejść X i wektora wyjść obiektu y

$$X = \begin{bmatrix} 1 & 245 & 405 \\ 1 & 255 & 405 \\ 1 & 245 & 417 \\ 1 & 255 & 417 \end{bmatrix} \quad y = \begin{bmatrix} 2,730 \\ 1,989 \\ 4,540 \\ 3,799 \end{bmatrix}$$

Ponieważ wyliczenie współczynników modelu matematycznego postaci (równania regresji):

$$b = (X^T X)^{-1} X^T y = \begin{bmatrix} b^0 \\ b^1 \\ b^2 \end{bmatrix}$$

jest zadaniem rachunkowo dość trudnym, dokonujemy prostej zamiany zmiennych (standaryzacji).

Standaryzacja polega na przeniesieniu początku układu współrzędnych do punktu startowego oraz zmiany skali tak aby wartości kroków (zmian) próbnych były w nowym układzie współrzędnych ujmowane jako jednostkowe (-1, +1). W tym celu przyjmuje się nowe zmienne (w przykładzie $S = 2$, $s: 1, 2$ w sposób następujący:

$$t^s = \frac{x^s - x_0^s}{\Delta x^s} \quad \text{stąd} \quad x^s = x_0^s + t^s \Delta x^s \quad \text{gdzie} \quad \Delta x^s \quad \text{to wartość bezwzględna kroku (wzdłuż}$$

odpowiedniej osi w nowej skali).

Po przekształceniach [Mańczak 1974] otrzymujemy wzór:

$$\hat{y} = k_0 + \sum_{s=1}^S k_s t_s \quad \text{gdzie:} \quad k_0 = b_0 + \sum_{s=1}^S b_s x_s^0 \quad \text{oraz} \quad k_s = b_s \Delta x_s$$

W omawianym przypadku obliczenia przeprowadzane są w sposób następujący:

$$t_1 = \frac{x_1 - 250}{5} \quad \text{oraz} \quad t_2 = \frac{x_2 - 411}{6} \quad \text{stąd} \quad x_1 = 250 + 5t_2 \quad \text{oraz} \quad x_2 = 411 + 6t_2,$$

w nowym układzie współrzędnych t_s : -1, +1

funkcja regresji w nowym układzie współrzędnych ma postać:

$$\hat{y} = k_0 + k_1 x^1 + k_2 x^2 \quad \text{gdzie} \quad k_0 = b_0 + 250b_1 + 411b_2, \quad k_1 = 5b_1, \quad k_2 = 6b_2$$

f) nowa macierz wejść T (plan eksperymentu) otrzymana po zamianie i standaryzacji zmiennych ma następującą postać:

$$T = \begin{bmatrix} 1 & -1 & -1 \\ 1 & +1 & -1 \\ 1 & -1 & +1 \\ 1 & +1 & +1 \end{bmatrix} \quad \text{stąd} \quad (T^T T)^{-1} = \frac{1}{4} I, \quad \text{oraz} \quad k = \frac{1}{4} T^T y$$

wyliczamy wektor k:

$$\begin{matrix} k_0 \\ k_1 \\ k_2 \end{matrix} = \frac{1}{4} \begin{bmatrix} & & & & \\ & 1 & & & \\ -1 & 1 & -1 & 1 & \\ -1 & -1 & 1 & 1 & \end{bmatrix} \begin{bmatrix} 2,73 \\ 1,98 \\ 4,54 \\ 3,80 \end{bmatrix} = \begin{bmatrix} 3,26 \\ -0,37 \\ 0,9 \end{bmatrix}$$

stąd model matematyczny w nowym układzie współrzędnych przedstawia się następująco:

$$\hat{y} = 3,26 - 0,37x^1 + 0,9x^2$$

natomiast w oryginalnym układzie współrzędnych (po rozwiązaniu układu równań z trzema niewiadomymi) współczynniki b wynoszą:

$$b = \begin{bmatrix} b^0 \\ b^1 \\ b^2 \end{bmatrix} = \begin{bmatrix} -39,89 \\ -0,074 \\ 0,15 \end{bmatrix} \text{ czyli } y = -39,89 - 0,074x^1 + 0,15x^2$$

g) po wyznaczeniu funkcji regresji przyjmujemy nowe kroki robocze (nieco mniejsze od obliczonych współczynników pomnożonych przez wartość kroku próbnego $k^s \Delta x^s$, staramy się aby były to wartości „okrągłe”) i wykonujemy dalsze doświadczenia [Tab. 8-1] zmieniając wartość zmiennych wejściowych zgodnie z wyznaczonymi kierunkami (krokami roboczymi). Za każdym razem sprawdzamy, czy nie naruszyliśmy ograniczeń.

Przyjeliśmy, że kroki robocze są równe:

$$\Delta_R^{x_1} = -1 \text{ oraz } \Delta_R^{x_2} = 5$$

Wyznaczone maksimum wyniku finansowego netto wynosi 9,470 mln zł i jest osiągnięte przy cenie wyrobu ALFA równej 242 złote oraz cenie wyrobu GAMMA równej 451 złotych.

Cała procedura powtarzana jest w drugiej iteracji [Tab. 8-2]. **Nowym punktem startowym staje się punkt ekstremalny.** Okazuje się, że wyznaczony w drugiej iteracji punkt ekstremalny nie zmienił położenia. Co więcej, dla czterech punktów otaczających punkt startowy także wynik finansowy netto jest niższy niż dla punktu ekstremalnego.

Ilustracją działania procedury Boxa-Wilsona dla omawianego przypadku jest rysunek [Rys. 8-2]. Średnica kół jest proporcjonalna do wartości wyniku finansowego. W prawej dolnej części widoczny jest punkt startowy iteracji pierwszej (numer 0), otoczony czterema punktami doświadczeń (o numerach od 1 do 4) oddalonych od niego o długość kroku próbnego w każdej osi. Zauważamy zwiększające się średnice kół w kierunku lewej górnej krawędzi. Widać wyraźnie że odległość między środkami kół jest stała. Są to dodatkowe doświadczenia o numerach od 5 do 12. Doświadczenie nr 12 jest wyznaczonym w pierwszej iteracji ekstremum. Następnie wykonywana jest iteracja nr 2. Rozpoczyna się ona od zbadania czterech punktów w otoczeniu punktu ekstremalnego (nowego punktu startowego). Wyznaczono następnie nowe kroki robocze. Okazało się jednak, że wykonywane

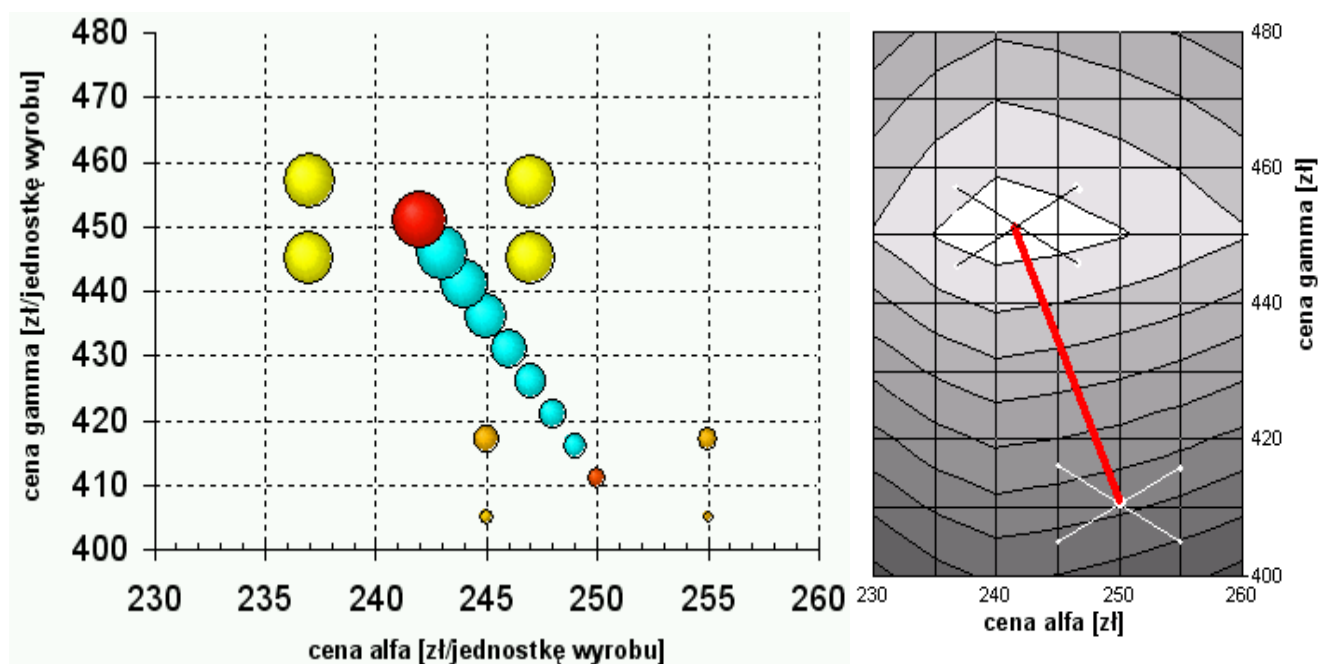
doświadczenia nie powodują zwiększenia wartości wyniku finansowego netto. Przyjmujemy zatem że punkt ekstremalny nie zmienił się.

Tab. 8-1 Przypadek 1 – Parametry metody Boxa-Wilsona: plan eksperymentu dla pierwszej iteracji

Czynniki x^s	x^1 - cena wyr. ALFA [zł / jednostkę wyr.]	x^2 - cena wyr. GAMMA [zł / jednostkę wyr.]	y - wynik eksperymentu: wynik finansowy netto [mln zł]
Pierwsza iteracja			
Poziom podstawowy x_0^s	250	411	3,320
Krok próbny x^s	5	6	
Poziom górny $x_0^s - \Delta x^s$	255	417	
Poziom dolny $x_0^s + \Delta x^s$	245	405	
Doświadczenie nr 1	245	405	2,730
Doświadczenie nr 2	255	405	1,989
Doświadczenie nr 3	245	417	4,540
Doświadczenie nr 4	255	417	3,799
Współczynniki k^s	-0,37	0,9	($k^0=3,26$)
Współczynniki b^s	-0,074	0,15	($b^0 = -39,89$)
Wyrażenie $k^s \Delta x^s$	-1,83	+5,42	
Krok roboczy Δx_R^s	-1	5	
Doświadczenie 5	249	416	4,146
Doświadczenie 6	248	421	4,968
Doświadczenie 7	247	426	5,785
Doświadczenie 8	246	431	6,598
Doświadczenie 9	245	436	7,406
Doświadczenie 10	244	441	8,210
Doświadczenie 11	243	446	9,010
Doświadczenie 12	242	451	9,470
Maksimum	242	451	9,470

Tab. 8-2 Przypadek 1 – Parametry metody Boxa-Wilsona: plan eksperymentu dla drugiej iteracji

Czynniki x^s	x^1 - cena wyr. Alfa [zł / jednostkę wyr.]	x^2 - cena wyr. Gamma [zł / jednostkę wyr.]	Y - wynik eksperymentu: wynik finansowy netto [mln zł]
Druga iteracja			
Poziom podstawowy x_0^s	242	451	9.470
Krok próbny x^s	5	6	
Poziom górny $x_0^s - \Delta x^s$	247	457	
Poziom dolny $x_0^s + \Delta x^s$	237	445	
Doświadczenie nr 1	237	445	8,808
Doświadczenie nr 2	247	445	8,650
Doświadczenie nr 3	237	457	8,970
Doświadczenie nr 4	247	457	8,813
Współczynniki k^s	-0,079	0,0811	($k^0=8,810$)
Wyrażenie $k^s \Delta x^s$	-0,395	0,487	
Krok roboczy Δx_R^s	-0,3	0,4	
Maksimum (takie samo jak w iteracji nr 1)	242	451	9,470



Rys. 8-2 Ilustracja metody Boxa-Wilsona dla przypadku z nieaktywnymi ograniczeniami na zmienne decyzyjne

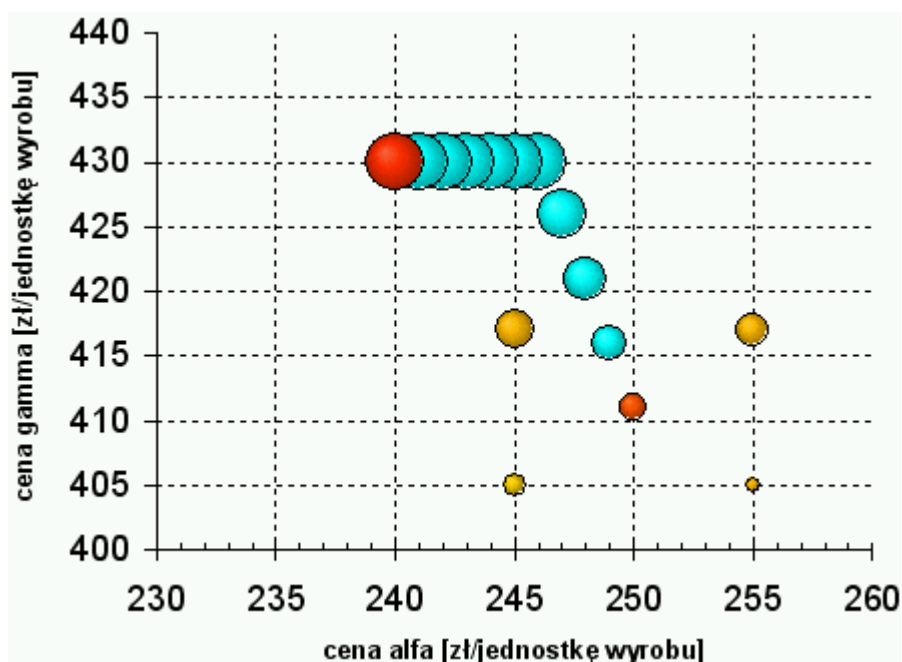
Przypadek 2. Warunki identyczne jak w przypadku 1, jednak zmienione zostają ograniczenia na zmienne decyzyjne (wejściowe).

liniowe ograniczenia dotyczące zmiennych wejściowych

wyrób ALFA: $240 \leq x^1 \leq 300$ [zł/jednostkę wyrobu]

wyrób GAMMA: $400 \leq x^2 \leq 430$ [zł/jednostkę wyrobu]

Ilustracją działania procedury Boxa-Wilsons dla omawianego przypadku jest rysunek [Rys. 8-3] i tabela [Tab. 8-3]. Średnica kół jest proporcjonalna do wartości wyniku finansowego. W prawej dolnej części widoczny jest punkt startowy iteracji pierwszej i jedynej (numer 0), otoczony czterema punktami doświadczeń (o numerach od 1 do 4) oddalonych od niego o długość kroku próbnego w każdej osi. Zauważamy zwiększające się średnice kół w kierunku lewej górnej krawędzi. Widać wyraźnie, że odległość między środkami kół jest stała. Są to dodatkowe doświadczenia o numerach od 5 do 14. Zauważyć należy, że w doświadczeniach o numerach od 5 do 7 ograniczenia nie ingerowały. Następnie w doświadczeniu nr 8 aktywne staje się ograniczenie na zmienną x^1 , czyli cenę wyrobu Gamma. W doświadczeniu o numerze 14 aktywne staje się także ograniczenie dotyczące zmiennej x^2 a poszukiwanie punktu ekstremalnego dobiegło końca. Kolejne iteracje algorytmu nie są wykonywane gdyż otoczenie (cztery punkty) nowego punktu startowego znajdowałoby się częściowo poza dziedziną zmiennych decyzyjnych (poza ograniczeniami).



Rys. 8-3 Ilustracja metody Boxa-Wilsons dla przypadku z aktywnymi ograniczeniami zmiennych decyzyjnych

Tab. 8-3 Przypadek 2 – Parametry metody Boxa-Wilsons: plan eksperymentu w trakcie którego dochodzi do naruszenia liniowych ograniczeń na zmienne decyzyjne. Pojedyncza iteracja

Czynniki x^s	x^1 - cena wyr. ALFA [zł / jednostkę wyr.]	x^2 - cena wyr. GAMMA [zł / jednostkę wyr.]	Y - wynik eksperymentu: wynik finansowy netto [mln zł]
Pierwsza iteracja (i jedyna)			
Poziom podstawowy x_0^s	250	411	3,320
Krok próbny x^s	5	6	
Poziom górny $x_0^s - \Delta x^s$	255	417	
Poziom dolny $x_0^s + \Delta x^s$	245	405	
Doświadczenie nr 1	245	405	2,730
Doświadczenie nr 2	255	405	1,989
Doświadczenie nr 3	245	417	4,540
Doświadczenie nr 4	255	417	3,799
Współczynniki k^s	-0,37	0,9	($k^0=3,26$)
Wyrażenie $k^s \Delta x^s$	-1,83	+5,42	
Krok roboczy Δ_R^s	-1	5	
Doświadczenie 5	249	416	4,146
Doświadczenie 6	248	421	4,968
Doświadczenie 7	247	426	5,785
Doświadczenie 8	246	430	6,447
Doświadczenie 9	245	430	6,501
Doświadczenie 10	244	430	6,551
Doświadczenie 11	243	430	6,596
Doświadczenie 12	242	430	6,637
Doświadczenie 13	241	430	6,674
Doświadczenie 14	240	430	6,706
Maksimum	240	430	6,706

Przypadek 3. Optymalizacja z wykorzystaniem metody Boxa-Wilsons dla planu dwupoziomowego, eksperymentu całkowitego, o trzech wejściach (S=3)

a) postać funkcji celu

$$y = f(x^1, x^2, x^3), \text{ gdzie:}$$

y – wynik finansowy netto w pierwszym roku sekwencji trzyletniej [zł], funkcja ciągła z pojedynczym ekstremum

x^1 - cena zbytu wyrobu gotowego ALFA [zł / jednostkę wyrobu]

x^2 - cena zbytu wyrobu gotowego BETA [zł / jednostkę wyrobu]

x^3 - cena zbytu wyrobu gotowego GAMMA [zł / jednostkę wyrobu]

Szukamy maksimum wyjścia obiektu o S=3 wejściach (np. ceny alfa, beta, gamma) opisanego nieznaną (z punktu widzenia obserwatora) funkcją nieliniową

b) punkty startowe (początkowe wartości zmiennych wejściowych)

cena wyrobu ALFA x_0^1 - 250 [zł/jednostkę wyrobu]

cena wyrobu BETA x_0^2 - 340 [zł/jednostkę wyrobu]

cena wyrobu GAMMA x_0^3 - 411 [zł/jednostkę wyrobu]

Wokół punktów startowych x_0 aproksymuje się charakterystykę ekstremalną $y = f(x)$ za pomocą hiperpowierzchni stopnia pierwszego $y = b_0 + b_1x^1 + b_2x^2 + b_3x^3$

c) liniowe ograniczenia dotyczące zmiennych wejściowych

liniowe ograniczenia dotyczące zmiennych wejściowych

wyrób ALFA: $240 \leq x^1 \leq 300$ [zł/jednostkę wyrobu]

wyrób BETA: $300 \leq x^2 \leq 370$ [zł/jednostkę wyrobu]

wyrób GAMMA: $400 \leq x^3 \leq 430$ [zł/jednostkę wyrobu]

d) arbitralnie przyjmujemy kroki próbne $\Delta x^1 = 5$; $\Delta x^2 = 5$; $\Delta x^3 = 6$ [zł/jednostkę wyrobu]

Postępujemy zgodnie z algorytmem, budując plan eksperymentu o $2^s = 2^3 = 8$ doświadczeniach. Punkty reprezentujące kombinacje wartości zmiennych wejściowych w poszczególnych doświadczeniach otaczają z czterech stron punkt startowy x_0 .

e) wyniki eksperymentu zapisujemy w postaci macierzy wejść X i wektora wyjść obiekt y

$$X = \begin{bmatrix} 1 & 245 & 335 & 405 \\ 1 & 255 & 335 & 405 \\ 1 & 245 & 345 & 405 \\ 1 & 255 & 345 & 405 \\ 1 & 245 & 335 & 417 \\ 1 & 255 & 335 & 417 \\ 1 & 245 & 345 & 417 \\ 1 & 255 & 345 & 417 \end{bmatrix} \quad y = \begin{bmatrix} 1,977 \\ 1,235 \\ 3,485 \\ 2,743 \\ 3,786 \\ 3,045 \\ 5,295 \\ 4,553 \end{bmatrix}$$

ponieważ wyliczenie współczynników modelu matematycznego postaci (równania regresji):

$$b = (X^T X)^{-1} X^T y = \begin{bmatrix} b^0 \\ b^1 \\ b^2 \\ b^3 \end{bmatrix}$$

jest zadaniem rachunkowo dość trudnym, dokonujemy prostej zamiany zmiennych (standaryzacji).

W omawianym przypadku obliczenia przeprowadzane są w sposób następujący:

$$t_1 = \frac{x_1 - 250}{5}, \quad t_2 = \frac{x_2 - 340}{5}, \quad t_3 = \frac{x_3 - 411}{6} \quad \text{stąd}$$

$$x_1 = 250 + 5t_2, \quad x_2 = 340 + 6t_2, \quad x_3 = 411 + 6t_3,$$

w nowym układzie współrzędnych t_s : -1, +1

funkcja regresji w nowym układzie współrzędnych ma postać:

$$\hat{y} = k_0 + k_1 x^1 + k_2 x^2 + k_3 x^3 \quad \text{gdzie}$$

$$k_0 = b_0 + 250b_1 + 340b_2 + 411b_3, \quad k_1 = 5b_1, \quad k_2 = 5b_2, \quad k_3 = 6b_3$$

f) **nowa macierz wejść** T (plan eksperymentu) otrzymana po zamianie i standaryzacji zmiennych ma następującą postać:

$$T = \begin{bmatrix} 1 & -1 & -1 & -1 \\ 1 & +1 & -1 & -1 \\ 1 & -1 & +1 & -1 \\ 1 & +1 & +1 & -1 \\ 1 & -1 & -1 & +1 \\ 1 & +1 & -1 & +1 \\ 1 & -1 & -1 & +1 \\ 1 & +1 & -1 & +1 \end{bmatrix} \quad \text{stąd } (T^T T)^{-1} = \frac{1}{8} I, \text{ oraz } k = \frac{1}{8} T^T y$$

Tab. 3. Przypadek 3 – Parametry metody Boxa-Wilsona: plan eksperymentu w trakcie którego dochodzi do naruszenia liniowych ograniczeń na zmienne decyzyjne. Pojedyncza iteracja

Czynniki x^s	x^1 - cena wyr. Alfa [zł / j.]	x^2 - cena wyr. Beta [zł / j.]	x^3 - cena wyr. Gamma [zł / j.]	y - wynik eksperymentu: wynik finansowy netto [mln zł]
Poziom podstawowy x_0^s	250	340	411	3,320
Krok próbny x^s	5	5	6	
Poziom górny $x_0^s - \Delta x^s$	255	345	417	
Poziom dolny $x_0^s + \Delta x^s$	245	335	405	
Dośw. nr 1	245	335	405	1,977
Dośw. nr 2	255	335	405	1,235
Dośw. nr 3	245	345	405	3,485
Dośw. nr 4	255	345	405	2,743
Dośw. nr 5	245	335	417	3,786
Dośw. nr 6	255	335	417	3,045
Dośw. nr 7	245	345	417	5,295
Dośw. nr 8	255	345	417	4,553
Współczynniki k^s	-0,371	0,754	0,905	($k^0=3,26$)
Wyrażenie $k^s \Delta x^s$	-1,854	3,770	5,429	
Krok roboczy Δ_R^s	-1,5	3	5	
Dośw. nr 9	248,5	343	416	4,633
Dośw. nr 10	247	346	421	5,936
Dośw. nr 11	245,5	349	426	7,229
Dośw. nr 12	244	352	430	8,361
Dośw. nr 13	242,5	355	430	8,879
Dośw. nr 14	241	358	430	9,388
Dośw. nr 15	240	361	430	9,873
Dośw. nr 16	240	364	430	10,104
Dośw. nr 17	240	367	430	10,190
Dośw. nr 18	240	370	430	10,249
Maksimum	240	370	430	10,249

Aktywację ograniczeń zaznaczono pogrubieniem czcionki.

Przypadek 4. Dobór wielkości zamówień i ceny wyrobu. Optymalizacja z wykorzystaniem metody Boxa-Wilsona dla planu dwupoziomowego, eksperymentu całkowitego, o dwu wejściach ($S=2$)

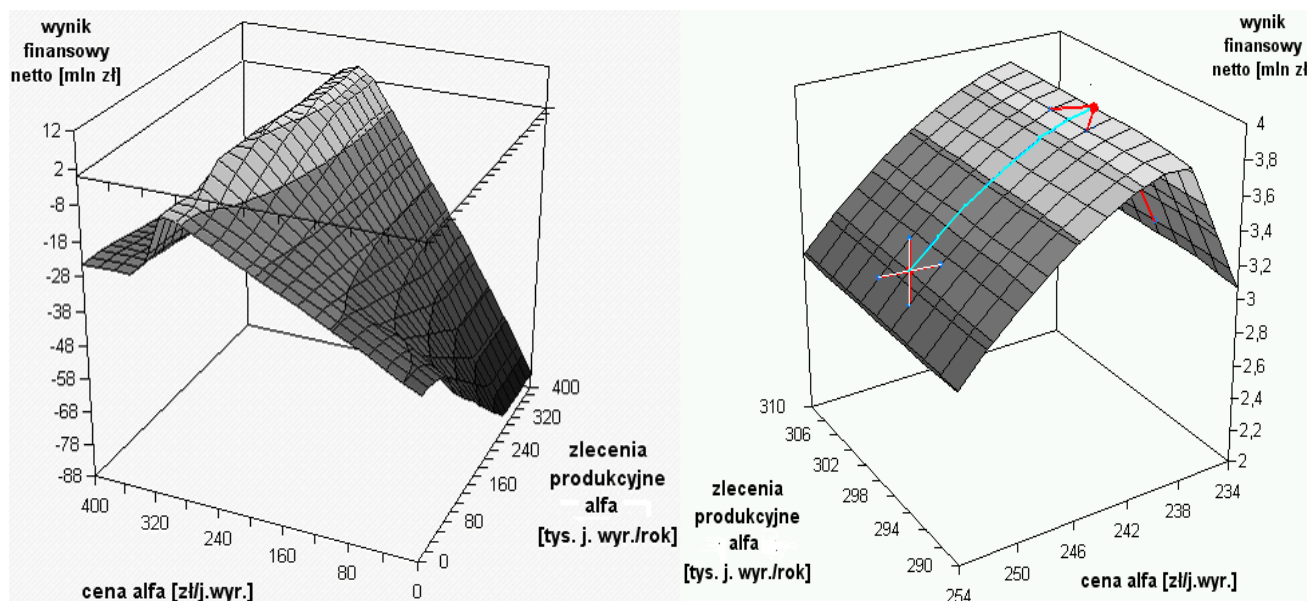
a) postać funkcji celu

$$y = f(x^1, x^2), \text{ gdzie:}$$

y – wynik finansowy netto w pierwszym roku sekwencji trzyletniej [zł], funkcja ciągła z pojedynczym ekstremum [Rys. 8-4]

x^1 - cena zbytu wyrobu gotowego ALFA [zł / jednostkę wyrobu]

x^2 - wielkość zamówień wyrobu gotowego ALFA [tys. jednostek wyrobu]



Rys. 8-4 Wykresy funkcji celu (fragment dziedziny) z zaznaczonym punktem startowym oraz punktem ekstremalnym znalezionym w efekcie rozwiązania problemu „Przypadek 4”

b) punkty startowe (początkowe wartości zmiennych wejściowych)

cena wyrobu ALFA x_0^1 - 250 [zł/jednostkę wyrobu]

zlecenia produkcyjne wyrobu GAMMA x_0^2 - 300 [tys. jednostek wyrobu / rok]

Wokół punktów startowych x_0 aproksymuje się charakterystykę ekstremalną $y = f(x)$ za pomocą hiperpowierzchni stopnia pierwszego $y = b_0 + b_1x^1 + b_2x^2$

c) liniowe ograniczenia dotyczące zmiennych wejściowych

ceny: $0 \leq x^1 \leq 400$ [zł/jednostkę wyrobu]

wyrób GAMMA: $0 \leq x^2 \leq 400$ [zł/jednostkę wyrobu]

d) arbitralnie przyjmujemy kroki próbne $\Delta x^1 = 2$ oraz $\Delta x^2 = 2$ [odpowiednie jednostki]

Tab. 8-4 Przypadek 4 –plan eksperymentu dla kolejno wykonywanych iteracji

Czynniki x^s	x^1 - cena wyr. ALFA [zł / jednostkę wyr.]	x^2 - zlecenie produkcyjne wyr. ALFA [tys. jednostek wyr./rok]	y - wynik eksperymentu: wynik finansowy netto [mln zł]
Pierwsza iteracja			
Poziom podstawowy x_0^s	250,00	300	3,32
Krok próbny x^s	2	2	
Poziom górny $x_0^s - \Delta x^s$	252,00	302	
Poziom dolny $x_0^s + \Delta x^s$	248,00	298	

Doświadczenie nr 2	252,00	298	3,163
Doświadczenie nr 3	248,00	302	3,456
Doświadczenie nr 4	252,00	302	3,163
Współczynniki k^s	-0,1483	0	($k^0=3,3113$)
Wyrażenie $k^s \Delta x^s$	-0,297	0	
Krok roboczy Δx_R^s	-0,2	0	
Doświadczenie 5	249,80	300	3,335
Doświadczenie 6	249,60	300	3,349
Doświadczenie 7	249,40	300	3,363
Doświadczenie 8	249,20	300	3,378
Doświadczenie 9	249,00	300	3,392
Doświadczenie 10	248,80	300	3,4059
Doświadczenie **	**	**	**
Doświadczenie 64	238,20	300	3,887
Maksimum	238,20	300	3,887
Druuga iteracja			
Poziom podstawowy x_0^s	238,20	300	3,887
Krok próbny x^s	2	2	
Poziom górny $x_0^s - \Delta x^s$	240,20	302	
Poziom dolny $x_0^s + \Delta x^s$	236,20	298	
Doświadczenie nr 1	236,20	298	3,527
Doświadczenie nr 2	240,20	298	3,834
Doświadczenie nr 3	236,20	302	3,527
Doświadczenie nr 4	240,20	302	3,834
Współczynniki k^s	-0,1483	0	($k^0=3,313$)
Wyrażenie $k^s \Delta x^s$	-0,297	0	Uwaga: kierowanie się zgodnie z gradientem pogarsza wynik
Krok roboczy Δx_R^s	-0,2	0	

Maksimum (takie samo jak w iteracji nr 1)	238,20	300	3,887
---	--------	-----	-------

Postępujemy zgodnie z algorytmem, budując plan eksperymentu o $2^S = 2^2 = 4$ doświadczeniach. Punkty reprezentujące kombinacje wartości zmiennych wejściowych w poszczególnych doświadczeniach otaczają z czterech stron punkt startowy x_0 . Sposób przeprowadzania obliczeń jest analogiczny jak w przypadku 1, jednak wprowadzono dodatkowy warunek stopu. Kompletna procedura poszukiwania rozwiązań zakończy się gdy liczba pozytywnych (tzn. legitymujących się poprawą funkcji celu) prób jest równa 0 dla bieżącego punktu startowego. Przebieg eksperymentu przedstawiono w tabeli [Tab. 8-4].

Na podstawie [Tab. 8-4] a w szczególności drugiej iteracji widać, że kierowanie się zgodnie z gradientem pogarsza wynik. Co więcej, w punktach otaczających nowy punkt startowy (reprezentujący ekstremum poprzedniej iteracji) wartość wyniku finansowego netto jest poniżej maksymalnej (wyniku finansowego netto dla punktu startowego). Z powyższego wnioskujemy, że maksimum lokalne znajduje się w punkcie o następujących współrzędnych: cena alfa w wysokości 238,20 zł oraz zleceń produkcyjnych w wysokości 300 j. wyr./rok

Podsumowanie. W zaprezentowanych eksperymentach kolejny raz wykazano że "liczne przykłady zastosowań świadczą o dużej wartości praktycznej idei Boxa i Wilsona - połączenia metody gradientowej z eksperymentem czynnikiem w celu identyfikacji i optymalizacji obiektów wielowymiarowych" [Mańczak 1974]. Okazało się to prawdą nawet bez standaryzacji zmiennych wejściowych.

8.4 Badanie przebiegu wartości zmiennych zależnych z wykorzystaniem języka LEKS

8.4.1 Wyznaczenie trójwymiarowego wykresu wartości funkcji celu za pomocą programu w języku LEKS

Sformułowanie ramowego problemu badawczego:

Opracować program w języku LEKS umożliwiający wykonanie **trójwymiarowego wykresu** zależności wartości zmiennej zależnej od wartości zmiennych niezależnych mieszczących się w określonym zakresie.

Konkretyzacja problemu badawczego:

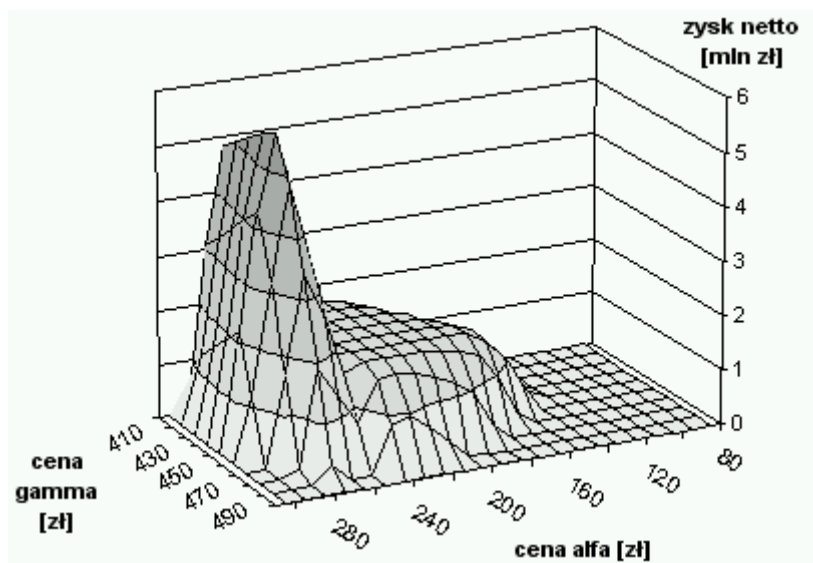
Opracować program w języku LEKS, efektem działania którego będzie liczbowe zestawienie zależności **wyniku finansowego netto** w pierwszym roku sekwencji decyzyjnej od wysokości cen wyrobu alfa i beta w tymże roku. Zakładamy że spełnione są jednocześnie następujące warunki (zakres wartości zmiennych niezależnych):

$$500 \geq \text{cena wyrobu alfa [zł/jedn.]} \geq 400,$$

$$300 \geq \text{cena wyrobu gamma [zł/jedn.]} \geq 80,$$

$$\text{cena wyrobu beta [zł/jedn.]} = 400$$

Wykonać wykres trójwymiarowy przedstawiające warstwy zakresów zmiennej zależnej. W przypadku występowania straty netto przyjąć wynik finansowy netto równy 0 zł [Rys. 8-5]. Rozwiązania wyznaczone zostaną za pomocą techniki krokowych zmian wartości zmiennych w przedziale (jest to symulacja inkrementacyjna).



Rys. 8-5 Wykres zmiennej zysk netto w zależności od cen wyrobów alfa i gamma mieszczących się z założonych przedziałach

Poniżej zamieszczono tekst źródłowy procedur w języku LEKS.

```
rem ----- wyzerowanie wartosci wszystkich zmiennych -----
rem ----- włączenie mozliwosci raportowania wartosci zmiennych -----
rem ----- uruchomienie ciaglego wykonywania programu -----
clear
raport
run

rem ----- deklaracje zmiennych -----
deklaracja cena_alfa
deklaracja cena_beta
deklaracja cena_gamma

deklaracja pocz_cena_alfa
deklaracja pocz_cena_beta
deklaracja pocz_cena_gamma

deklaracja zysk_netto

deklaracja wezel_poczkowy
deklaracja wezel_ojciec
deklaracja wezel_potomek

rem ----- ustalenie który wezel bedzie wezlem wzorcowym -----
odczyt wezel wezel_poczkowy
odczyt ojciec wezel_poczkowy, wezel_ojciec

rem ----- początkowe wartosci zmiennych -----

przypisz pocz_cena_alfa, 80
przypisz pocz_cena_beta, 400.00
przypisz pocz_cena_gamma, 410
przypisz cena_alfa, pocz_cena_alfa
przypisz cena_beta, pocz_cena_beta
przypisz cena_gamma, pocz_cena_gamma

rem ----- zewnętrzna pętla ----- zmiany wartosci ceny wyrobu alfa -----
while <= cena_alfa, 300
  przypisz cena_gamma, pocz_cena_gamma

rem ----- wewnętrzna pętla ----- zmiany wartosci ceny wyrobu gamma ----
while <= cena_gamma, 500

rem ----- utworzenie eksperymentalnego wezla -----
dodaj wezel wezel_ojciec, wezel_poczkowy, wezel_potomek
ustaw wezel wezel_potomek

rem ----- zmiana wartosci podmacierzy zmiennych decyzyjnych -----
zmiana cena alfa rok1 cena_alfa
zmiana cena beta rok1 cena_beta
zmiana cena gamma rok1 cena_gamma
```

```

rem ----- wykonanie eksperymentu symulacyjnego -----
symuluj węzeł wezel_potomek
ustaw węzeł wezel_potomek

rem ----- odczyt zmiennej zależnej (funkcji celu) -----
rem ----- zapis wartości do pliku -----
odczyt bilans 5, 2, 3, zysk_netto
zannotuj cena_alfa
zannotuj cena_gamma
zannotuj zysk_netto

rem ----- inkrementacja zmiennej cena_gamma (petla wewnętrzna) -----
+ cena_gamma, cena_gamma, 10

rem ----- wezeł eksperymentalny już niepotrzebny -----
kasuj węzeł wezel_potomek
rem ----- koniec pętli wewnętrznej -----
endwhile

rem ----- inkrementacja zmiennej cena_alfa (petla zewnętrzna) -----
+ cena_alfa, cena_alfa, 10
rem ----- koniec pętli zewnętrznej -----
endwhile

rem ----- koniec programu -----

stop all

```

8.4.2 Zastosowanie języka LEKS w problemie wyznaczania hiperpowierzchni progu rentowności

Sformułowanie **ramowego problemu badawczego**:

Opracować program w języku LEKS umożliwiający **wyznaczenie progu rentowności** w postaci **wielowymiarowej hiperpowierzchni** utworzonej przez punkty o współrzędnych równych wartościom zmiennych niezależnych, z których niektóre mieszczą się w określonym zakresie, natomiast wartości pozostałych są wyznaczane w toku prowadzonych eksperymentów.

Konkretyzacja problemu badawczego:

Opracować program w języku LEKS dostarczający danych liczbowych obrazujących pojęcie **progu rentowności** czyli wyniku finansowego netto równego (lub bliskiego) zeru dla wielkości zleceń produkcyjnych wyrobu Gamma w funkcji zleceń produkcyjnych pozostałych wyrobów: Alfa i Beta w pierwszym roku sekwencji decyzyjnej.

Specyfikację zadania wyrazić można następująco:

Założenie: Stała struktura asortymentowa produkcji. Symulowany czas : 1 rok.

Cel: Wyznaczenie progu rentowności (ilościowego) dla wyrobu Gamma w funkcji zleceń produkcyjnych wyrobów Alfa i Beta przy założonej dokładności (+/- 1 tys. j. wyr.)

Technika wyznaczenia rozwiązań: sterowane krokowe przeszukiwanie przedziału.

Metoda weryfikacji znalezionej odpowiedzi: wykres wyniku finansowego netto w funkcji zleceń produkcyjnych wyrobów Alfa i Beta. Wykonać wykres trójwymiarowy z wyszczególnionymi warstwami przedstawiającymi zakresy wyznaczonych wartości zmiennej niezależnej.

Autorska technika **sterowanego krokowego przeszukiwania przedziału** w przypadku wyznaczania miejsc zerowych (progu rentowności) składa się z następujących etapów:

- a) Przypisanie wartości początkowych zmiennym
- b) Generowanie zestawów zmiennych decyzyjnych. Określić należy rodzaj modyfikowanych zmiennych, ich zakres oraz regułę modyfikacji wartości w poszczególnych krokach. W prezentowanej implementacji kolejno badane wartości poszczególnych zmiennych oddalone są od siebie o stałą wartość (krok). Dlatego też opracowane na podstawie badań wykresy mogą korzystać z interpolacji za pomocą hiperpowierzchni. Kolejność warstw w strukturze pętli jest dowolna, jednak pętla odpowiedzialna za wyszukiwanie miejsca zerowego musi być umieszczona w najgłębszej warstwie (będzie uruchamiana dla każdego zestawu zmiennych decyzyjnych)
- c) Właściwe poszukiwanie progu rentowności (krokowo; od początku zakresu do końca zakresu). Przebiega w dwu etapach: w pierwszym zmiany wartości badanej zmiennej są dość duże a w drugim znacznie mniejsze. Każdy z etapów kończy się gdy zaobserwowana zostanie **zmiana znaku badanej zmiennej**.

Poniżej przedstawiono **tekst źródłowy programu**, wykres wyników [Rys. 8-6] osiągniętych po jego zakończeniu oraz przykłady (dwa wykresy) interpretacji progu rentowności.

Wartości parametrów są następujące:

Założono pojedynczy próg rentowności. Jeśli próg nie zostanie znaleziony, to przyjmuje się umownie odpowiednio dużą wartość dodatnią lub ujemną (np. równą zakresowi zmiennej zależnej na wykresie). W pozostałych wypadkach algorytm sprawdzi się w poszukiwaniu pierwszego (licząc od początku zakresu poszukiwań) progu rentowności przy czym kolejne progi rentowności nie powinny być od siebie oddalone o mniej niż ± 20 i 2 [tys. j. wyr./rok] w odpowiednio w pierwszym i drugim etapie poszukiwań.

Cena wyrobu Alfa [zł/jedn. wyr] = 260

Cena wyrobu Beta [zł/jedn. wyr] = 395

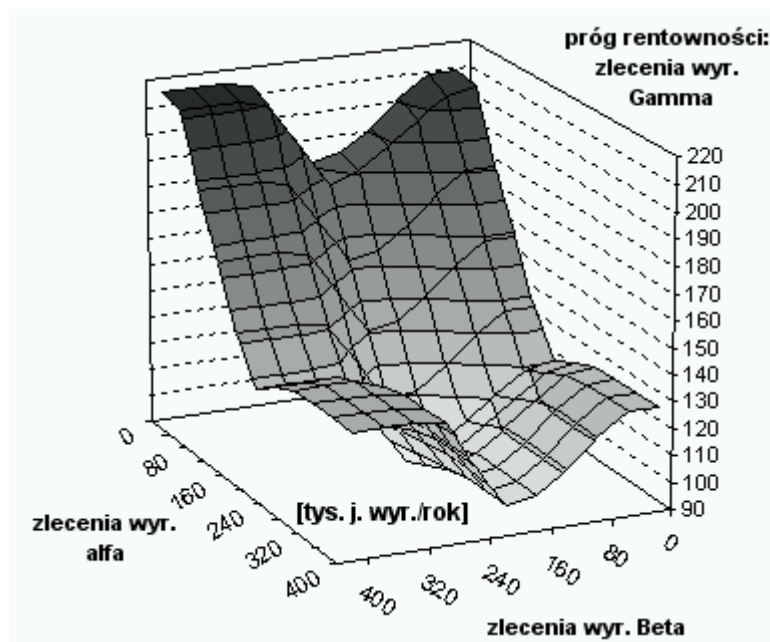
Cena wyrobu Gamma [zł/jedn. wyr] = 470

Zakres zmian zleceń produkcyjnych wyrobów Alfa i Beta: od 0 do 400 [tys. j. wyr./rok]

Zakres poszukiwań wartości progu rentowności zleceń produkcyjnych wyrobu Gamma: od 0 do 800 [tys. j. wyr./rok]

Początkowa długość kroku poszukiwań progu rentowności: 20 [tys. j. wyr./rok]

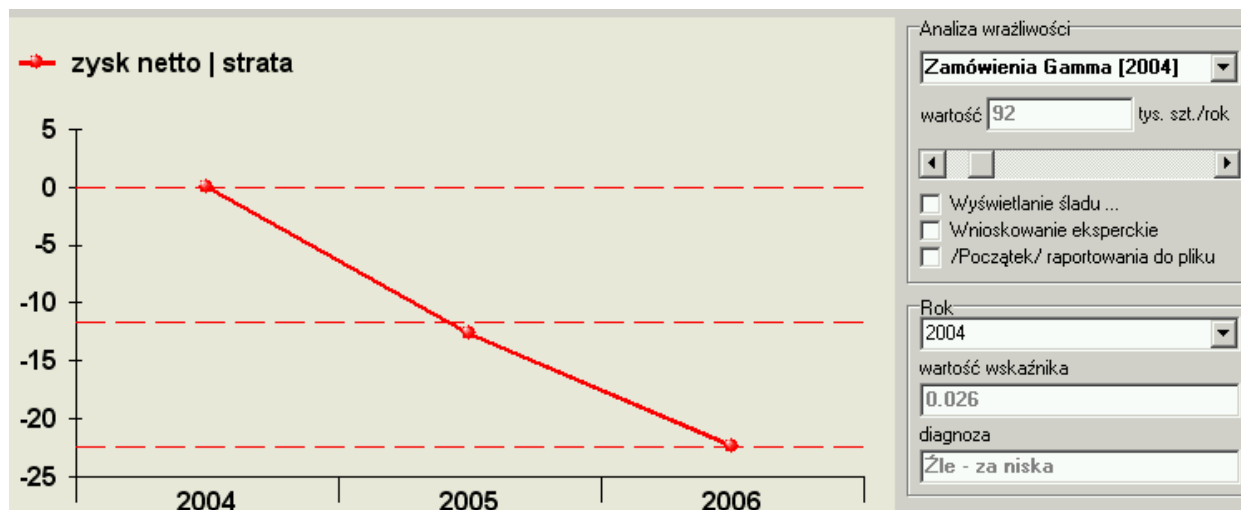
Końcowa długość kroku poszukiwań progu rentowności: 2 [tys. j. wyr./rok], stąd analogiczna dokładność ± 2 [tys. j. wyr./rok]



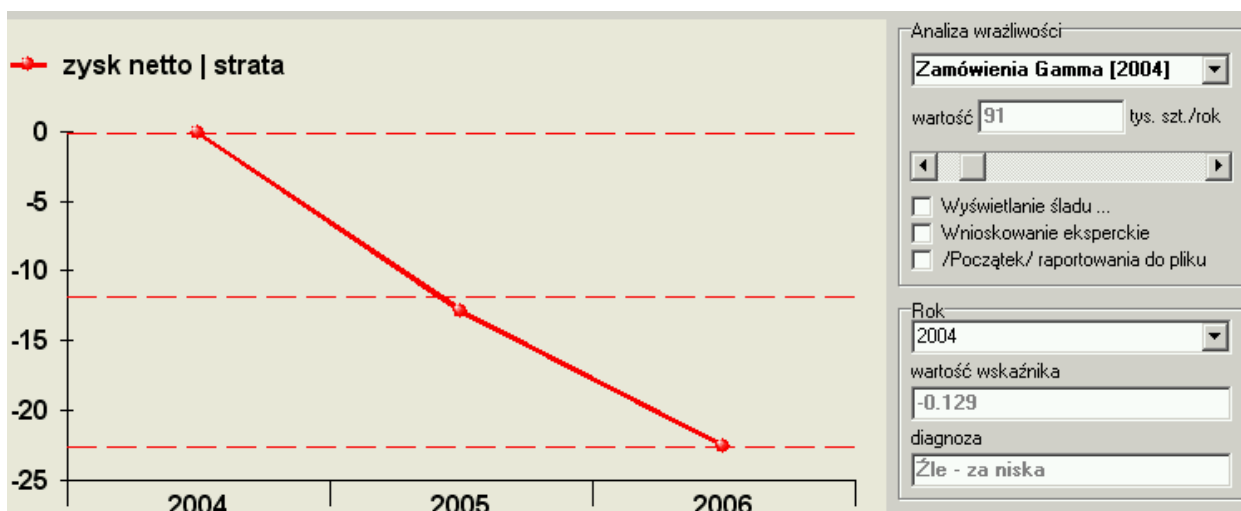
Rys. 8-6 Wykres progu rentowności (ilościowego, mierzonego wysokością zleceń produkcyjnych wyrobu Gamma) w funkcji zleceń produkcyjnych wyrobu Alfa i Beta

Interpretacja progu rentowności jest następująca: dla zamówień produkcyjnych wyrobu Alfa w wysokości 200 [tys. j. wyr./rok] i Beta w wysokości 200 [tys. j. wyr./rok] wg danych będących

podstawą do opracowania wykresu próg rentowności wynosi około 91 [tys. j. wyr./rok]. Teza ta zostanie zweryfikowana na podstawie wyników symulacji 2 zestawów zmiennych decyzyjnych, wykazujących odpowiednio wynik finansowy netto bliski zeru ale dodatni [Rys. 8-7] oraz bliski zeru ale ujemny [Rys. 8-8].



Rys. 8-7 Fragment interfejsu użytkownika w trakcie poszukiwania parametrów wielkości zamówień produkcyjnych wyrobu Gamma dla którego obserwowany jest próg rentowności; wartość dodatnia najbliższa zeru



Rys. 8-8 Fragment interfejsu użytkownika w trakcie poszukiwania parametrów wielkości zamówień produkcyjnych wyrobu Gamma dla którego obserwowany jest próg rentowności; wartość ujemna najbliższa zeru

Poniżej przedstawiono tekst źródłowy procedur w języku LEKS.

```

rem ---- szukanie miejsc zerowych -----
rem ---- wyzerowanie wartosci wszystkich zmiennych -----
rem ---- włączenie mozliwosci raportowania wartosci zmiennych -----
rem ---- uruchomienie ciaglego wykonywania programu -----
rem ---- ukrycie okna pogladu interpretatora programu -----

clear
raport
run
hide

rem ---- deklaracje zmiennych -----
deklaracja zerowe

```



```

deklaracja aktualny_zysk
deklaracja nowy_zysk
rem -----
deklaracja alfa_wartosc
deklaracja alfa_pocz
deklaracja alfa_kon
deklaracja alfa_krok
rem -----
deklaracja beta_wartosc
deklaracja beta_pocz
deklaracja beta_kon
deklaracja beta_krok
rem -----
deklaracja gamma_wartosc
deklaracja gamma_pocz
deklaracja gamma_kon
deklaracja gamma_krok
deklaracja pocz_gamma_krok
deklaracja elementarny_gamma_krok
deklaracja gamma_nowy
rem -----
deklaracja cena_alfa
deklaracja cena_beta
deklaracja cena_gamma
rem -----
deklaracja krok
deklaracja max
deklaracja rok_badany
deklaracja rok_do_oceny
deklaracja wartosc_zysku
deklaracja wezel_ojciec
deklaracja wezel_poczkowy
deklaracja wezel_potomek

rem ---- przypisanie zmiennym wartosci poczkowych -----
przypisz alfa_pocz, 0
przypisz alfa_wartosc, alfa_pocz
przypisz alfa_kon, 400
przypisz alfa_krok, 40
przypisz beta_pocz, 0
przypisz beta_wartosc, beta_pocz
przypisz beta_kon, 400
przypisz beta_krok, 40
przypisz gamma_pocz, 0
przypisz gamma_wartosc, gamma_pocz
przypisz gamma_kon, 800
przypisz pocz_gamma_krok, 20
przypisz gamma_krok, pocz_gamma_krok
przypisz elementarny_gamma_krok,2
rem -----
przypisz cena_alfa, 260
przypisz cena_beta, 395
przypisz cena_gamma, 470
rem -----
przypisz krok, 10
przypisz max, -1000
przypisz rok_do_oceny, 1
przypisz rok_badany, 1
rem -----
przypisz krok, 10
przypisz max, -1000

rem ---- ustalenie który wezel bedzie wezlem wzorcowym -----
odczyt węzeł wezel_poczkowy

```

```

odczyt ojciec wezel_początkowy, wezel_ojciec

przypisz beta_wartosc, beta_pocz
rem ---- zewnętrzna pętla ---- zmiany wartosci zamowien wyrobu beta ----
while <= beta_wartosc, beta_kon

przypisz alfa_wartosc, alfa_pocz
rem ---- wewnętrzna pętla ---- zmiany wartosci zamowien wyrobu alfa --
while <= alfa_wartosc, alfa_kon

przypisz gamma_wartosc, gamma_pocz

rem ---- utworzenie eksperymentalnego wezla -----
dodaj węzeł wezel_ojciec, wezel_początkowy, wezel_potomek
ustaw węzeł wezel_potomek

rem ---- zmiana wartosci podmacierzy zmiennych decyzyjnych -----
zmiana zlecenia alfa rok1 alfa_wartosc
zmiana zlecenia beta rok1 beta_wartosc
zmiana zlecenia gamma rok1 gamma_wartosc
zmiana cena alfa rok1 cena_alfa
zmiana cena beta rok1 cena_beta
zmiana cena gamma rok1 cena_gamma

rem ---- wykonanie eksperymentu symulacyjnego -----
symuluj węzeł wezel_potomek

rem ---- aktualny zysk sluzzy wyznaczeniu kierunku przejścia -----
rem ---- przez miejsce zerowe -----
odczyt bilans 5, 2, rok_do_oceny, aktualny_zysk

rem ---- wezel eksperymentalny już niepotrzebny -----
kasuj węzeł wezel_potomek

rem ---- szukanie miejsc zerowych w petli -----

przypisz zerowe, -1000
przypisz gamma_krok, pocz_gamma_krok

rem ---- wewnętrzna pętla, zmiany wartosci zamowien wyrobu gamma -
przypisz gamma_nowy, gamma_wartosc
while < gamma_nowy, gamma_kon
dodaj węzeł wezel_ojciec, wezel_początkowy, wezel_potomek
ustaw węzeł wezel_potomek

rem ---- zmiana wartosci podmacierzy zmiennych decyzyjnych ----
zmiana zlecenia alfa rok1 alfa_wartosc
zmiana zlecenia beta rok1 beta_wartosc
zmiana zlecenia gamma rok1 gamma_nowy
zmiana cena alfa rok1 cena_alfa
zmiana cena beta rok1 cena_beta
zmiana cena gamma rok1 cena_gamma

rem ---- wykonanie eksperymentu symulacyjnego -----
symuluj węzeł wezel_potomek
odczyt bilans 5, 2, rok_do_oceny, nowy_zysk
kasuj węzeł wezel_potomek

rem ---- sprawdzenie występowania miejsca zerowego -----
if > nowy_zysk, 0
if < aktualny_zysk, 0
if = gamma_krok, pocz_gamma_krok
rem ---- skrocenie dlugosci kroku (zwiększenie dokładności) -----

```

```

- gamma_nowy, gamma_nowy, gamma_krok
przypisz gamma_krok,elementarny_gamma_krok
przypisz nowy_zysk,aktualny_zysk
else
rem -- 'zerowe' jest srednia arytmetyczna dwu ostatnich pomiarow ---
przypisz zerowe, gamma_nowy
+ zerowe, zerowe, gamma_wartosc
/ zerowe, zerowe, 2
przypisz gamma_wartosc, gamma_kon
przypisz gamma_nowy, gamma_kon
endif
endif
else
if > aktualny_zysk, 0
if = gamma_krok,pocz_gamma_krok
rem ----- skrocenie dlugosci kroku (zwiekszenie dokladnosc) -----
- gamma_nowy, gamma_nowy, gamma_krok
przypisz gamma_krok,elementarny_gamma_krok
przypisz nowy_zysk,aktualny_zysk
else
rem -- 'zerowe' jest srednia arytmetyczna dwu ostatnich pomiarow ---
przypisz zerowe, gamma_nowy
+ zerowe, zerowe, gamma_wartosc
/ zerowe, zerowe, 2
przypisz gamma_wartosc, gamma_kon
przypisz gamma_nowy, gamma_kon
endif
endif
endif

rem ----- zapamietaj ostatnia wartosc gamma -----
przypisz gamma_wartosc, gamma_nowy

rem ----- inkrementacja zamowien gamma -----
if = zerowe, -1000
+ gamma_nowy, gamma_nowy, gamma_krok
endif

przypisz aktualny_zysk, nowy_zysk

endwhile

if = zerowe, -1000
if > nowy_zysk, 0
przypisz zerowe, 1000
rem ----- nie znaleziono nowych miejsc zerowych -----
rem ----- mimo to koniec petli -----
endif
endif

rem ----- zapis wartosci zmiennych do pliku -----
zanotuj alfa_wartosc
zanotuj beta_wartosc
zanotuj zerowe

+ alfa_wartosc, alfa_wartosc, alfa_krok
endwhile
+ beta_wartosc, beta_wartosc, beta_krok
endwhile

rem ----- koniec programu -----
stop all

```

8.5 Procedura optymalizacyjna w języku LEKS - metoda Hooke'a-Jeevesa

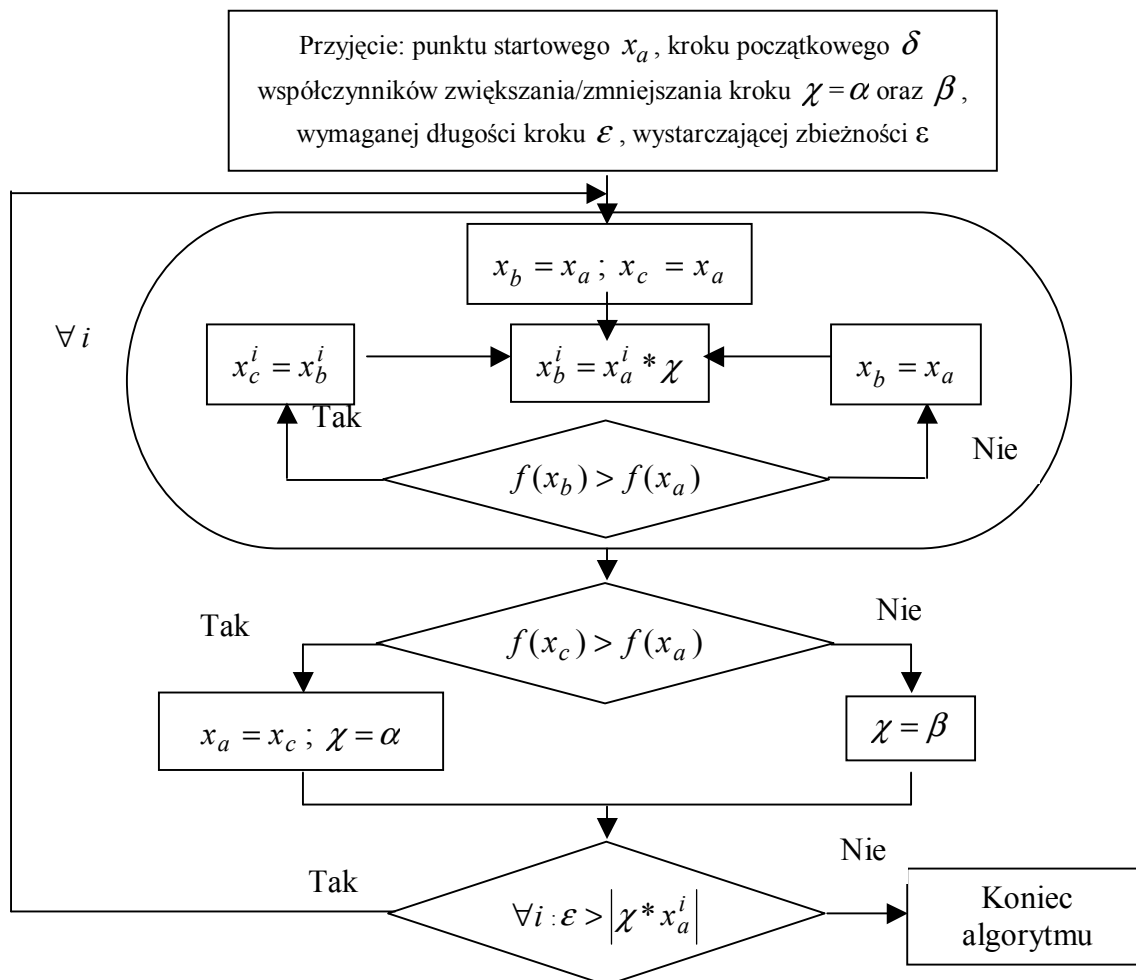
Sformułowanie **ramowego problemu badawczego**:

Opracować program w języku LEKS implementujący **optymalizacyjną metodę Hooke'a-Jeevesa**.

Do znanych **metod poszukiwania rozwiązań optymalnych** należy metoda **Hooke'a-Jeevesa** (tzw. metoda błędzenia) oraz metoda Rosenbrocka która jest jej modyfikacją. Wraz z metodą Nelder-Meada (sympleksem) należy ona do metod przeszukiwania prostego [200], które z kolei tworzą wraz z metodami kierunków poprawy (np. Gaussa-Seidela, najmniejszego spadku, Powella) grupę metod optymalizacji wielowymiarowej. Istotą metody Hooke'a-Jeevesa jest wykorzystywanie **informacji o rezultatach poprzedniego kroku** w taki sposób, że **jeśli krok jest korzystny to długość następnego kroku** (w tym samym kierunku) **jest odpowiednio zwiększana**, w **przeciwnym wypadku** długość kroku jest **zmniejszana**, a krok wykonywany jest w **kierunku przeciwnym**. W ogólnym przypadku kroki wykonywane są w **kierunkach ortogonalnych**. W metodzie Rosenbrocka stosuje się obrót układu współrzędnych po etapie próbnym, co pomaga w osiągnięciu szybszej zbieżności w szczególnych warunkach.

Algorytm metody Hooke'a-Jeevesa można opisać za pomocą następujących kroków [Rys. 8-9]:

1. Określenie **liczby osi układu współrzędnych** (kierunków w których będzie dokonywana optymalizacja) czyli optymalizowanych zmiennych, współrzędne **punktu startowego**, **warunek zakończenia algorytmu**: liczba iteracji oraz zbieżność (epsilon, ϵ), dobór **współczynników zwiększania/zmniejszania długości kroków**, obliczenie **funkcji celu** dla punktu startowego
2. Wykonanie **etapu próbnego** czyli elementarnych kroków względem punktu startowego we wszystkich kierunkach wzdłuż osi układu współrzędnych i obliczenie wartości funkcji celu dla wszystkich punktów próbnych
3. Wykonanie **etapu właściwego** czyli konstrukcja **kierunku optymalizacji** utworzonego przez **wektor udanych kroków** wykonanych w etapie próbnym. Wyznaczenie nowego punktu próbnego na podstawie poprzedniej długości kroku próbnego i **współczynnika zwiększania długości kroku** (większego od zera) i obliczenie wartości funkcji celu
4. W przypadku **zmniejszenia się wartości funkcji celu** (kroku niepomyślnego) zmniejszenie długości kroku elementarnego o współczynnik zmniejszania ze znakiem przeciwnym i powrót do kroku 2. W przeciwnym wypadku powrót do kroku 3
5. Zakończenie algorytmu jeśli długość kroku etapu próbnego jest mniejsza od zadanej (ϵ) lub wykonano zadaną liczbę kroków.



Rys. 8-9 Algorytm Hooke'a-Jeevesa. Wersja podstawowa

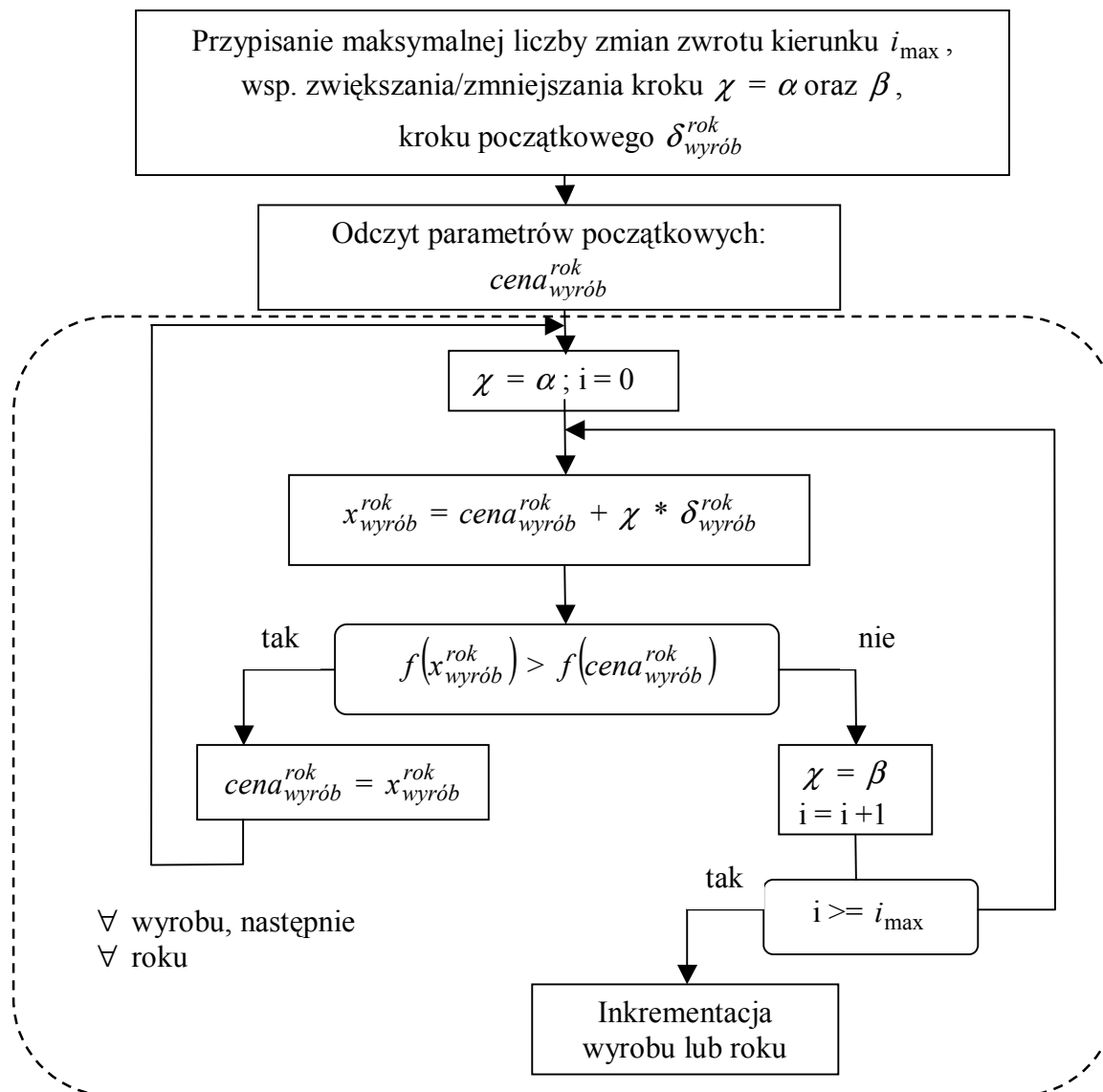
Konkretyzacja problemu badawczego.

Dla potrzeb niniejszej pracy metodę Hooke'a-Jeevesa **zmodyfikowano** [Rys. 8-10] w sposób zaprezentowany na przykładzie **zadania optymalizacji wartości podzbioru macierzy zmiennych decyzyjnych**. Optymalizacji podlegać będą ceny wyrobów gotowych w trzech kolejnych latach. Wymiary **podzbioru macierzy** podlegającego optymalizacji to [3(identyfikatory wyrobów)*3(lata)] czyli [3 x 3].

Zadanie optymalizacji **podzielono na trzy etapy** (kolejność przedmiotowa: ceny, zlecenia produkcyjne, zamówienia surowców). Funkcją celu będzie **zysk netto** w poszczególnych latach trzyletniej sekwencji. Oznacza to że zysk netto powinien mieć jak najwyższą wartość w każdym roku przy oczywistym założeniu że decydent **nie ma wpływu na decyzje podjęte w przeszłości**. Optymalizacja przeprowadzana jest kolejno w poszczególnych komórkach macierzy, stąd metoda rozwiązania zawiera opis strategii (konfiguracji) dotyczącej kolejności poszczególnych operacji procesu optymalizacji. Zastosowana zostanie strategia „najpierw pierwszy rok, kolejno wyroby np.: Alfa, Beta, Gamma którą symbolicznie zapisać można jako:

A1->B1->G1 lub A1B1G1-A2B2G2- A3B3G3.

Warunki początkowe tym razem zostaną przyjęte **na podstawie odczytanych parametrów aktywnego węzła** (aby eksperymentator mógł zintegrować metodę numeryczną z poszukiwaniem optimum za pośrednictwem interaktywnego graficznego interfejsu użytkownika). Długość kroku początkowego nadawana jest programowo (na podstawie doświadczenia) natomiast współczynniki $\alpha > 0$ i $\beta < 0$ przy czym $|\alpha| > |\beta|$.



Rys. 8-10 Schemat zmodyfikowanego algorytmu Hooke'a-Jeevesa

Modyfikacja standardowej procedury metody Hooke'a-Jeevesa polega zatem na:

- Rozszerzeniu etapu próbnego w taki sposób, aby każdy kierunek był optymalizowany od początku do końca czyli traktowany jako oddzielny problem optymalizacji; skutkiem tego wynik algorytmu może zależeć od zastosowanej strategii (kolejności optymalizowanych kierunków), co może być korzystne (lub nie), jeśli chodzi o szybkość osiągnięcia zbieżności
- Pominięciu etapu właściwego czyli konstrukcji wektora poszukiwań; zamiast tego pełnej optymalizacji podlega każdy kierunek
- Umożliwieniu wielokrotnego powtarzania procedury, gdyż wynik końcowy algorytmu może być traktowany jako punkt początkowy dla kolejnych iteracji pełnego algorytmu. Poszukiwane

rozwiązanie przechowywane jest w określonym węźle w drzewie decyzyjnym. Umożliwia to odczytanie parametrów początkowych dla kolejnego przebiegu algorytmu bezpośrednio z węzłów drzewa decyzyjnego

d) Zakończenie procedury optymalizacji danej zmiennej (kierunku) jest uwarunkowane liczbą następujących bezpośrednio po sobie niepomyślnych kroków. Wynika to z faktu, że wykonanie niepomyślnego kroku (tzn. zmniejszenie wartości funkcji celu) skutkuje zmianą zwrotu kierunku na przeciwny. Z drugiej strony wykonanie kroku pomyślnego powoduje wyzerowanie licznika kroków niepomyślnych.

Poniżej przedstawiono tekst źródłowy zmodyfikowanej metody Hooke'a-Jeevesa zapisany w języku LEKS. W programie założono, że optymalizacji podlegają ceny wyrobów Alfa i Gamma, czyli że cena wyrobu Beta jest niezmienna.

```
rem ----- konfiguracja eksperymentu -----
clear
raport
run
rem ----- deklaracje zmiennych (cen) -----
deklaracja alfa_wartosc
deklaracja alfa_pocz
deklaracja alfa_opt
deklaracja alfa_nowy
rem -----
deklaracja beta_wartosc
deklaracja beta_pocz
deklaracja beta_opt
deklaracja beta_nowy
rem -----
deklaracja gamma_wartosc
deklaracja gamma_pocz
deklaracja gamma_opt
deklaracja gamma_nowy
rem ----- deklaracje zmiennych sterujących przebiegiem eksperymentu -----
rem ----- i przechowujących wyniki -----
deklaracja aktualny_zysk
deklaracja nowy_zysk
deklaracja krok
deklaracja b_krok
deklaracja max
deklaracja rok_badany
deklaracja rok_do_oceny
deklaracja wyrob
deklaracja wartosc_zysku
deklaracja wezel_ojciec
deklaracja wezel_początkowy
deklaracja wezel_potomek
deklaracja licznik
deklaracja nr_wyrobu
deklaracja liczba_zmian

rem ----- początkowe wartości zmiennych sterujących przebiegiem eksperymentu -----
przypisz krok,10
przypisz max,-1000
przypisz rok_do_oceny,1
przypisz rok_badany,1
przypisz wyrob,1
```

```

przypisz licznik,0

rem ----- początek głównej pętli (dwukrotne powtarzanie algorytmu)
while < licznik,3
  pokaz licznik

  przypisz nr_wyrobu,1
rem ----- początek pętli (optymalizacja cen poszczególnych wyrobów)
while < nr_wyrobu, 4
  if = nr_wyrobu,2
    przypisz nr_wyrobu,3
  endif
  pokaz nr_wyrobu
  odczyt węzeł wezeł_początkowy
  odczyt ojciec wezeł_początkowy, wezeł_ojciec
  dodaj węzeł wezeł_ojciec,wezeł_początkowy,wezeł_potomek
  ustaw węzeł wezeł_potomek

  przypisz krok,10
  przypisz max,-1000

  odczyt węzeł wezeł_potomek

  if = nr_wyrobu,1
    odczyt zmienne decyzyjne 4, rok_badany, alfa_pocz
    przypisz alfa_wartosc, alfa_pocz
    przypisz alfa_nowy,alfa_wartosc
    if = rok_badany,1
      zmiana cena alfa rok1 alfa_wartosc
    endif
    if = rok_badany,2
      zmiana cena alfa rok2 alfa_wartosc
    endif
    if = rok_badany,3
      zmiana cena alfa rok3 alfa_wartosc
    endif
  endif
  if = nr_wyrobu,2
    odczyt zmienne decyzyjne 5, rok_badany, beta_pocz
    przypisz beta_wartosc, beta_pocz
    przypisz beta_nowy,beta_wartosc
    if = rok_badany,1
      zmiana cena beta rok1 beta_wartosc
    endif
    if = rok_badany,2
      zmiana cena beta rok2 beta_wartosc
    endif
    if = rok_badany,3
      zmiana cena beta rok3 beta_wartosc
    endif
  endif
  if = nr_wyrobu,3
    odczyt zmienne decyzyjne 6, rok_badany, gamma_pocz
    przypisz gamma_wartosc, gamma_pocz
    przypisz gamma_nowy,gamma_wartosc
    if = rok_badany,1
      zmiana cena gamma rok1 gamma_wartosc
    endif
    if = rok_badany,2
      zmiana cena gamma rok2 gamma_wartosc
    endif
    if = rok_badany,3
      zmiana cena gamma rok3 gamma_wartosc
    endif
  endif
endif

```



```
symuluj węzeł wezel_potomek
odczyt bilans 5,2,rok_do_oceny,aktualny_zysk
kasuj węzeł wezel_potomek
przypisz b_krok, krok
przypisz liczba_zmian, 0
```

```
while > b_krok, 1
  if = nr_wyrobu,1
    + alfa_nowy, alfa_wartosc,krok
  endif
  if = nr_wyrobu,2
    + beta_nowy, beta_wartosc,krok
  endif
  if = nr_wyrobu,3
    + gamma_nowy, gamma_wartosc,krok
  endif
```

```
dodaj węzeł wezel_ojciec,wezel_początkowy,wezel_potomek
ustaw węzeł wezel_potomek
```

```
if = nr_wyrobu,1
  if = rok_badany,1
    zmiana cena alfa rok1 alfa_nowy
  endif
  if = rok_badany,2
    zmiana cena alfa rok2 alfa_nowy
  endif
  if = rok_badany,3
    zmiana cena alfa rok3 alfa_nowy
  endif
endif
if = nr_wyrobu,2
  if = rok_badany,1
    zmiana cena beta rok1 beta_nowy
  endif
  if = rok_badany,2
    zmiana cena beta rok2 beta_nowy
  endif
  if = rok_badany,3
    zmiana cena beta rok3 beta_nowy
  endif
endif
if = nr_wyrobu,3
  if = rok_badany,1
    zmiana cena gamma rok1 gamma_nowy
  endif
  if = rok_badany,2
    zmiana cena gamma rok2 gamma_nowy
  endif
  if = rok_badany,3
    zmiana cena gamma rok3 gamma_nowy
  endif
endif
```

```
symuluj węzeł wezel_potomek
rem wartości 5,2 są współrzędnymi pozycji w bilansie
odczyt bilans 5,2,rok_do_oceny,nowy_zysk
kasuj węzeł wezel_potomek
```

```
if >= nowy_zysk, aktualny_zysk
  if = nr_wyrobu,1
    przypisz alfa_wartosc, alfa_nowy
  endif
  if = nr_wyrobu,2
```

```

    przypisz beta_wartosc, beta_nowy
endif
if = nr_wyrobu,3
    przypisz gamma_wartosc, gamma_nowy
endif
przypisz aktualny_zysk, nowy_zysk
* krok, krok, 3
else
* krok, krok, -0.5
endif

if > aktualny_zysk, max
    przypisz max, aktualny_zysk
    if = nr_wyrobu,1
        przypisz alfa_opt, alfa_wartosc
    endif
    if = nr_wyrobu,2
        przypisz beta_opt, beta_wartosc
    endif
    if = nr_wyrobu,3
        przypisz gamma_opt, gamma_wartosc
    endif
    przypisz liczba_zmian,0
else
    + liczba_zmian,liczba_zmian,1
endif

if >= krok,0
    * b_krok, krok, 1
else
    * b_krok, krok, -1
endif
if > liczba_zmian, 5
    przypisz b_krok,0.5
endif

endwhile

zanotuj alfa_opt

dodaj węzeł wezeł_ojciec,wezeł_początkowy,wezeł_potomek
ustaw węzeł wezeł_potomek

if = nr_wyrobu,1
    if = rok_badany,1
        zmiana cena alfa rok1 alfa_opt
    endif
    if = rok_badany,2
        zmiana cena alfa rok2 alfa_opt
    endif
    if = rok_badany,3
        zmiana cena alfa rok3 alfa_opt
    endif
endif
if = nr_wyrobu,2
    if = rok_badany,1
        zmiana cena beta rok1 beta_opt
    endif
    if = rok_badany,2
        zmiana cena beta rok2 beta_opt
    endif
    if = rok_badany,3
        zmiana cena beta rok3 beta_opt
    endif
endif
endif

```

```

if = nr_wyrobu,3
  if = rok_badany,1
    zmiana cena gamma rok1 gamma_opt
  endif
if = rok_badany,2
  zmiana cena gamma rok2 gamma_opt
endif
if = rok_badany,3
  zmiana cena gamma rok3 gamma_opt
endif
endif

symuluj węzeł wezel_potomek

przypisz wezel_poczkowy, wezel_potomek
+ nr_wyrobu,nr_wyrobu,1

endwhile
+ licznik,licznik,1
endwhile

stop all

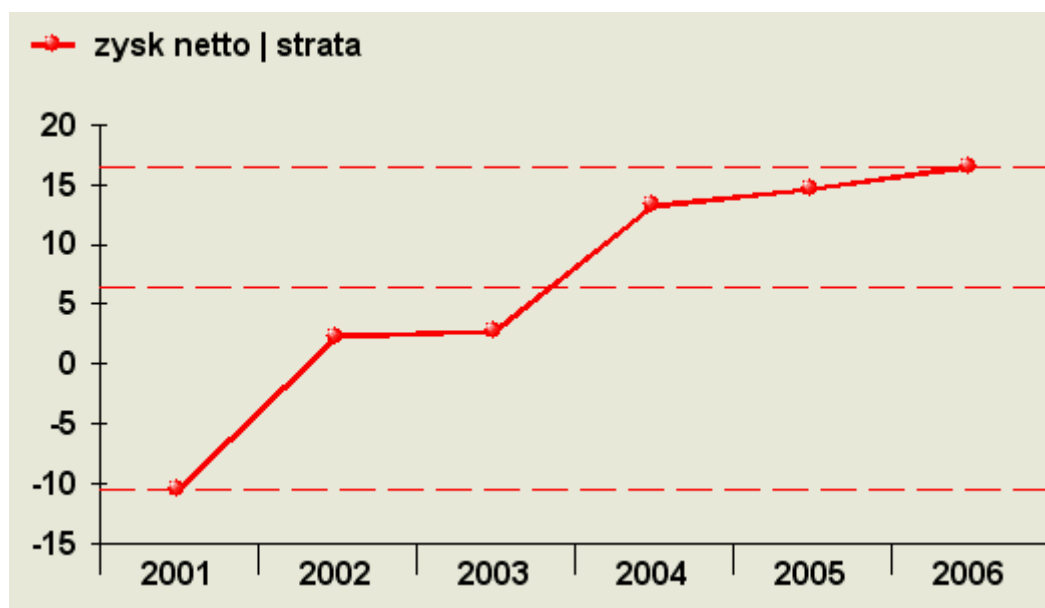
```

Wyniki eksperymentów przedstawiono w tabeli [Tab. 8-5] i na wykresie [Rys. 8-11]. Wskaźnik jakości w_j obliczany jest w następujący sposób:

$$w_j = \sum_{rok} \frac{\text{wynik finansowy}_{rok}}{\text{wynik finansowy zestawu bazowego}_{rok}} / \text{liczba lat}$$

Tab. 8-5 Wyniki zmodyfikowanego algorytmu Hooke'a-Jeevesa. Porównanie zestawu bazowego i optymalnego

Iteracja	Optymalne ceny wyrobu ALFA [zł]	Optymalne ceny wyrobu BETA [zł]	Optymalne ceny wyrobu GAMMA [zł]	Wynik finansowy [mln zł]	Wskaźnik jakości w_j
Zestaw bazowy	250.00	340.00	411.00	3,320	1
Rok1	238.44	375,08	449,75	13,16	7,191
Rok2	263.28	375,08	462,29	14,73	
Rok3	268,91	375,08	476,00	16,58	



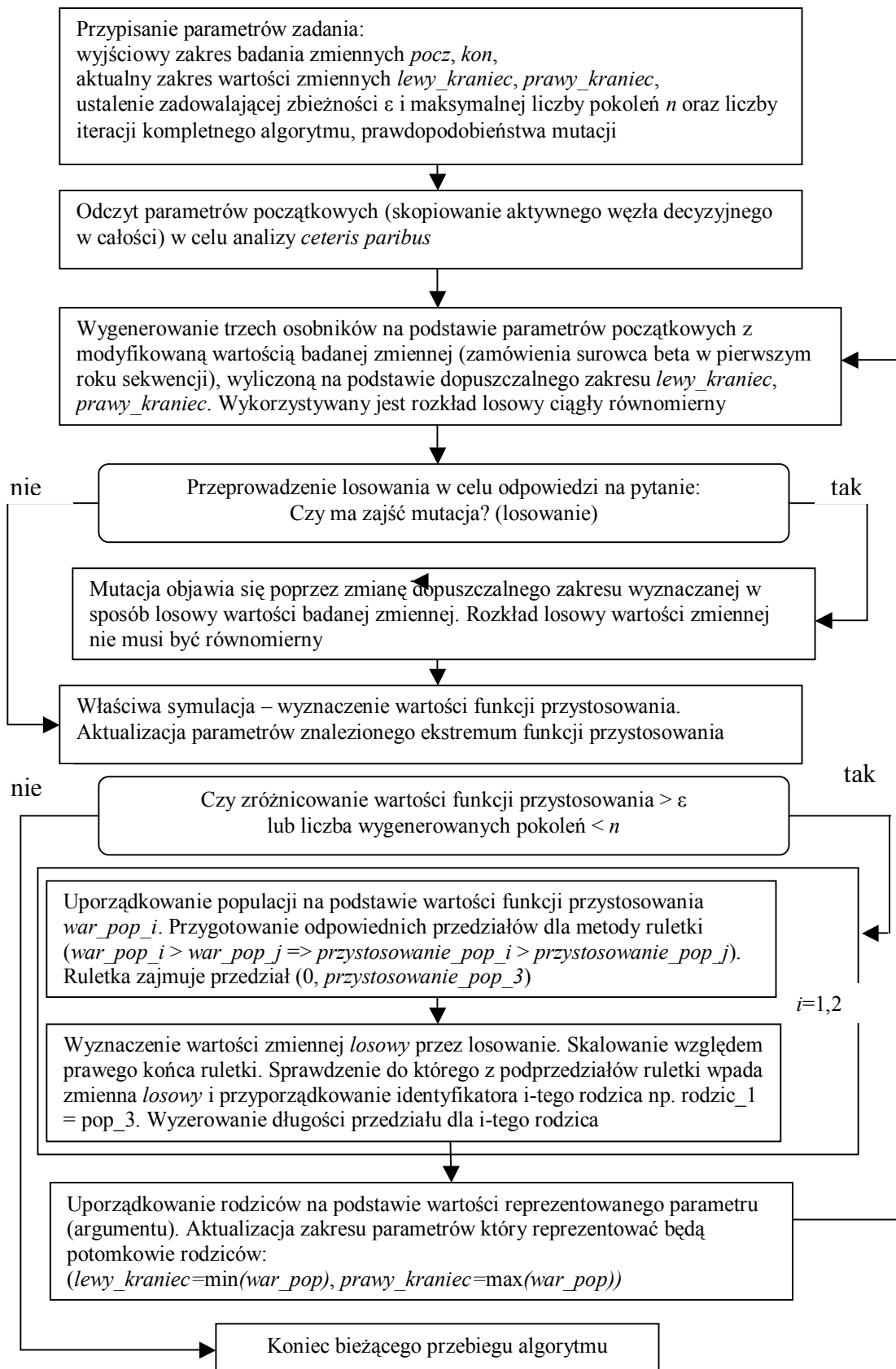
Rys. 8-11 Algorytm Hooke'a-Jeevesa. Wykres wyniku finansowego dla rozwiązania optymalnego

8.6 Procedura optymalizacyjna w języku LEKS – hybryda symulacji, języka sterowania eksperymentem i algorytmu genetycznego

Kolejną **techniką optymalizacyjną** którą postanowiłem **zaimplementować z języku LEKS** jest **algorytm genetyczny**. Algorytm genetyczny posługuje się stochastyczną zmiennością w celu symulacji wybranych modeli mechanizmów istniejących w świecie istot żywych: rozmnażania się i doboru naturalnego. Algorytmu genetycznego nie konstruuje się zazwyczaj w celu modelowania zjawisk z dziedziny biologii, a w celu optymalizacji wyrażeń (funkcji dostosowania / funkcji celu) zapisanych w terminologii modeli matematycznych, opisujących systemy całkowicie odmienne od organizmów biologicznych. Koncepcję algorytmów genetycznych omówiono w innym miejscu niniejszej pracy, tutaj jedynie przypomnieć należy, że podstawą zasadą algorytmu genetycznego jest przeszukiwanie kolejno generowanych populacji w celu dokonania (z określonym czynnikiem losowym) wyboru („dobrych”) osobników (reprezentantów) do rozmnażania się (czyli pojawienia się ich cech w kolejnej populacji). Proces rozmnażania się (powstania nowych osobników - genotypów), przez analogię biologiczną polega na rekombinacji genotypów (materiału genetycznego). Możliwość przeszukiwania całej przestrzeni cech (dziedziny funkcji celu) tzn. sprawdzenia wartości argumentów nieobecnych w danym pokoleniu jest zapewniona dzięki zjawisku (losowej) mutacji czyli występowania zmienności cech. **Zadaniem twórcy implementacji algorytmu genetycznego** jest ustawienie następujących parametrów: rozmiaru populacji, częstości rekombinacji, częstości mutacji. Należy także ustalić sposób kodowania osobników w populacji (reprezentujących konkurujące ze sobą rozwiązania problemu), sposób rekombinacji osobników (rozmnażania się) oraz działanie operatora mutacji. W dalszej części wyszczególniono etapy implementowanego algorytmu genetycznego oraz jedną z możliwych reprezentacji graficznych algorytmu w postaci bloków.

Przed opracowaniem prezentowanego poniżej algorytmu genetycznego przyjęto następujące **założenia wstępne**:

- reprezentacja genotypu zmiennoprzecinkowa
- operator krzyżowania, dwoje rodziców; efektem krzyżowania jest mieszaniec (nowy osobnik) którego cecha ma wartość pośrednią między wartością cechy rodziców
- metoda ruletki (losowania rodziców). Szerokość gniazd proporcjonalna do wartości funkcji przystosowania kandydata na rodzica
- rozmiar populacji: trzy osobniki
- dowolna liczba pokoleń
- warunek zatrzymania: założona liczba pokoleń lub "epsilon" wyliczany na podstawie zróżnicowania rodziców wybranych z aktualnej populacji



Rys. 8-12 Schemat blokowy algorytmu genetycznego zaimplementowanego w postaci programu w języku LEKS

Omawiana implementacja algorytmu genetycznego składa się z wyszczególnionych poniżej kroków [Rys. 8-12]. Jest to **sformułowanie ramowego problemu badawczego**:

1. Określenie parametrów początkowych i warunku zakończenia algorytmu
2. Wygenerowanie populacji początkowej (możliwość mutacji) oraz wyznaczenie wartości funkcji przystosowania
3. Uporządkowanie osobników w populacji na podstawie kryterium wartości funkcji przystosowania
4. Przyporządkowanie osobnikom przedziałów proporcjonalnych do wartości funkcji przystosowania w celu zastosowania metody ruletki
5. Wyznaczenie (przez losowanie metodą ruletki) kandydatów na rodziców osobników następnego pokolenia
6. Zadbanie o to, aby poszczególni rodzice charakteryzowali się niepowtarzalnością cech w danej populacji
7. Zaktualizowanie parametrów zakresu do którego należeć będą potomkowie rodziców
8. Wygenerowanie kolejnej populacji (możliwość mutacji) i wyznaczenie wartości funkcji przystosowania
9. Zaktualizowanie informacji o osobniku o najwyższej wartości funkcji przystosowania
10. W razie bardzo niskiego zróżnicowania w populacji wartości funkcji przystosowania - zakończenie algorytmu i powrót do punktu 3.

Poniżej przedstawiono **tekst źródłowy** implementacji algorytmu genetycznego zapisany w języku LEKS. W programie wykorzystano **szereg procedur** zapisywanych w odrębnych plikach. Całość programu napisano w konwencji modułów, aby ułatwić modyfikację i zmianę parametrów w poszczególnych wersjach algorytmu. Dlatego też prezentacja rozpocznie się od **najistotniejszych modułów**.

W programie głównym korzysta się z procedur: *generowanie_osobnika*, *odczyt_osobnikaX* oraz *obliczanie_min_max*

Zadaniem modułu *generowanie_osobnika* jest przypisanie (na podstawie rozkładu losowego i dopuszczalnego zakresu zmienności) wartości badanej zmiennej oraz opcjonalne uruchomienie możliwości mutacji tej zmiennej.

```
run
losuj losowy
- parametr,prawy_kraniec,lewy_kraniec
* parametr,losowy,parametr
+ parametr, parametr, lewy_kraniec
if = typ_badania,5
call mutacja.roz
```

```
endif
```

Zadaniem modułu *mutacja* jest przeprowadzenie mutacji. Uwarunkowane jest to jednak realizacją wartości zmiennej losowej w określonym przedziale (0,9;1), co jest równoważne wartości prawdopodobieństwa 0,1. Operator mutacji zadziała prawidłowo, jeśli zmodyfikowana dzięki niemu badana zmienna parametr nie wyjdzie poza dopuszczalny zakres (pocz, kon).

```
run
rem ustalamy czestosc mutacji
losuj czestosc_mutacji

if > czestosc_mutacji,0.9
  call kontrola_mutacji.roz
  - przesuniecie, kon, pocz
  * przesuniecie, przesuniecie, normalny_x
  rem zanotuj przesuniecie
  + parametr, parametr, przesuniecie
endif
```

Zadaniem modułu *kontrola_mutacji* jest wygenerowanie dopuszczalnej wartości zmiennej. Jeśli to się nie powiodło to próba jest powtarzana.

```
run
przypisz dobrze,0
while <> dobrze, 1
  call rozklad_normalny.roz
  - przesuniecie, kon, pocz
  * przesuniecie, przesuniecie, normalny_x
  + proba, parametr, przesuniecie
  rem zanotuj proba
  if <|> pocz,proba, kon
    przypisz dobrze,1
  endif
endwhile
```

Zadaniem modułu *rozklad_normalny* jest wygenerowanie wartości zgodnej z rozkładem normalnym o parametrach $N(0,1)$ metodą odwracania dystrybucyj podaną przez Kanha [340].

```
run
losuj rrr
```

```

+ normalny_licznik,1,rrr
- normalny_mianownik,1,rrr
/ normalny_ulamek,normalny_licznik,normalny_mianownik
rem wynik logarytmowania "b" zapisujemy w "a" gdy ln a,b
ln logarytm_normalny_ulamek,normalny_ulamek
* normalny_x, w_r_n, logarytm_normalny_ulamek
przypisz znak_w_rozkładzie_normalnym,-1
losuj znak
if > znak,0.5
  przypisz znak_w_rozkładzie_normalnym,1
endif
* normalny_x, znak, znak_w_rozkładzie_normalnym

```

Zadaniem modułu ***odczyt_osobnika1*** (ogólnie: *odczyt_osobnikaN*) jest przygotowanie do odczytu wartości ze sprawozdania finansowego a następnie (opcjonalnie) skorygowanie go w przypadku zastosowaniu metody z funkcją kary.

```

run
call problem_war_pop_1.roz
rem
przypisz korekta,0
if = typ_badania, 4
  call kara.roz
endif
if = typ_badania, 6
  call kara.roz
endif
+ war_pop_1, war_pop_1, korekta

```

Zadaniem modułu ***problem_war_pop_1*** jest bezpośrednio odczytanie pozycji sprawozdania finansowego. Jest to najmniejszy moduł jednak umożliwia szybką zmianę optymalizowanej zmiennej (szerzej: wyrażenia zbudowanego na podstawie wartości pozycji w sprawozdaniach finansowych).

```

run
rem odczyt zysku netto w pierwszym roku sekwencji
odczyt bilans 5,2,1,war_pop_1

```


Zadaniem modułu *obliczanie_min_max* jest wyznaczenie węzłów decyzyjnych o najmniejszej i największej wartości funkcji przystosowania.

```
run
rem obliczanie minimum i maksimum
przypisz min,1000
przypisz max,-1000
przypisz poz_min,0
przypisz poz_max,0
rem szukamy minimum
if < war_pop_1,min
  przypisz min,war_pop_1
  przypisz poz_min,pop_1
endif
if < war_pop_2,min
  przypisz min,war_pop_2
  przypisz poz_min,pop_2
endif
if < war_pop_3,min
  przypisz min,war_pop_3
  przypisz poz_min,pop_3
endif

rem szukamy maximum
if > war_pop_1,max
  przypisz max,war_pop_1
  przypisz poz_max,pop_1
endif
if > war_pop_2,max
  przypisz max,war_pop_2
  przypisz poz_max,pop_2
endif
if > war_pop_3,max
  przypisz max,war_pop_3
  przypisz poz_max,pop_3
endif
```

Przedstawienia wreszcie wymaga tekst źródłowy programu *ostateczny.roz* implementującego algorytm genetyczny. Tekst źródłowy jest okrojona częścią większego środowiska badawczego, dotyczącego różnych wariantów zastosowań algorytmów genetycznych i tradycyjnych metod optymalizacji. Na jego podstawie został przygotowany artykuł autora pracy [336].

clear
raport
run
show
deklaracja typ_badania
przypisz typ_badania, 5

rem 1: polowienie przedzialow, ogolne
rem 2: polowienie przedzialow, produkcyjne
rem 3: genetyczny bez mutacji, ogolne
rem 4: genetyczny bez mutacji, produkcyjne
rem 5: genetyczny z mutacją, ogolne
rem 6: genetyczny z mutacją, produkcyjne

deklaracja wezel_poczkowy
deklaracja wezel_ojciec
deklaracja wezel_potomek
deklaracja epsilon
deklaracja max_licznik
deklaracja licznik
deklaracja roznica
deklaracja nowy_kraniec
deklaracja optymalny_argument
deklaracja optymalna_wartosc
deklaracja pocz
deklaracja kon
deklaracja lewy_kraniec
deklaracja prawy_kraniec
deklaracja max
deklaracja min
deklaracja losowy
deklaracja korekta
deklaracja dodatnia_roznica
deklaracja indeks
deklaracja max_indeks

deklaracja p1
deklaracja p2
deklaracja p3
deklaracja arg_max
deklaracja war_max
deklaracja znacznik

przypisz znacznik,-1000

```
if = typ_badania,5
  call deklaracje_dla_mutacji.roz
  deklaracja pop_1
  deklaracja pop_2
  deklaracja pop_3
  deklaracja war_pop_1
  deklaracja war_pop_2
  deklaracja war_pop_3
  deklaracja przystosowanie_pop_1
  deklaracja przystosowanie_pop_2
  deklaracja przystosowanie_pop_3
endif
```

```
if = typ_badania,3
  call deklaracje_dla_mutacji.roz
  deklaracja pop_1
  deklaracja pop_2
  deklaracja pop_3
  deklaracja war_pop_1
  deklaracja war_pop_2
  deklaracja war_pop_3
  deklaracja przystosowanie_pop_1
  deklaracja przystosowanie_pop_2
  deklaracja przystosowanie_pop_3
endif
```

```
deklaracja poz_max
deklaracja poz_min
deklaracja parametr
deklaracja odleglosc
deklaracja rodzic_1
deklaracja rodzic_2
```

```
rem ----- parametry problemu optymalizacyjnego -----
```

```
przypisz epsilon,0.000005
rem ----- max_licznik wraz z epsilon dotyczy pętli (warunku stopu) we wszystkich algorytmach -----
przypisz max_licznik,30
rem ----- max_indeks dotyczy pętli dla alg. gen. w programie głównym -----
przypisz max_indeks,10
rem ----- -----
```

```

odczyt węzeł wezel_początkowy
odczyt ojciec wezel_początkowy, wezel_ojciec

if > typ_badania, 2
  przypisz indeks,1

rem ----- pętla kolejnych eksperymentów (dla potrzeb badań) -----
while <= indeks, max_indeks

  dodaj węzeł wezel_ojciec,wezel_początkowy,wezel_potomek

  rem ----- warunki początkowe -----
  przypisz licznik,0
  przypisz pocz,0
  przypisz kon,400
  przypisz lewy_kraniec,pocz
  przypisz prawy_kraniec,kon

  rem ----- pierwsza, robocza populacja -----

  dodaj węzeł wezel_ojciec,wezel_potomek,pop_1
  dodaj węzeł wezel_ojciec,wezel_potomek,pop_2
  dodaj węzeł wezel_ojciec,wezel_potomek,pop_3

  rem ----- wygenerowanie osobnika nr 1 -----
  call generowanie_osobnika.roz
  ustaw węzeł pop_1
  zmiana zamówienia surowców black rok1 parametr
  przypisz p1,parametr
  symuluj węzeł pop_1
  call odczyt_osobnika1.roz

  rem ----- wygenerowanie osobnika nr 2 -----
  call generowanie_osobnika.roz
  if > typ_badania, 4
    while = p1, parametr
      call generowanie_osobnika.roz
    endwhile

```

```

endif
ustaw węzeł pop_2
zmiana zamówienia surowców black rok1 parametr
przypisz p2,parametr
symuluj węzeł pop_2
call odczyt_osobnika2.roz

rem ----- wygenerowanie osobnika nr 3 -----
call generowanie_osobnika.roz
if > typ_badania, 4
  while = p1, parametr
    call generowanie_osobnika.roz
  endwhile
  while = p2, parametr
    call generowanie_osobnika.roz
  endwhile
endif
ustaw węzeł pop_3
zmiana zamówienia surowców black rok1 parametr
przypisz p3,parametr
symuluj węzeł pop_3
call odczyt_osobnika3.roz

rem ----- sortowanie wg wartości funkcji przystosowania -----
przypisz war_max,war_pop_1
przypisz arg_max,p1

if > war_pop_2,war_max
  przypisz war_max,war_pop_2
  przypisz arg_max,p2
endif
if > war_pop_3,war_max
  przypisz war_max,war_pop_3
  przypisz arg_max,p3
endif

call obliczanie_min_max.roz
rem ----- zmienne min i max zawierają ekstremalne wartości funkcji -----
rem ----- przystosowania -----
rem ----- zmienne poz_min i poz_max zawierają numery osobników -----
rem ----- z ekstremalnymi wartościami funkcji przystosowania -----

```

```

rem ----- jesli długość przedziału między max a min -----
rem ----- jest mniejsza niz epsilon to koniec -----
- odleglosc,max,min

if < odleglosc, 0
  * odleglosc, odleglosc, -1
endif

if < odleglosc,epsilon
  przypisz licznik, max_licznik
endif

if = max_licznik, licznik
  rem pokaz licznik
endif

rem ----- pętla generowania kolejnych pokoleń -----
while < licznik, max_licznik

  rem ----- obliczanie szerokosci przedzialow w ruletce -----
  rem ----- dla poszczegolnych osobnikow -----

  - przystosowanie_pop_1,var_pop_1,min
  / przystosowanie_pop_1,przystosowanie_pop_1,odleglosc

  - przystosowanie_pop_2,var_pop_2,min
  / przystosowanie_pop_2,przystosowanie_pop_2,odleglosc
  + przystosowanie_pop_2,przystosowanie_pop_2,przystosowanie_pop_1

  - przystosowanie_pop_3,var_pop_3,min
  / przystosowanie_pop_3,przystosowanie_pop_3,odleglosc
  + przystosowanie_pop_3,przystosowanie_pop_3,przystosowanie_pop_2

rem ----- wyznaczenie kto bedzie pierwszym rodzicem -----

losuj losowy
* losowy,losowy,przystosowanie_pop_3

rem ----- czy rodzicem nr 1 będzie trzeci osobnik?
rem ----- jesli tak, to bedzie miał potem szerokosc 0 w ruletce
rem ----- gdyż w liczniku wyrażenia przystosowanie_pop_1 min-min=0
if <|> losowy, przystosowanie_pop_2,przystosowanie_pop_3
  przypisz rodzic_1,pop_3

```

```

    przypisz war_pop_3,min
endif

rem ----- czy rodzicem nr 1 będzie drugi osobnik?
rem ----- jesli tak, to bedzie miał potem szerokosc 0 w ruletce
if <|> losowy, przystosowanie_pop_1,przystosowanie_pop_2
    przypisz rodzic_1,pop_2
    przypisz war_pop_2,min
endif

rem ----- czy rodzicem nr 1 będzie pierwszy osobnik?
rem ----- jesli tak, to bedzie miał potem szerokosc 0 w ruletce
if <|> losowy, 0,przystosowanie_pop_1
    przypisz rodzic_1,pop_1
    przypisz war_pop_1,min
endif

rem ----- obliczanie szerokosci przedzialow dla poszczegolnych osobnikow -----
- przystosowanie_pop_1,war_pop_1,min
/ przystosowanie_pop_1,przystosowanie_pop_1,odleglosc

- przystosowanie_pop_2,war_pop_2,min
/ przystosowanie_pop_2,przystosowanie_pop_2,odleglosc
+ przystosowanie_pop_2,przystosowanie_pop_2,przystosowanie_pop_1

- przystosowanie_pop_3,war_pop_3,min
/ przystosowanie_pop_3,przystosowanie_pop_3,odleglosc
+ przystosowanie_pop_3,przystosowanie_pop_3,przystosowanie_pop_2

rem ----- wyznaczenie kto bedzie drugim rodzicem
losuj losowy
* losowy,losowy,przystosowanie_pop_3

rem ----- czy rodzicem nr 2 będzie trzeci osobnik?
if <|> losowy, przystosowanie_pop_2,przystosowanie_pop_3
    przypisz rodzic_2,pop_3
endif

rem ----- czy rodzicem nr 2 będzie drugi osobnik?
if <|> losowy, przystosowanie_pop_1,przystosowanie_pop_2
    przypisz rodzic_2,pop_2
endif

```

rem ---- czy rodzicem nr 2 będzie pierwszy osobnik?

if <|> losowy, 0, przystosowanie_pop_1

przypisz rodzic_2, pop_1

endif

ustaw węzeł rodzic_1

odczyt zmienne decyzyjne 4,1, lewy_kraniec

ustaw węzeł rodzic_2

odczyt zmienne decyzyjne 4,1, prawy_kraniec

rem początek nowego przedziału musi być mniejszy od końca

if > lewy_kraniec, prawy_kraniec

ustaw węzeł rodzic_1

odczyt zmienne decyzyjne 4,1, prawy_kraniec

ustaw węzeł rodzic_2

odczyt zmienne decyzyjne 4,1, lewy_kraniec

endif

kasuj węzeł pop_1

kasuj węzeł pop_2

kasuj węzeł pop_3

dodaj węzeł wezel_ojciec, wezel_potomek, pop_1

dodaj węzeł wezel_ojciec, wezel_potomek, pop_2

dodaj węzeł wezel_ojciec, wezel_potomek, pop_3

rem ---- wygenerowanie osobnika nr 1

call generowanie_osobnika.roz

ustaw węzeł pop_1

zmiana zamówienia surowców black rok1 parametr

przypisz p1, parametr

symuluj węzeł pop_1

call odczyt_osobnika1.roz

rem ---- wygenerowanie osobnika nr 2

call generowanie_osobnika.roz

if > typ_badania, 4

while = p1, parametr

call generowanie_osobnika.roz

endwhile

endif

ustaw węzeł pop_2


```

rem pokaz parametr
zmiana zamówienia surowców black rok1 parametr
przypisz p2,parametr
symuluj węzeł pop_2
call odczyt_osobnika2.roz

rem ----- wygenerowanie osobnika nr 3 -----
call generowanie_osobnika.roz
if > typ_badania, 4
  while = p1, parametr
    call generowanie_osobnika.roz
  endwhile
  while = p2, parametr
    call generowanie_osobnika.roz
  endwhile
endif
ustaw węzeł pop_3
rem pokaz parametr
przypisz p3,parametr
zmiana zamówienia surowców black rok1 parametr
symuluj węzeł pop_3
call odczyt_osobnika3.roz

call obliczanie_min_max.roz
rem ----- zmienne min i max zawierają ekstremalne wartości funkcji -----
rem ----- przystosowania -----
rem ----- zmienne poz_min i poz_max zawierają numery osobników -----
rem ----- z ekstremalnymi wartościami funkcji przystosowania -----
rem ----- jeśli długość przedziału między max a min -----
rem ----- jest mniejsza niż epsilon to koniec -----
- odleglosc,max,min

rem ----- licznik pokoleń
+ licznik,licznik,1

if < odleglosc,epsilon
  przypisz licznik,max_licznik
endif

if > war_pop_1,war_max
  przypisz war_max,war_pop_1
  przypisz arg_max,p1
endif

```

```

if > war_pop_2,war_max
  przypisz war_max,war_pop_2
  przypisz arg_max,p2
endif

if > war_pop_3,war_max
  przypisz war_max,war_pop_3
  przypisz arg_max,p3
endif

rem zanotuj arg_max
rem zanotuj war_max

endwhile

rem ----- koniec pętli generowania kolejnych pokoleń -----

kasuj węzeł pop_1
kasuj węzeł pop_2
kasuj węzeł pop_3

ustaw węzeł wezel_potomek
zmiana zamówienia surowców black rok1 arg_max

symuluj węzeł wezel_potomek
odczyt bilans 5,2,1,optymalna_wartosc
rem ----- zapisanie do pliku wartości argumentu dla którego
rem ----- obserwuje się ekstremum
zanotuj arg_max
zanotuj optymalna_wartosc
+ indeks,indeks,1

endwhile
rem ----- koniec pętli kolejnych eksperymentów (dla potrzeb badań) -----

endif

stop all

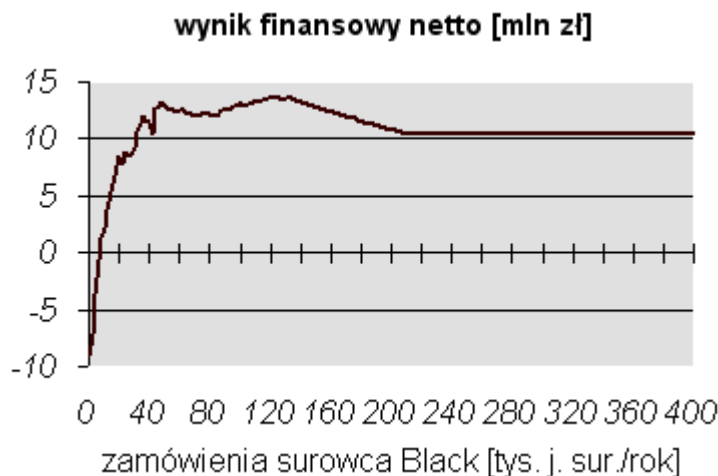
```

Konkretyzacja programu badawczego.

Działanie zaimplementowanego algorytmu genetycznego przedstawione zostanie na przykładzie zadania optymalizacji wielkości zamówień jednego z surowców niezbędnych do produkcji. Warunki początkowe (odczytane z aktywnego węzła, tzn. punkt startowy został wyznaczony za pomocą graficznego interfejsu użytkownika systemu wspomaganego decyzji) tym razem były następujące:

- cena wyrobu Alfa [zł/jedn.] = 260
- cena wyrobu Beta [zł/jedn.] = 395
- cena wyrobu Gamma [zł/jedn.] = 470
- zamówienia surowca White [tys. j. sur. / rok] = 219
- $0 \leq$ zamówienia surowca Black [tys. j. sur. / rok] ≤ 400
- zamówienia surowca Red [tys. j. sur. / rok] = 177

Optymalizacji podlegać będzie wielkość zamówienia surowca Black w pierwszym roku sekwencji a funkcją celu zysk netto na koniec tego roku. Jak widać na rysunku przedstawiającym przebieg wartości [Rys. 8-13] funkcji celu, nie jest łatwo wyznaczyć w jej przypadku wartość ekstremalną. Badany zakres zmiennej parametr to (0, 400), maksymalna liczba pokoleń jest równa 30, warunek zbieżności dotyczy $\varepsilon = 0,000005$. Eksperyment (pełny przebieg algorytmu genetycznego) wykonano wielokrotnie (10 razy), badano także zachowanie algorytmu dla przypadku bez operatora mutacji.



Rys. 8-13 Przebieg wartości zmiennej zależnej wynik finansowy netto w funkcji wielkości zamówień surowca Black (pozostałe parametry są stałe)

Poniżej przedstawiono wyniki eksperymentów [Tab. 8-6], [Tab. 8-7] w przypadku algorytmu z mutacją oraz bez mutacji, w odpowiednich jednostkach: argumenty w tys. j. surowca / rok i wynik w milionach zł. W kolejnych wierszach przedstawiono iteracje algorytmu genetycznego uporządkowane na podstawie wartości wyniku finansowego netto.

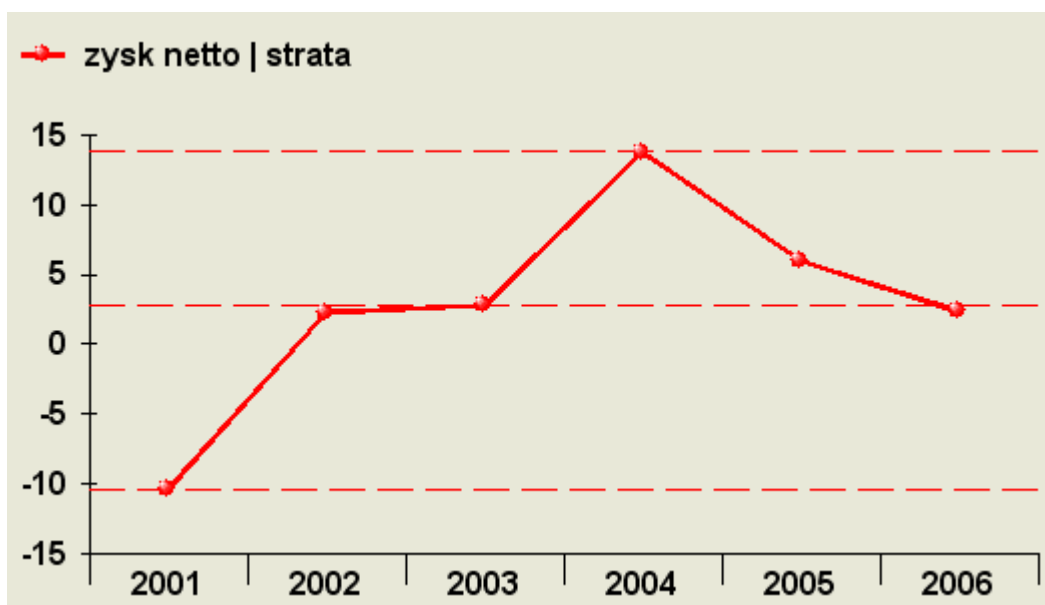
Tab. 8-6 Wyniki 10 iteracji algorytmu genetycznego. Po lewej stronie wersja z operatorem mutacji, po prawej bez operatora mutacji

Arg	max	arg	max
126,7375	13,77259	130,1875	13,64005
128,325	13,63191	132,2375	13,561
115,9625	13,58679	111,775	13,44189
118	13,54483	122,2875	13,42614
132,7625	13,52403	109,35	13,37235
142,4625	13,07061	52,1625	12,83689
103,3375	13,03859	63,5375	12,35516
237	10,33836	76,775	12,30261
285,125	10,33836	181,3375	11,49733
315,875	10,33836	241,925	10,33836

Tab. 8-7 Przetworzone wyniki otrzymane z wykorzystaniem algorytmu genetycznego

Wyszczególnienie	Alg. genetyczny z operatorem mutacji	Alg. genetyczny bez operatora mutacji
Absolutnie optymalna wartość parametru	126,74	130,19
Absolutnie optymalna wartość funkcji celu	13,77	13,64
Srednia optymalna wartość funkcji	12,52	12,68
Minimalna optymalna wartość funkcji celu	10,35	10,34
Wariancja optymalnej funkcji celu	2,32	1,16

Jak widać z powyżej przedstawionych obliczeń i tabeli [Tab. 8-7], wyniki są porównywalne. W przypadku wariantu z operatorem mutacji znaleziono lepsze rozwiązanie. Poniżej przedstawiono przebieg [Rys. 8-14] wyniku finansowego netto dla optymalnej wielkości zamówień surowca. Jak łatwo się przekonać maksymalny dostępny wynik finansowy netto w roku 2004 nie gwarantuje jeszcze zadowalających jego wartości w następnych latach, stąd warto byłoby rozpatrzyć uruchomienie algorytmu także dla następnych lat, być może przy założeniu optymalnej wartości parametru jako nowego, przesuniętego wprzód na skali czasu punktu początkowego.



Rys. 8-14 Przebieg wyniku finansowego dla optymalnej wielkości zamówień surowca Black w roku 2004

9 Dodatki

9.1 Informacje o wybranych ośrodkach uniwersyteckich prowadzące badania nad hybrydowymi systemami wspomaganie decyzji

- Ackoff Center for Advancement of Systems Approaches, Penn University of Pennsylvania, USA
- School of Computing and Technology, University of Sunderland, Wielka Brytania
- Knowledge Engineering & Discovery Research Institute, Auckland University of Technology, Nowa Zelandia
- PATH Modeling and Evaluation Group (MEG), California Department of Transportation (Caltrans) and University of California, USA
- L.A.A.S -C.N.R.S, "Laboratoire d'Analyse et d'Architecture des Systemes", Laboratory of Automation and Robotics (LAAR) of the University of Patras, Grecja
- Information Science, Department of Computer and Information Sciences of the University of Strathclyde, in Glasgow, Wielka Brytania
- School of Mathematical and Physical Sciences, Murdoch University, Zachodnia Australia
- Department of Computing, Imperial College, London, Wielka Brytania
- Departamento de Informatica, Inteligencia Artificial y Electronica, Facultad de Informatica, Universidad de Murcia, Hiszpania
- MTA-SZTAKI, Computer and Automation Research Institute of the Hungarian Academy of Sciences, Węgry
- The Research Group Data Bionics, Department of Mathematics and Computer Science, Philipps-University Marburg, Niemcy
- University of Sunderland, Hybrid Intelligent Systems Research Group, <http://www.his.sunderland.ac.uk/>, Wielka Brytania
- IHE Delft i STOWA (Stichting Toegepaste Onderzoek Waterbeheer), <http://www.stowa-nn.ihe.nl/FrameSet.htm> Use of Artificial Neural Networks and Fuzzy Logic for Integrated Water Management, Holandia

9.2 Wybrane konferencje dotyczące systemów hybrydowych

- IPMU, 9th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems July, 1-5, 2002, Annecy – France, <http://www.univ-savoie.fr/ipmu2002>
- 6th International Workshop, Fuzzy-Neuro Systems '99, German Society of Computer Science GI (Gesellschaft für Informatik e. V.) and Universität Leipzig, March 18 - 19, 1999
- AAAI 1999 Spring Symposium. Hybrid Systems and AI: Modeling, Analysis and Control of Discrete + Continuous Systems. March 22-24, 1999. Stanford University, California; Uwagi: tematem sympozjum było symulacyjne modelowanie hybrydowe (integrujące podejście systemodynamiczne i dyskretne)
- 13th European Simulation Multiconference, Warsaw, Poland, June 1-4, 1999, Track A6. Decision Processes in Management
- World Automation Congress 2004, SEVILLE, Spain, Dates: June 28 - July 1, 2004
- International Conference on Computational Intelligence for Modelling, Control and Automation, 12-14 July 2004 Gold Coast, Australia
- HIS 01, International Workshop on Hybrid Intelligent Systems, Adelaide, Australia, 2001
- EUNITE 2001: European Symposium on Intelligent Technologies, Hybrid Systems and their implementation on Smart Adaptive Systems, 13 - 14 December 2001, Tenerife, Spain
- EUFIT - The European Congress on Intelligent Techniques and Soft Computing annually organized by ELITE Foundation (European Laboratory for Intelligent Techniques Engineering), 1994 - 1999
- Wybrane książki omawiające systemy hybrydowe:
- Intelligent Hybrid Systems, By Suran Goonatilake, Sukhdev Khebbal John Wiley & Son Ltd, 1995
- Wermter and R. Sun, (eds.) Hybrid Neural Systems. Springer-Verlag, Heidelberg. 2000
- Decision Support Systems & Intelligent Systems, By Efraim Turban, Jay E. Aronson, Prentice Hall November 2000
- Intelligent Hybrid Systems: Fuzzy Logic, Neural Networks, & Genetic Algorithms, By Da Ruan, Da Ruan, Kluwer Academic Publishers 1997
- Fuzzy Logic and Expert Systems Applications, by Cornelius T. Leondes (Editor), Academic Press; 1st edition (January 15, 1998)
- Hybrid Intelligent System Applications, By Jay Liebowitz, Cognizant Communication Corp 1996

9.3 Wybrana literatura poruszająca zagadnienie hybrydowości

- Larry R. Medsker, Hybrid Intelligent Systems, Kluwer Academic Publishers 1995
- Ajith Abraham, Mario Koeppen, Hybrid Information Systems, Physica Verlag 2002
- Abraham Kandel, Gideon Langholz, Hybrid Architectures for Intelligent Systems, CRC Press 1992
- Suran Goonatilake, Sukhdev Khebbal, Intelligent Hybrid Systems, John Wiley & Sons Ltd 1995
- Yan-Qing Zhang, Abraham Kandel, Compensatory Genetic Fuzzy Neural Networks & Their Applications, World Scientific Pub Co. 1998
- Frank Hoffmann, Can Isik, Volker Zacharias, Takeshi Furuhashi, Learning & Adaptation in Fuzzy Control, John Wiley & Sons 2002
- Peter Jackson, Introduction to Expert Systems, Addison Wesley 1990
- Kurt Fedra, E. Weigkrecht, A hybrid approach to information and decision support systems: Hazardous substances and industrial risk management, Laxenburg, Pergamon Books Ltd. 1987
- Suran Goonatilake, Philip Treleaven, Intelligent Systems for Finance & Business, John Wiley & Sons 1995
- Jerzy Kisielnicki, Henryk Sroka, Systemy informacyjne biznesu. Informatyka dla zarządzania. Metody projektowania i wdrażania systemów, PLACET, Warszawa 1999
- Boris Kovalerchuk and Evgenii Vityaev, Data Mining in Finance: Advances in Relational and Hybrid Methods, Kluwer Academic Publishers. 2000

Wykaz materiałów dostępnych w sieci internet: <http://www.cecs.missouri.edu/~rsun/hybrid-sites.html>

9.4 Ewolucja systemu Ekanwin

Data	Program	Autorzy	Efekty badawcze		
1985	DEF	E.Radosiński	Symulator przedsiębiorstwa przemysłowego.		
1990		A.Sip	Symulator przedsiębiorstwa - spółka skarbu państwa.		
1994			Dostosowanie generowanych sprawozdań finansowych do wymogów ustawy o rachunkowości.		
1998 XI	Ek_An_		Przekazanie kodu źródłowego przez E.Radosińskiego. Analiza kodu źródłowego i studium wykonalności translacji ze środowiska DOS i dotychczasowego, autorskiego interfejsu użytkownika w Turbo Pascalu do wymagań środowiska MS Windows i Borland Delphi.		
1999 VI	Ekanwin	J. Zabawa	Implementacja pierwszego działającego prototypu na platformie MS Windows: operacje wejścia-wyjścia, wprowadzanie zmiennych decyzyjnych, interaktywna konfiguracja eksperymentu, symulacja i wyświetlanie sprawozdań finansowych.		
1999 IX			Artykuł - Interface użytkownika w inteligentnych hybrydowych systemach wspomaganie decyzji. Symulacja systemów gospodarczych Antałówka'99, Warszawa 1999, WSPiZ i PWr		
2000 VI			Nowe menu, wielopoziomowe drzewo eksperymentów, przyciski ekranowe.		
2000 IX			Artykuł (z E.Radosińskim): Koncepcja hybrydy inteligentnego systemu wspomaganie decyzji z symulatorem przedsiębiorstwa przemysłowego, Symulacja systemów gospodarczych Antałówka'2000, Warszawa 2000, WSPiZ i PWr		
2001 II			Pierwsza implementacja hybrydy symulacja-system ekspercki typu shell integracja modułu wizualizacji (pierwsza wersja).		
2001 III	Ek_An_	A. Górski	Udostępnienie kodu źródłowego środowiska Ekanwin A. Górskiemu, który opracował na tej podstawie m.in. własne hybrydy Ek_An_AG_, Ek_An_SE, Ek_An_SE_AG_. Efektem prac jest uzyskanie stopnia doktora w 2004 r. Jest to inna gałąź rozwoju Ekanwin		
2001 IV	Ekanwin	J. Zabawa	Modularyzacja symulatora – pierwsze procedury optymalizacyjne, jeszcze w wersji parametryzowanych procedur w Delphi (kompilowane).		
2001 VI			Pierwsza wersja języka planowania eksperymentu LEKS. Wbudowanie architektury tablicowej do systemu eksperckiego. Pierwsze prototypowe procedury optymalizacyjne w języku LEKS		
2001 IX			Artykuł: EKANWIN2001 - Implementacja hybrydy systemu wspomaganie decyzji i dydaktycznego symulatora przedsiębiorstwa. Symulacja systemów gospodarczych Antałówka'2001 Warszawa 2001, WSPiZ i PWr		
2002 VI			Prace implementacyjne. Integracja modułu wizualizacji z systemem eksperckim o architekturze tablicowej. Implementacja modułowej, proceduralnej architektury języka LEKS		
2002 IX			Artykuł (z E.Radosińskim): Interface użytkownika w inteligentnych hybrydowych systemach wspomaganie decyzji, Symulacja systemów gospodarczych Duszniki'2002, Wrocław 2002, PWr		
2003 VI			Prace nad polepszeniem interaktywności. Rozbudowa języka planowania eksperymentu		
2003 XII			Wbudowanie możliwości edycji parametrów eksperymentu także poza zmiennymi decyzyjnymi wprowadzenie nazw zmiennych do języka LEKS. Uzgodnienie modelu symulacyjnego oraz generowanych sprawozdań ze znowelizowaną ustawą o rachunkowości (także w wersji DOS - EK_AN_)		
2004 III			Implementacja automatycznego instalatora Ekanwina na podstawie środowiska programistycznego InnoSetup (jrsoftware.org). Implementacja automatycznego raportowania czynności analitycznych i operatorskich oraz modułu systemu eksperckiego i sprawozdań finansowych do plików tekstowych.		
2004 VI			Opracowanie zestawu procedur optymalizacyjnych w języku LEKS: metoda połowienia przedziału, Hooke'a-Jeevesa, opracowanie techniki wyznaczania hiperpowierzchni proggu rentowności i jej implementacja opracowanie koncepcji i implementacji algorytmu genetycznego w języku LEKS - wszystko zintegrowane z symulatorem Ekanwin druga wersja interaktywnego interfejsu użytkownika: obsługa analizy dynamiki wskaźników, wbudowanie zgrubnej analizy wartości wskaźnika (bez konieczności budowania bazy wiedzy) wbudowanie możliwości animacji wykresów w zależności od zakresu i szybkości zmian zmiennych decyzyjnych (rozszerzenie analizy wrażliwości)		
2004 IX			Artykuł: Wybrane techniki poszukiwania optymalnych rozwiązań w symulacyjnej analizie ekonomicznej przedsiębiorstwa, Symulacja systemów gospodarczych, Wrocław 2004, PWr		
2004 XII			Język LEKS: opracowanie procedur systematycznego przeglądu wartości wskaźników ekonomicznych w zależności od wartości macierzy zmiennych decyzyjnych rozszerzenie interfejsu o pulpit sterujący umożliwiający równoczesne zmiany wielu komórek macierzy zmiennych decyzyjnych		
2005 I					Artykuł (z E.Radosińskim) do Badań Operacyjnych i Decyzji (w recenzji)

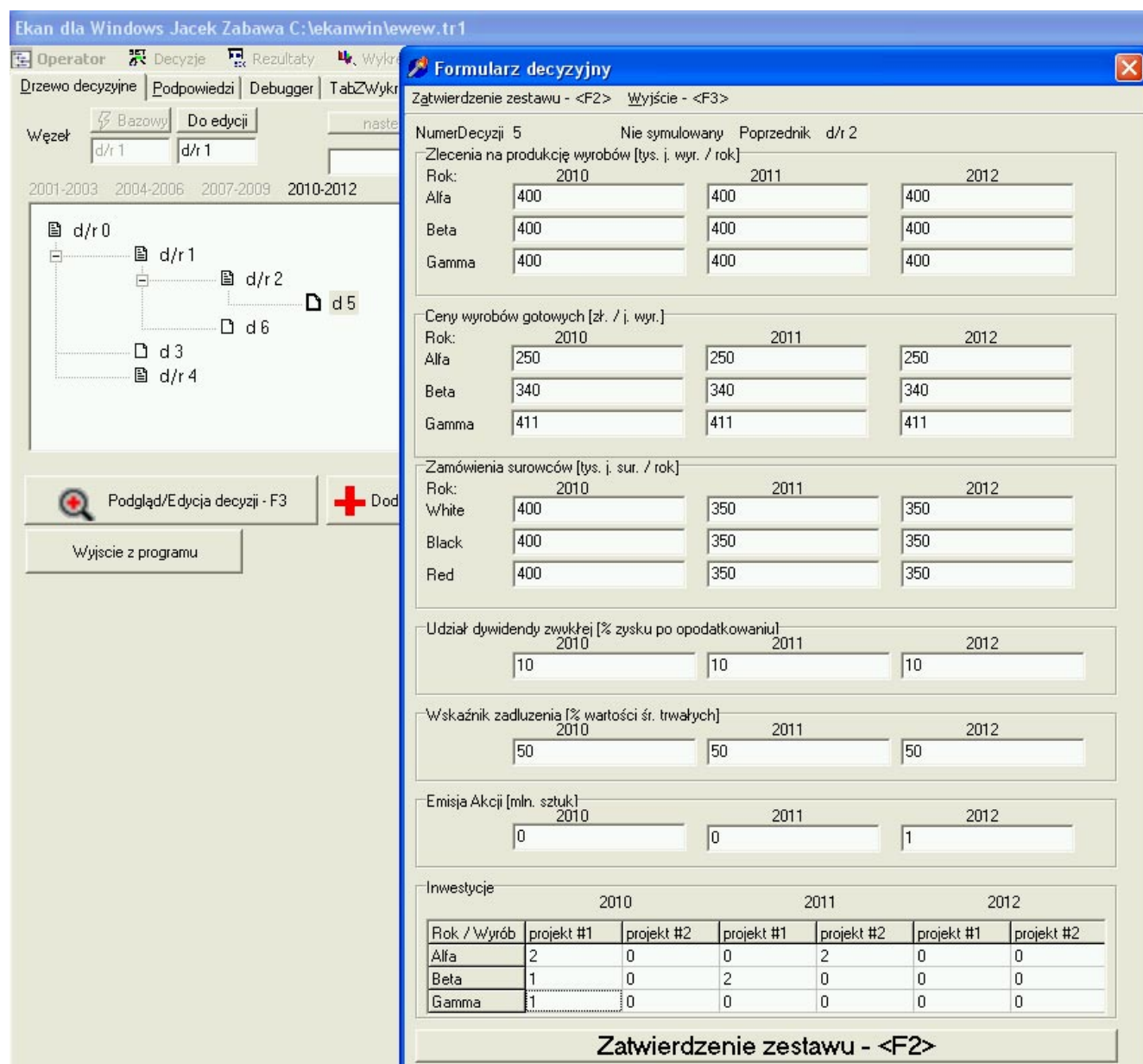
9.5 Opis symulatora Ek_An i przykłady generowanych sprawozdań finansowych

W dydaktycznych symulatorach z rodziny Ek_An zamodelowane jest przedsiębiorstwo przemysłowe charakteryzujące się następującymi cechami:

- Spółka akcyjna z udziałem Skarbu Państwa, finansująca swoją działalność kapitałami własnymi i obcymi
- Każdy z trzech wydziałów produkcyjnych o nazwach: Alfa, Beta, Gamma produkuje wyroby odpowiednio o nazwach: Alfa, Beta, Gamma
- Wyroby wytwarzane są z trzech rodzajów surowców: White, Black, Red wykorzystywanych w różnych proporcjach. Macierz surowcochłonności określa niezbędne do wytworzenia danego wyrobu (jednostkowego) ilości danego rodzaju surowca, analogicznie wektor pracochłonności określa czas pracy (liczbę roboczogodzin) niezbędny do wytworzenia danego rodzaju wyrobu
- Reprodukacja środków trwałych, ulegających dekapitalizacji, realizowana jest przez uruchomienie inwestycji w środki trwałe
- Przychody i wydatki przedsiębiorstwa, spłaty kredytów, wypłaty dywidend i podatków itp. realizowane są za pośrednictwem bankowego rachunku rozliczeniowego
- Istnieje możliwość emisji akcji zwykłych oraz zaciągania kredytów długoterminowych (hipotecznych) i krótkoterminowych
- Inwestycje finansowane są ze środków własnych lub kredytu bankowego (inwestycyjnego)
- Przedsiębiorstwo ponosi wydatki na finansowanie środków trwałych w eksploatacji (remonty, konserwacja)
- Produkcja jest natychmiast sprzedawana, popyt zależy liniowo od ceny wyrobu
- Zysk dzielony jest na wypłatę dywidendy i powiększenie kapitałów przedsiębiorstwa
- Parametry eksperymentu obejmują m.in.: cenę surowca, pracochłonność, narzut kosztów usług, stopa oprocentowania kredytów, płace i narzuty na płace, stawkę amortyzacyjną, okres spłaty należności i zobowiązań, dywidendę skarbu państwa, stawkę podatku dochodowego.

Operator systemu Ek_An wprowadza decyzje wypełniając formularz [Rys. 9-1] dotyczący trzyletniego okresu funkcjonowania przedsiębiorstwa. Formularz składa się z trzech części, określających politykę dotyczącą odpowiednio: wytwarzania i sprzedaży, finansowania i podziału zysku oraz inwestycji w majątek trwały. Trzyletni okres decyzyjny i odpowiadające mu wyniki działalności przedsiębiorstwa (sprawozdania finansowe) tworzą węzeł drzewa decyzyjnego. W drzewie eksperymentów występuje relacja dziedziczenia. Dotyczy ona warunków początkowych i końcowych danego węzła. Warunki końcowe węzła-rodzica są warunkami początkowymi węzła-potomka. Dwa warianty (węzły) decyzji dotyczące tego samego trzyletniego okresu mogą (ale nie muszą) posiadać tego samego rodzica. Zawsze jednak posiadają co najmniej jednego wspólnego

przodka – tzw. sekwencję inicjującą D/R 0 odpowiadająca okresowi początkowemu. Dla sekwencji inicjującej zablokowano możliwość edycji, pozostawiono podgląd wartości zmiennych decyzyjnych.



Rys. 9-1 Ekanwin - drzewo eksperymentów oraz formularz zmiennych decyzyjnych

Dla każdego z węzłów można wyświetlić sprawozdania finansowe dotyczące trzyletniego okresu: bilans, rachunek wyników, przepływy pieniężne, koszty itd. Na poziomie fizycznym drzewo ma postać pliku binarnego zawierającego hierarchicznie – relacyjną bazę danych. Rekordem bazy danych jest węzeł drzewa decyzyjnego (dane wejściowe + wyniki).

Na poziomie węzłów dostępne są następujące operacje: przeglądanie decyzji, kasowanie, dodawanie, przeprowadzenie, eksperymentu symulacyjnego. W graficznej (wyświetlanej na ekranie) formie drzewa eksperymentów [Rys. 9-1] węzeł reprezentowany jest przez ciąg znaków w postaci D numer_węzła (decyzja wprowadzona, bez wykonania eksperymentu) i D/R numer_węzła (decyzja wprowadzona, wykonany eksperyment, dostępne sprawozdania finansowe). Każdy węzeł może stanowić źródło–matrycę konkretnych danych (do skopiowania) podczas przygotowywania

# RACHUNEK ZYSKÓW I STRAT (WARIANT KALKULACYJNY) #				
# oraz RACHUNEK ZYSKÓW/STRAT ZATRZYMANÝCH #				
# W Y S Z C Z E G Ó L N I E N I E #				
# Na koniec roku: #				
# /mln zł/ #				
# 2004 # 2005 # 2006 #				
A.Przychody netto ze sprzedaży produktów, towarów i materiałów	288.2	260.8	235.9	
A.1.Przychody netto ze sprzedaży produktów, towarów i materiałów	288.2	260.8	235.9	
A.1.I.Przychody netto ze sprzedaży produktów	288.2	260.8	235.9	
B.Koszty sprzedanych produktów, towarów i materiałów	210.9	190.8	172.7	
B.I.Koszt wytworzenia sprzedanych produktów	210.9	190.8	172.7	
C.Zysk(strata) brutto ze sprzedaży (A-B)	77.3	69.9	63.3	
D.Koszty sprzedaży	11.5	10.4	9.4	
E.Koszty ogólnego zarządu, w tym	48.8	44.1	39.9	
- koszty wydziałowe	31.6	28.6	25.9	
- pozostałe	17.2	15.5	14.0	
F.Zysk(strata) ze sprzedaży (C-D-E)	17.1	15.4	14.0	
G.Pozostałe przychody operacyjne	0.0	0.0	0.0	
H.Pozostałe koszty operacyjne	0.0	0.0	0.0	
I.Zysk(strata) z działalności operacyjnej	17.1	15.4	14.0	
J.Przychody finansowe	0.0	0.0	0.0	
K.Koszty finansowe	5.8	5.4	5.0	
K.I.Odsetki	5.8	5.4	5.0	
L.Zysk(strata) z działalności gospodarczej (I+J-K)	11.2	10.0	9.0	
M.Wynik zdarzeń nadzwyczajnych (M.I-M.II)	0.0	0.0	0.0	
M.I. Zyski nadzwyczajne	0.0	0.0	0.0	
M.II. Straty nadzwyczajne	0.0	0.0	0.0	
N.Zysk(strata) brutto (L+/-M)	11.2	10.0	9.0	
O.Podatek dochodowy	3.1	2.8	2.5	
P.Pozostałe obowiązkowe zmniejszenia zysku (zwiększenia straty)	1.2	1.1	1.0	
- dywidenda Skarbu Państwa	1.2	1.1	1.0	
R.Zysk(strata) netto (N-O-P), w tym	6.9	6.1	5.5	
- dywidenda zwykła	0.7	0.6	0.6	
- zyski zatrzymane	6.2	5.5	5.0	
- odpis na kapitał zapasowy	0.5	0.4	0.4	
- odpis na kapitał rezerwowy	5.7	5.1	4.6	
# RACHUNEK ZYSKÓW /STRAT ZATRZYMANÝCH #				
# Na koniec roku: #				
# /mln zł/ #				
# 2004 # 2005 # 2006 #				
1. Zyski zatrzymane/ straty poniesione: (skum.) 1 Stycznia	0.0	6.2	11.7	
+ Zysk netto/strata w danym roku	6.9	6.1	5.5	
Razem	6.9	12.3	17.2	
Minus: Dywidenda zwykła	0.7	0.6	0.6	
2. Zyski zatrzymane/ straty poniesione (skum.) 31 Grudnia	6.2	11.7	16.6	

Z E S T A W I E N I E Z M I A N W K A P I T A L E (F U N D U S Z U) W Ł A S N Y M				
W Y S Z C Z E G Ó L N I E N I E				
	Stan:			
	/mln zł/			
	2004	2005	2006	2
I.Kapitał(fundusz) własny na początek okresu (BO)	187.5	193.7	199.2	
- korekty błędów podstawowych	0.0	0.0	0.0	
I.a. Kapitał (fundusz) własny na początek okresu (BO), po korektach	187.5	193.7	199.2	
I.a.1. Kapitał (fundusz) podstawowy na początek okresu	171.0	171.0	171.0	
I.a.1.1. Zmiany kapitału (funduszu) podstawowego	0.0	0.0	1.0	
I.a.1.1.a) zwiększenie (z tytułu)	0.0	0.0	1.0	
- wydania udziałów (emisji akcji)	0.0	0.0	1.0	
I.a.1.1.b) zmniejszenie (z tytułu)	0.0	0.0	0.0	
- umorzenia udziałów (akcji)	0.0	0.0	0.0	
I.a.1.2. Kapitał (fundusz) podstawowy na koniec okresu	171.0	171.0	172.0	
I.a.2. Należne wpłaty na kapitał podstawowy na początek okresu	0.0	0.0	0.0	
I.a.2.1. Zmiana należnych wpłat na kapitał podstawowy	0.0	0.0	0.0	
I.a.2.1.a) zwiększenie	0.0	0.0	0.0	
I.a.2.1.b) zmniejszenie	0.0	0.0	0.0	
I.a.2.2. Należne wpłaty na kapitał podstawowy na koniec okresu	0.0	0.0	0.0	
I.a.3. Udziały (akcje) własne na początek okresu	170.0	170.0	170.0	
I.a.3.a) zwiększenie	0.0	0.0	0.0	
I.a.3.b) zmniejszenie	0.0	0.0	0.0	
I.a.3.1. Udziały (akcje) własne na koniec okresu	170.0	170.0	170.0	
I.a.4. Kapitał (fundusz) zapasowy na początek okresu	1.3	1.8	2.3	
I.a.4.1. Zmiany kapitału (funduszu) zapasowego	0.5	0.4	0.4	
I.a.4.1.a) zwiększenie (z tytułu)	0.5	0.4	0.4	
- emisji akcji powyżej wartości nominalnej	0.0	0.0	0.0	
- z podziału zysku (ustawowo)	0.5	0.4	0.4	
- z podziału zysku (ponad wymaganą ustawowo minimalną wartość)	0.0	0.0	0.0	
I.a.4.1.b) zmniejszenie (z tytułu)	0.0	0.0	0.0	
- pokrycia straty	0.0	0.0	0.0	
I.a.4.2. Stan kapitału (funduszu) zapasowego na koniec okresu	1.8	2.3	2.7	
I.a.5. Kapitał (fundusz) z aktualizacji wyceny na początek okresu	0.0	0.0	0.0	
I.a.5.1. Zmiany kapitału (funduszu) z aktualizacji wyceny	0.0	0.0	0.0	
I.a.5.1.a) zwiększenie (z tytułu)	0.0	0.0	0.0	
I.a.5.1.b) zmniejszenie (z tytułu)	0.0	0.0	0.0	
- zbycia środków trwałych	0.0	0.0	0.0	
I.a.5.2. Kapitał (fundusz) z aktualizacji wyceny na koniec okresu	0.0	0.0	0.0	
I.a.6. Pozostałe kapitały (fundusze) rezerwowe na początek okresu	15.2	20.9	26.0	
I.a.6.1. Zmiany pozostałych kapitałów (funduszy) rezerwowych	5.7	5.1	4.6	
I.a.6.1.a) zwiększenie (z tytułu)	5.7	5.1	4.6	
I.a.6.1.b) zmniejszenie (z tytułu)	0.0	0.0	0.0	
I.a.6.2. Pozostałe kapitały (fundusze) rezerwowe na koniec okresu	20.9	26.0	30.5	
I.a.7. Zysk (strata) z lat ubiegłych na początek okresu	0.0	0.0	0.0	
I.a.7.1. Zysk z lat ubiegłych na początek okresu	0.0	0.0	0.0	
- korekty błędów podstawowych	0.0	0.0	0.0	
I.a.7.2. Zysk z lat ubiegłych na początek okresu, po korektach	0.0	0.0	0.0	
I.a.7.2.a) zwiększenie (z tytułu)	0.0	0.0	0.0	
- podziału zysku z lat ubiegłych	0.0	0.0	0.0	
I.a.7.2.b) zmniejszenie (z tytułu)	0.0	0.0	0.0	
I.a.7.3. Zysk z lat ubiegłych na koniec okresu	0.0	0.0	0.0	
I.a.7.4. Strata z lat ubiegłych na początek okresu	0.0	0.0	0.0	
- korekty błędów podstawowych	0.0	0.0	0.0	
I.a.7.5. Strata z lat ubiegłych na początek okresu, po korektach	0.0	0.0	0.0	
I.a.7.5.a) zwiększenie (z tytułu)	0.0	0.0	0.0	
- przeniesienia straty z lat ubiegłych do pokrycia	0.0	0.0	0.0	
I.a.7.5.b) zmniejszenie (z tytułu)	0.0	0.0	0.0	
I.a.7.6. Strata z lat ubiegłych na koniec okresu	0.0	0.0	0.0	
I.a.7.7. Zysk (strata) z lat ubiegłych na koniec okresu	0.0	0.0	0.0	
I.a.8. Wynik netto	6.9	6.1	5.5	
I.a.8.a) zysk netto	6.9	6.1	5.5	
I.a.8.b) strata netto	0.0	0.0	0.0	
I.a.8.c) odpisy z zysku	6.2	5.5	5.0	
II. Kapitał (fundusz) własny na koniec okresu (BZ)	193.7	199.2	205.2	
III. Kapitał (fundusz) własny, po uwzględnieniu proponowanego podziału zysku (pokrycia straty)	193.7	199.2	205.2	

K O S Z T Y									
Na koniec roku:					Na koniec roku:				
/mln zł/					/mln zł/				
T R E Ś Ć:	2004	2005	2006	T R E Ś Ć:	2004	2005	2006		
KOSZTY OPERACYJNE:	268.0	242.5	219.4	3. AMORTYZACJA:	8.5	7.7	6.9		
- wyrób ALFA	63.0	57.0	51.6	- wydział ALFA	2.0	1.8	1.6		
- wyrób BETA	105.8	95.8	86.6	- wydział BETA	2.0	1.8	1.6		
- wyrób GAMMA	99.1	89.7	81.2	- wydział GAMMA	2.0	1.8	1.6		
w tym:									
#1. ZUŻYCIE	93.1	84.2	76.2	4. USŁUGI OBCE:	21.5	19.5	17.6		
MATERIAŁÓW:									
- wydział ALFA	30.5	27.6	24.9	- wydział ALFA	2.8	2.5	2.3		
- wydział BETA	19.5	17.6	15.9	- wydział BETA	3.8	3.4	3.1		
- wydział GAMMA	43.2	39.1	35.3	- wydział GAMMA	7.6	6.9	6.2		
#2. WYNAGRODZENIA I	143.3	129.6	117.3	5. PODATKI I OPŁATY	1.6	1.4	1.3		
ŚWIADCZENIA									
- wydział ALFA	22.3	20.1	18.2	KOSZTY FINANSOWE	5.8	5.4	5.0		
- wydział BETA	66.8	60.4	54.6						
- wydział GAMMA	37.1	33.6	30.4						
				KOSZT wyrobów	210.9	190.8	172.7		
				SPRZEDANYCH					
				- wyrób ALFA	51.6	46.6	42.2		
				- wyrób BETA	81.2	73.5	66.5		
				- wyrób GAMMA	78.1	70.7	64.0		

R A C H U N E K P R Z E P Ł Y W Ó W P I E N I Ę Ż N Y C H (met. bezpośr.)									
Na koniec roku:					Na koniec roku:				
/mln zł/					/mln zł/				
T R E Ś Ć	2004	2005	2006	T R E Ś Ć	2004	2005	2006		
A.Przepływy środków pieniężnych z działalności operacyjnej	-7.8	-6.9	-15.4						
A.I.Wpływy	291.8	263.9	238.8						
A.II.Wydatki	299.6	270.8	254.2						
A.III.Przepływy pieniężne netto z działalności operacyjnej	-7.8	-6.9	-15.4						
B.Przepływy środków pieniężnych z działalności inwestycyjnej	0.0	0.0	0.0						
B.I.Wpływy	0.0	0.0	0.0						
B.II.Wydatki	0.0	0.0	0.0						
B.II.1.Nabycie wartości niematerialnych i prawnych oraz rzeczowych aktywów trwałych	0.0	0.0	0.0						
B.II.2.Inwestycje w nieruchomości oraz wartości niematerialne i prawne	0.0	0.0	0.0						
B.III.Przepływy pieniężne netto z działalności inwestycyjnej	0.0	0.0	0.0						
C.Przepływy środków pieniężnych z działalności finansowej	-10.3	-10.5	-9.0						
C.I.Wpływy	0.0	0.0	1.0						
C.I.1.Wpływy netto z wydania udziałów (emisji akcji) i innych instrumentów kapitałowych oraz dopłat do kapitału	0.0	0.0	1.0						
C.I.2.Kredyty i pożyczki	0.0	0.0	0.0						
C.II.Wydatki	10.3	10.5	10.0						
C.II.2.Dywidendy i inne wypłaty na rzecz właścicieli	1.9	1.7	1.5						
C.II.4.Spłaty kredytów i pożyczek	2.6	3.4	3.5						
C.II.7.Odsetki	5.8	5.4	5.0						
C.III.Przepływy pieniężne netto z działalności finansowej	-10.3	-10.5	-9.0						
D.Przepływy pieniężne netto, razem (A.III+B.III+C.III)	-18.1	-17.4	-24.4						
F.Środki pieniężne na początek okresu	66.4	48.3	30.8						
G.Środki pieniężne na koniec okresu (F-D)	48.3	30.8	6.3						

Z A P A S Y									
Stan na koniec roku:					Stan na koniec roku:				
W I E L K O Ś Ć					W I E L K O Ś Ć				
	2004	2005	2006		2004	2005	2006		
#1. Zapas surowców				#2. Zapas wyrobów gotowych					
[tys.j.sur.]				[tys.j.wyr.]					
- surowiec WHITE	188.2	282.9	401.9	- wyrób ALFA	5.5	4.9	4.5		
- surowiec BLACK	188.2	282.9	401.9	- wyrób BETA	5.5	4.9	4.5		
- surowiec RED	188.2	282.9	401.9	- wyrób GAMMA	5.5	4.9	4.5		
#3. Zapas produkcji w toku									
[tys.j.pwt.]									
- wyrób ALFA	49.1	44.4	40.2						
- wyrób BETA	49.1	44.4	40.2						
- wyrób GAMMA	49.1	44.4	40.2						

R A C H U N E K Z Y S K Ó W I S T R A T (wariant porównawczy)

Na koniec roku:			
/mln zł/			
	2004	2005	2006
# WYSZCZEGÓLNIENIE			
# A.Przychody netto ze sprzedaży i zrównane z nimi,	285.0	257.9	233.3
w tym:			
# A.I.Przychody netto ze sprzedaży produktów	288.2	260.8	235.9
- produkt ALFA	72.0	65.1	58.9
- produkt BETA	97.9	88.6	80.1
- produkt GAMMA	118.3	107.1	96.9
# A.II.Zmiana stanu produktów	-3.2	-2.9	-2.6
(zwiększenie - wartość dodatnia,			
zmniejszenie - wartość ujemna)			
# B.Koszty działalności operacyjnej	268.0	242.5	219.4
# B.I.Amortyzacja produktów	8.5	7.7	6.9
# B.II.Zużycie materiałów i energii	93.1	84.2	76.2
# B.III.Uслуги obce	21.5	19.5	17.6
# B.IV.Podatki i opłaty	1.6	1.4	1.3
# B.V.Wynagrodzenia	96.8	87.6	79.3
# B.VI.Ubezpieczenia społeczne i inne świadczenia	46.5	42.0	38.0
# C.Zysk(strata) ze sprzedaży (A-B)	17.1	15.4	14.0
# D.Pozostałe przychody operacyjne	0.0	0.0	0.0
# E.Pozostałe koszty operacyjne	0.0	0.0	0.0
# F.Zysk(strata) z działalności operacyjnej (C+D-E)	17.1	15.4	14.0
# G.Przychody finansowe	0.0	0.0	0.0
# H.Koszty finansowe	5.8	5.4	5.0
# H.I.Odsetki	5.8	5.4	5.0
# I.Zysk(strata) z działalności gospodarczej (F+G-H)	11.2	10.0	9.0
# J.Wynik zdarzeń nadzwyczajnych (J.I-J.II)	0.0	0.0	0.0
# J.I.Zyski nadzwyczajne	0.0	0.0	0.0
# J.II.Straty nadzwyczajne	0.0	0.0	0.0
# K.Zysk(strata) brutto (I+/-J)	11.2	10.0	9.0
# L.Podatek dochodowy	3.1	2.8	2.5
# M.Pozostałe obowiązkowe zmniejszenia zysku	1.2	1.1	1.0
(zmniejszenia straty)			
# N.Zysk(strata) netto (K-L-M)	6.9	6.1	5.5
- dywidenda zwykła	0.7	0.6	0.6
- zyski zatrzymane	6.2	5.5	5.0

9.6 Opis języka sterowania eksperymentem LEKS

Operacje plikowe

// * wczytaj plik nazwa_pliku

wczytanie pliku struktury drzewa (należy podać rozszerzenie nazwy „tr1”) oraz (automatycznie) pliku binarnego (nazwa_pliku.sir). Ciąg znaków identyfikatora nazwa_pliku nie może mieć więcej niż osiem znaków. Aktywacja wczytanego pliku jako drzewa decyzyjnego. Bieżącym węzłem będzie węzeł D/R 0

Przykład:

wczytaj plik „variantc.tr1” załaduje także plik „variantc.sir”

// * utwórz plik nazwa_pliku.

Utworzenie pliku struktury drzewa (nazwa_pliku.tr1) oraz (automatycznie) pliku binarnego (nazwa_pliku.sir). Ciąg znaków identyfikatora nazwa_pliku nie może mieć więcej niż osiem znaków. Aktywacja utworzonego pliku jako drzewa decyzyjnego. Bieżącym węzłem będzie węzeł D/R 0

Przykład:

utwórz plik „variantc.tr1” utworzy także plik „variantc.sir”

Operacje dotyczące struktury drzewa

// * ustaw węzeł numer_wezla

Aktywowanie węzła o unikatowym, jednoznacznym identyfikatorze numer_wezla. Kolejne operacje dotyczące danych reprezentowanych przez węzeł-sekwencję decyzyjną (wartości zmiennych decyzyjnych, wyniki symulacji) będą dotyczyć tego węzła. Numer węzła widoczny jest na schematycznym rysunku bieżącego drzewa decyzyjnego

Przykład:

ustaw węzeł lewy „lewy” jest identyfikatorem uprzednio zadeklarowanej zmiennej numerycznej; jeśli w drzewie nie ma węzła którego identyfikator jest równy wartości zmiennej reprezentowanej przez identyfikator to aktywny węzeł nie ulega zmianie.

ustaw węzeł X3 aktywuje węzeł o identyfikatorze równym wartości zmiennej „X3” (wewnętrzna reprezentacja zmiennych). Reszta uwag jw. Ten sposób nie jest zalecany.

ustaw węzeł 3 aktywuje węzeł o identyfikatorze „3”. Reszta uwag jw.

// * kasuj węzeł numer_wezla

Usuwanie węzła o unikatowym, jednoznacznym identyfikatorze numer_wezla. Numer węzła widoczny jest na schematycznym rysunku bieżącego drzewa decyzyjnego. Nie powinno się kasować aktywnego węzła oraz węzła posiadającego potomków.

Przykłady:

kasuj węzeł lewy „lewy” jest identyfikatorem uprzednio zadeklarowanej zmiennej numerycznej; jeśli w drzewie nie ma węzła którego identyfikator jest równy wartości zmiennej reprezentowanej przez identyfikator to aktywny węzeł nie ulega zmianie.

kasuj węzeł X3 usuwa węzeł o identyfikatorze równym wartości zmiennej „X3” (wewnętrzna reprezentacja zmiennych). Reszta uwag jw. Ten sposób nie jest zalecany.

kasuj węzeł 3 usuwa węzeł o identyfikatorze „3”. Reszta uwag jw.

// * odczyt węzeł zmienna

Ustalenie identyfikatora aktywnego węzła i przechowanie tej informacji w zmiennej

Przykłady:

odczyt węzeł lewy „lewy” jest identyfikatorem uprzednio zadeklarowanej zmiennej numerycznej.

odczyt węzeł X3 wczytuje identyfikator aktywnego węzła do zmiennej „X3” (wewnętrzna reprezentacja zmiennych).

// * odczyt ojciec zmienna

Ustalenie identyfikatora przodka (rodzica) aktywnego węzła i przechowanie tej informacji w zmiennej

Przykłady:

ojciec węzeł lewy „lewy” jest identyfikatorem uprzednio zadeklarowanej zmiennej numerycznej.

ojciec węzeł X3 wczytuje identyfikator rodzica aktywnego węzła do zmiennej „X3” (wewnętrzna

reprezentacja zmiennych). Ten sposób nie jest zalecany.

```
// * dodaj węzeł numer_wezla_ojca,numer_wezla_kopiowanego,numer_dodanego_wezla
```

Dodanie nowego węzła do drzewa. Przodkiem nowego węzła jest węzeł o identyfikatorze równym wartości zmiennej „numer_wezla_ojca” a parametry nowego węzła pochodzą z szablonu wzorcowego wartości węzła o identyfikatorze równym wartości zmiennej „numer_wezla_kopiowanego”. Nowemu węzłowi przydzielony zostanie nowy identyfikator (numer) którego wartość zostanie zapisana w zmiennej „numer_dodanego_wezla”. Dwa pierwsze parametry procedury mogą być zarówno identyfikatorami zmiennych jak i stałymi wartościami liczbowymi natomiast trzeci parametr musi być identyfikatorem zmiennej.

Przykłady:

dodaj węzeł 0, 4, nowosc Dodanie nowego węzła; rodzicem będzie węzeł 0, węzłem – szablonem będzie węzeł nr 4. Po dodaniu nowego węzła jego identyfikator będzie zapisany w zmiennej „nowosc”

dodaj węzeł aa, bb, nowosc Dodanie nowego węzła; rodzicem będzie węzeł o identyfikatorze równym wartości zmiennej „aa”, węzłem – szablonem będzie węzeł o identyfikatorze równym wartości zmiennej „bb”. Po dodaniu nowego węzła jego identyfikator będzie zapisany w zmiennej „nowosc”.

dodaj węzeł X1, X3, X2 Dodanie nowego węzła; rodzicem będzie węzeł o identyfikatorze równym wartości zmiennej „X1” (wewnętrzna reprezentacja), węzłem – szablonem będzie węzeł o identyfikatorze równym wartości zmiennej „X3” (wewnętrzna reprezentacja). Po dodaniu nowego węzła jego identyfikator będzie zapisany w zmiennej „X2” (wewnętrzna reprezentacja). Ten sposób nie jest zalecany.

```
// * symuluj węzeł numer_wezla
```

Uruchamia proces obliczeń (symulację) określonego węzła-sekwencji – dzięki którym dostępna będzie informacja o wynikach finansowych oraz węzeł będzie mógł być węzłem bazowym (przodkiem, rodzicem)

Przykłady:

symuluj węzeł lewy „lewy” jest identyfikatorem uprzednio zadeklarowanej zmiennej numerycznej; *symuluj węzeł X3* uruchamia obliczenia dla węzła o identyfikatorze równym wartości zmiennej „X3” (wewnętrzna reprezentacja zmiennych). Reszta uwag jw. Ten sposób nie jest zalecany.

symuluj węzeł 3 uruchamia obliczenia dla węzła o identyfikatorze „3”. Reszta uwag jw.

Operacje dotyczące ustawienia parametrów poszczególnych węzłów

```
// * zmiana zlecenia alfa|beta|gamma rok1|rok2|rok3 wartosc
// * zmiana cena alfa|beta|gamma rok1|rok2|rok3 wartosc
// * zmiana zamówienia surowców white|black|red rok1|rok2|rok3 wartosc
// * zmiana dywidenda rok1|rok2|rok3 wartosc
// * zmiana zadłużenie rok1|rok2|rok3 wartosc
// * zmiana emisja akcji rok1|rok2|rok3 wartosc
// * zmiana inwestycje alfa|beta|gamma rok1|rok2|rok3 projekt1|projekt2 wartosc
```

Operację zmieniającą wartości macierzy zmiennych decyzyjnych z domyślnych na definiowane przez użytkownika. Wartość która powinna znaleźć się w odpowiedniej komórce macierzy przechowywana jest w zmiennej o identyfikatorze „wartosc”

Przykłady:

zmiana zlecenia alfa rok3 lewy Nadanie zleceniom produkcyjnym dla linii produkcyjnej Alfa w trzecim roku sekwencji wartości przechowywanej w zmiennej o identyfikatorze „lewy”

zmiana zlecenia alfa rok3 X4 Nadanie zleceniom produkcyjnym dla linii produkcyjnej Alfa w trzecim roku sekwencji wartości przechowywanej w zmiennej o identyfikatorze „X4” (reprezentacja wewnętrzna. Sposób nie jest zalecany.

zmiana inwestycje beta rok3 projekt2 4 Uruchomienie inwestycji typu „4” na linii produkcyjnej Beta w drugim roku sekwencji. Uwaga: w każdym roku na danej linii można uruchomić dwie inwestycje: tu uruchomiono drugą (projekt2).

Operacje dotyczące ustawienia parametrów działania interpretera oraz przebiegu programu

```
// * run
```

Flaga wyłączająca pracę krokową i umożliwiającą wykonywanie programu z pełną szybkością. Wyłącza działanie przycisku ekranowego „Następne polecenia”. Domyślnie wyłączona

Przykłady:

```
run
```

```
// * call nazwa_pliku_z_rozkazami.roz
```

Przekazuje sterowanie plikowi tekstowemu „nazwa_pliku_z_rozkazami.roz”. Po zakończeniu podprogramu sterowanie ponownie obejmuje program główny (o wyższym poziomie).

Przykłady:

```
call optymalizacja_alfa.roz wywołuje podprogram zawarty w pliku „optymalizacja_alfa.roz”
```

```
// * call nazwa_pliku_z_rozkazami.roz nazwa_zmiennej_1, wartość1, nazwa_zmiennej_2, wartość2
```

Przekazuje sterowanie plikowi tekstowemu „nazwa_pliku_z_rozkazami.roz” przy czym zmiennej o identyfikatorze „nazwa_zmiennej_1” nadawana jest wartość równa „wartość1” oraz zmiennej o identyfikatorze „nazwa_zmiennej_2” nadawana jest wartość równa „wartość2”. Umożliwia to realizację przekazywania wartości parametrów między procedurami. Po zakończeniu podprogramu sterowanie ponownie obejmuje program główny (o wyższym poziomie).

Przykłady:

```
call optymalizacja_alfa.roz pocz,10, kon, 20 wywołuje podprogram zawarty w pliku „optymalizacja_alfa.roz” oraz zmienna „pocz” ma wartość 10 i zmienna „kon” ma wartość 20
```

```
// * show
```

Flaga włączająca okno podglądu wykonywania programu i wartości pierwszych dwudziestu zmiennych. Domyślnie włączona. Porównaj z *hide*

Przykłady:

```
show
```

```
// * hide
```

Flaga wyłączająca okno podglądu wykonywania programu i wartości pierwszych dwudziestu zmiennych. Domyślnie wyłączona. Porównaj z *show*

Przykłady:

```
hide
```

```
// * raport
```

Flaga włączająca raportowanie do pliku tekstowego. Współdziała z instrukcją „zanotuj nazwa_zmiennej”, tzn. że aby instrukcja „zanotuj” działała wpierw musi zostać włączona flaga „raport”

Przykłady:

```
raport
```

```
// * stop
```

Instrukcja kończąca wykonywanie programu bez zamknięcia okna podglądu wykonywania programu.

Przykłady:

```
stop
```

```
// * stop all
```

Instrukcja kończąca wykonywanie programu oraz zamykająca okno podglądu wykonywania programu.

Przykłady:

```
stop all
```

```
// * rem
```

Instrukcja którą można wykorzystać w celu zapisania komentarzy. Wszystkie znaki po słowie kluczowym „rem” aż do końca wiersza nie są interpretowane

```
// * pauza
```

Instrukcja pusta. Przydatna w przypadku pracy krokowej.

Przykład:

```
pauza
```

Operacje we-wy klawiatura-ekran

```
// * pokaz Xnumer_zmiennej
```

Instrukcja umożliwia wyświetlenie wartości danej zmiennej w specjalnym okienku

Przykłady:

```
    pokaz lewy      wyświetlenie wartości zmiennej „lewy” w okienku
```

```
    pokaz X20      wyświetlenie wartości zmiennej „X20” (reprezentacja wewnętrzna) w okienku. Sposób niezalecany
```

```
// * zapytaj nazwa_zmiennej
```

Instrukcja umożliwia nadanie zmiennej o nazwie „nazwa_zmiennej” wartości liczbowej

Przykłady:

```
    zapytaj lewy    Nadanie zmiennej „lewy” wartości wprowadzonej z klawiatury
```

```
    zapytaj X3      Nadanie zmiennej „X3” (wewnętrzna reprezentacja) wartości wprowadzonej z klawiatury.
```

Sposób niezalecany.

```
// * zanotuj nazwa_zmiennej
```

Instrukcja wyprowadzania wartości liczbowej zmiennej o nazwie „nazwa_zmiennej” do pliku tekstowego. Aby zadziałała należy wywołać instrukcję „raport”. Nazwa otwartego pliku tekstowego pochodzi od nazwy głównego (podstawowego) pliku z programem w języku LEKS (*.roz)

Przykład:

```
    zanotuj lewy   zapisanie wartości zmiennej „lewy” do pliku tekstowego.
```

```
    zanotuj X4     zapisanie wartości zmiennej „X4” (reprezentacja wewnętrzna) do pliku tekstowego. Sposób niezalecany.
```

```
// * message ciąg znaków
```

Instrukcja umożliwiającą wyświetlenie na górnej belce okna podglądu komunikatu słownego.

Przykład:

```
    message napis na belce; wyświetla tekst „napis na belce”
```

Operacje arytmetyczne na zmiennych i liczbach

```
// * deklaracja nazwa_zmiennej
```

Instrukcja, zwykle umieszczana na początku programu służy interpretowaniu określonego ciągu znaków „nazwa_zmiennej” jako identyfikatora zmiennej przechowującej wartości liczbowe. Wewnętrzna reprezentacja zmiennych dotyczy identyfikatorów X1, X2, itd. które są zajmowane przez kolejno deklarowane zmienne. Bezpośrednio po zadeklarowaniu wartość zmiennej jest równa odpowiedniej wartości reprezentacji wewnętrznej.

Przykłady:

```
    deklaracja lewy deklaruje ciąg znaków „lewy” jako zmienną
```

```
// * przypisz zmienna_docelowa, zmienna_zrodlowa
```

Instrukcja kopiowania wartości zmiennej o identyfikatorze „zmienna_zrodlowa” do zmiennej o identyfikatorze „zmienna_docelowa”. Odpowiednik instrukcji przypisania w Pascalu: cel:=zrodlo;

Przykłady:

```
    przypisz lewy, 2 nadanie zmiennej „lewy” wartości 2
```

```
    przypisz lewy, prawy nadanie zmiennej „lewy” wartości reprezentowanej przez zmienną „prawy”
```

```
    przypisz X1, X2 nadanie zmiennej „X1” wartości reprezentowanej przez zmienną „X2”. Nie jest zalecane
```

```
// * zmienna_docelowa ,zmienna_zrodlowa_pierwsza , zmienna_zrodlowa_druga
```

```
// / zmienna_docelowa ,zmienna_zrodlowa_pierwsza , zmienna_zrodlowa_druga
```

```
// + zmienna_docelowa ,zmienna_zrodlowa_pierwsza , zmienna_zrodlowa_druga
```

```
// - zmienna_docelowa ,zmienna_zrodlowa_pierwsza , zmienna_zrodlowa_druga
```

Podstawowe działania arytmetyczne. Wynik operacji na drugim i trzecim parametrze zapisywany jest w zmiennej podanej na pierwszej pozycji. Drugim i trzecim argumentem mogą być wartości liczbowe

Przykłady:

```
    * cel, 2, 3 wartość zmiennej „cel” będzie równa 6
```

```
    / cel, parA, parB wartość zmiennej „cel” będzie równa ilorazowi parA i parB
```

```
    + cel, parA, 3 wartość zmiennej „cel” będzie równa wartości zmiennej parA powiększonej o 3
```

```
    - cel, 3, parB wartość zmiennej „cel” będzie równa różnicy liczby 3 i wartości zmiennej parB
```

+ X10, 1, X10 wartość zmiennej „X10” (reprezentacja wewnętrzna) będzie równa sumie poprzedniej wartości zmiennej „X10” powiększonej o 1. Sposób nie jest zalecany.

// * clear // przypisanie wszystkim zmiennym x?? wartosci zero
Instrukcja przypisująca wszystkim zmiennym (zadeklarowanym i także przed deklaracją) wartość zero

Przykłady:
clear

// * ln zmienna_docelowa, zmienna
Instrukcja umożliwiająca przypisanie zmiennej o identyfikatorze „zmienna_docelowa” wartości logarytmu naturalnego zmiennej o identyfikatorze „zmienna”

Przykłady:
ln cel, 2 logarytm naturalny liczby 2 zapisywany jest w zmiennej „cel”
ln cel, ww logarytm naturalny zmiennej o identyfikatorze „ww” zapisywany jest w zmiennej „cel”
ln X1, X2 logarytm naturalny zmiennej o identyfikatorze „X2” zapisywany jest w zmiennej „X1”. Sposób nie jest zalecany

// * exp zmienna_docelowa, zmienna
Instrukcja umożliwiająca przypisanie zmiennej o identyfikatorze „zmienna_docelowa” wartości potęgi o podstawie e zmiennej o identyfikatorze „zmienna”

Przykłady:
exp cel, 2 potęga o podstawie e liczby 2 zapisywana jest w zmiennej „cel”
exp cel, ww potęga o podstawie e zmiennej o identyfikatorze „ww” zapisywana jest w zmiennej „cel”
exp X1, X2 potęga o podstawie e zmiennej o identyfikatorze „X2” zapisywana jest w zmiennej „X1”. Sposób nie jest zalecany.

// * losuj nazwa_zmiennej
Instrukcja generująca liczbę pseudolosową z przedziału (0,1) o rozkładzie równomiernym rzeczywistym

Przykład:
losuj lewy zmiennej „lewy” przypisana zostanie wartość liczby pseudolosowej z przedziału (0,1)
losuj X4 zmiennej „X4” przypisana zostanie wartość liczby pseudolosowej z przedziału (0,1). Sposób nie jest zalecany

Złożone instrukcje programowe – warunki i pętle

// * while...endwhile

Pętla typu while, tzn. wykonywana wtedy gdy warunek testowany jest spełniony. W programie głównym możliwe jest dziesięciokrotne zagnieżdżanie pętli.

Przykłady:
przypisz lewy, 1
while < lewy, 4
+ lewy, lewy, 1
endwhile

Powyższa konstrukcja zostanie wykonana 2 razy a w jej wyniku wartość zmiennej „lewy” będzie wynosić 4.

przypisz lewy, 1
while <= lewy, 4
+ lewy, lewy, 1
endwhile

Powyższa konstrukcja zostanie wykonana 3 razy a w jej wyniku wartość zmiennej „lewy” będzie wynosić 5

przypisz lewy, 1
while <|> lewy, 2, 5
+ lewy, lewy, 1
endwhile

Powyższa konstrukcja nie zostanie wykonana gdyż 1 jest mniejsze niż 2 (zastosowano operator „należy” a w jej wyniku wartość zmiennej „lewy” będzie wynosić 1

przypisz lewy, 1
while equal ___ lewy, 1

```
+ lewy, lewy, 1
endwhile
```

Powyższa konstrukcja zostanie wykonana jeden raz a w jej wyniku wartość zmiennej „lewy” będzie wynosić 2

Wykaz operatorów:

```
<, >, <>, =, <|>, <=, >=,
less___, greater_, notequal, equal___, należy___, less_eq_, great_eq
```

```
// * if else endif
```

Instrukcja warunkowa. Część do else wykonywana jest gdy warunek jest spełniony a między else i endif gdy nie jest spełniony. W programie głównym możliwe jest dziesięciokrotne zagnieżdżanie instrukcji.

Przykłady:

```
przypisz lewy, 3
przypisz srodek, 2
if > lewy, srodek
    przypisz prawy, 1
else
    przypisz prawy, 0
endif
```

Zmiennej o identyfikatorze „prawy” zostanie przypisana wartość 1.

```
// * if endif
```

Instrukcja warunkowa. Część do endif wykonywana jest gdy warunek jest spełniony. W programie głównym możliwe jest dziesięciokrotne zagnieżdżanie instrukcji.

Przykłady:

```
przypisz lewy, 3
przypisz srodek, 2
przypisz prawy, 0
if > lewy, srodek
    przypisz prawy, 1
endif
```

Zmiennej o identyfikatorze „prawy” zostanie przypisana wartość 1.

Operacje odczytywania wartości pozycji sprawozdań finansowych

Zastosowano lokalizację pozycji sprawozdania w konwencji: numer_wiersza (od 1 wzwyż), strona (od 1 wzwyż), rok (od 1 do 3)

```
// * odczyt rachunek wyników numer_wiersza, strona, rok, nazwa_zmiennej
```

Instrukcja służąca do odczytania wartości dowolnej pozycji sprawozdania „rachunek wyników”. Współrzędne danej pozycji (a właściwie szablon składni kompletnej instrukcji) najwygodniej odczytać za pomocą kreatora definicji wskaźników dostępnego w menu DSS.

Przykład:

```
odczyt rachunek wyników 2, 1, 3, amort odczyt amortyzacji w kosztach operacyjnych w trzecim roku sekwencji i przypisanie ich wartości do zmiennej „amort”
```

```
// * odczyt bilans numer_wiersza, strona, rok, nazwa_zmiennej
```

Instrukcja służąca do odczytania wartości dowolnej pozycji sprawozdania „bilans”. Współrzędne danej pozycji (a właściwie szablon składni kompletnej instrukcji) najwygodniej odczytać za pomocą kreatora definicji wskaźników dostępnego w menu DSS.

Przykład:

```
odczyt bilans 14, 2, 3, suma_bilansowa odczyt sumy bilansowej (sumy aktywów) i w trzecim roku sekwencji i przypisanie ich wartości do zmiennej „suma_bilansowa”
```

```
// * odczyt zmienne decyzyjne numer_wiersza, rok, nazwa_zmiennej
```

Instrukcja służąca do odczytania wartości dowolnej zmiennej decyzyjnej. Współrzędne danej pozycji (a właściwie szablon składni kompletnej instrukcji) najwygodniej odczytać za pomocą kreatora definicji wskaźników dostępnego w menu DSS.

Przykład:

```
odczyt zmienne decyzyjne 8, 3, surowiecRed odczyt zmiennej decyzyjnej „zakup surowca Red w
```

trzecim roku sekwencji i przypisanie jej wartości do zmiennej „surowiecRed”

// * odczyt produkcja numer_wiersza,rok,nazwa_zmiennej
Instrukcja służąca do odczytania wartości dowolnej pozycji sprawozdania „produkcja”. Współrzędne danej pozycji (a właściwie szablon składni kompletnej instrukcji) najwygodniej odczytać za pomocą kreatora definicji wskaźników dostępnego w menu DSS.

Przykład:

odczyt produkcja 1,2, produkcjaAlfarok2

// * odczyt rachunek wyników 8, 1, nrroku, Xnr
Instrukcja służąca do odczytania wartości dowolnej pozycji sprawozdania „rachunek wyników”. Współrzędne danej pozycji (a właściwie szablon składni kompletnej instrukcji) najwygodniej odczytać za pomocą kreatora definicji wskaźników dostępnego w menu DSS.

Przykład:

odczyt rachunek wyników 1, 1, 3, koszty_operacyjne_ogolem dotyczy trzeciego roku

// * odczyt zapasy 6, nrroku, Xnr
Instrukcja służąca do odczytania wartości dowolnej pozycji sprawozdania „zapasy”. Współrzędne danej pozycji (a właściwie szablon składni kompletnej instrukcji) najwygodniej odczytać za pomocą kreatora definicji wskaźników dostępnego w menu DSS.

Przykład:

odczyt zapasy 3, 1, zapas_surowca_red dotyczy pierwszego roku

// * odczyt koszty 1, nrroku, Xnr
Instrukcja służąca do odczytania wartości dowolnej pozycji sprawozdania „koszty”. Współrzędne danej pozycji (a właściwie szablon składni kompletnej instrukcji) najwygodniej odczytać za pomocą kreatora definicji wskaźników dostępnego w menu DSS.

Przykład:

odczyt koszty 8, 3, wydzial_gamma_zuzycie_materialow dotyczy trzeciego roku

// * odczyt cash flow 1, nrroku, Xnr
Instrukcja służąca do odczytania wartości dowolnej pozycji sprawozdania „płynność finansowa przedsiębiorstwa”. Współrzędne danej pozycji (a właściwie szablon składni kompletnej instrukcji) najwygodniej odczytać za pomocą kreatora definicji wskaźników dostępnego w menu DSS.

Przykład:

odczyt cash flow 3,1,przychody_pieniezne_z_dzialalnosci_operacyjnej dotyczy pierwszego roku

// * odczyt wskaźniki rentowności 1, nrroku, nazwa_zmiennej
Instrukcja służąca do odczytania wartości dowolnej pozycji sprawozdania „wskaźniki rentowności”. Brak wsparcia przez kreator definicji wskaźników dostępnego w menu DSS. Proszę zapoznać się ze strukturą sprawozdania.

// * odczyt wskaźniki płynności 1, nrroku, nazwa_zmiennej
Instrukcja służąca do odczytania wartości dowolnej pozycji sprawozdania „wskaźniki płynności”. Brak wsparcia przez kreator definicji wskaźników dostępnego w menu DSS. Proszę zapoznać się ze strukturą sprawozdania.

// * odczyt wskaźniki zadłużenia 1, nrroku, nazwa_zmiennej
Instrukcja służąca do odczytania wartości dowolnej pozycji sprawozdania „wskaźniki zadłużenia”. Brak wsparcia przez kreator definicji wskaźników dostępnego w menu DSS. Proszę zapoznać się ze strukturą sprawozdania.

// * odczyt wskaźniki obrotowości 1, nrroku, nazwa_zmiennej
Instrukcja służąca do odczytania wartości dowolnej pozycji sprawozdania „wskaźniki obrotowości”. Brak wsparcia przez kreator definicji wskaźników dostępnego w menu DSS. Proszę zapoznać się ze strukturą sprawozdania.

// * odczyt wynik syntetyczny numer_wiersza,numer_kolumny, nazwa_zmiennej
Instrukcja służąca do odczytania wartości dowolnej pozycji sprawozdania „wynik syntetyczny”. Brak wsparcia przez kreator definicji wskaźników dostępnego w menu DSS. Proszę zapoznać się ze strukturą sprawozdania.

Bibliografia

1. Ackley David H., Hinton Geoffrey F., Sejnowski Terry J., A learning algorithm for Boltzman machines, *Cognitive Science*, Vol. 9, 1985
2. Adya Monica, Armstrong J. Scott, Collopy Fred, Kennedy Miles, An Application of Rule-based Forecasting to a Situation Lacking Domain Knowledge, *International Journal of Forecasting*, Volume 16, Elsevier Science B.V. 2000, URL: [http://www-marketing.wharton.upenn.edu/forecast/paperpdf/An Application of Rule Based Forecasting.pdf](http://www-marketing.wharton.upenn.edu/forecast/paperpdf/An%20Application%20of%20Rule%20Based%20Forecasting.pdf)
3. Aguirre Jose Luis, Brena Ramon, Cantu Francisco J., Multiagent-based knowledge networks, *Expert Systems with Application*, Vol. 20, Pergamon, Elsevier Science Ltd. 2001
4. Ahn B.S., Cho S.S., Kim C.Y., The integrated methodology of rough set theory and artificial neural network for business failure prediction, *Expert Systems with Application*, Vol. 18, Elsevier Science Ltd. 2000
5. Aitech DSS, Katowice 2005, URL: <http://www.aitech.com.pl>
6. Ajjanagadde Venkat, Shastri Lokendra, Rules and variables in neural nets, *Neural Computation*, Vol. 3, No. 1, 1991
7. Alonso F., Martinez L., Montes C., SACM: Semantic Agenda Control Model, *Expert Systems with Application*, Vol. 12, No. 3, 1997
8. Altman Edward., Avery, R. B., Eisenbeis, R. A., and Sinkey, J. F. jr., *Application of Classification Techniques in Business, Banking and Finance*, Jai Press Inc. 1981
9. Andrzejczak Karol, *Elementy Analizy Decyzyjnej (Zanim podejmiesz decyzję Tom 1)*, Karol Andrzejczak, Poznań 1992
10. Angehrn Albert A., *Computers That Criticize You: Stimulus-based Decision Support System*, *Interfaces*, Vol. 23, Iss. 3, The Institute of Management Sciences 1993
11. Anthony Robert N., *Planning and Control Systems: A Framework for Analysis*, Harvard University Graduate School of Business Administration, Cambridge, MA 1965
12. Ariav Gad, Ginzberg Michael J., *DSS Design: A Systemic View of Decision Support*, *Communications of the ACM* Volume 28, Iss. 10, 1985
13. Armstrong Scott, Collopy Fred, *Principles of Forecasting: A Handbook for Researchers and Practitioners*, Kluwer Academic Publishers, Norwell, MA 2002, *Answers to Frequently Asked Questions*, URL: <http://morris.wharton.upenn.edu/forecast/FAQ.html>
14. Arsham Hossein, *Applied Management Science: Making Good Strategic Decisions*, University of Baltimore 2004, URL: <http://home.ubalt.edu/ntsbarsh/opre640online/opre640online.htm>
15. Bachem Christian W.B., Oomen Ronald J.F.J., Visser Richard G.F., *Transcript Imaging with cDNA-AFLP: A Step-by-Step Protocol*, *Plant Molecular Biology Reporter*, Vol. 16, Kluwer Academic Publishers 1998
16. Bäck Asta, Lehtonen Tapio, Karttunen Anneli, Kuusisto Matti, Launonen Raimo, *Evaluation of Printing Production in Networks by Means of Computer Simulation Models*, VTT Information Technology Finland 2000, URL: http://www.pira.co.uk/admin/_private/TechnicalArticles/00013i.pdf
17. Baldwin-Morgan Amelia A., Sangster Alan, *Strategy and Impacts of Expert Systems for Bank Lending*, *Expert Systems with Application*, Vol. 11, No. 4, Elsevier Science Ltd. 1996
18. Barthelemy J.P., Bisdorff R., Coppin G., *Human Centered Processes and Decision Support Systems*, *European Journal of Operational Research*, Vol. 136, Elsevier Science B.V. 2002
19. Benbasat Izak, Nault Barrie R., *An Evaluation of Empirical Research in Managerial Support Systems*, *Decision Support Systems*, Vol. 6, Elsevier Science B.V. 1990
20. Bidgoli Hossein, *Integrated Artificial Intelligence Technologies Present Unique Decision Support Capabilities*, *Conference Proceedings Of Western Decision Sciences Institute*, Reno, Nevada 1998
21. Black Fischer, Scholes Myron S., *The Pricing of Options and Corporate Liabilities*, *Journal of Political Economy*, Vol. 81 (May/June), The University of Chicago Press 1973
22. Bobsguide, *The Guide to Software & Technology in Asset Management, Banking & Risk Management*, MyGuides Ltd 2002, URL: <http://www.bobsguide.com/guide/risk1.html>
23. Bookstaber Richard, *Global Risk Management: Are We Missing the Point?*, *The Journal of Portfolio Management*, Spring, Institutional Investor, Inc. Journals Group, New York 1997

24. Boryczka Mariusz, Słowiński Roman, Derivation of optimal decision algorithms from decision tables using rough sets, *Bulletin of the Polish Academy of Sciences 1988: Series Technical Sciences*, 36, 252–260
25. Box George E.P, Wilson K.B., On the experimental attainment of optimum conditions, *Journal of the Royal Statistical Society, Series B, Volume 13*, Blackley, Manchester 1951
26. Bradley John H., Hauser Richard D., A framework for Expert System Implementation, *Expert Systems with Application*, Vol. 8, No. 1, Elsevier Science Ltd. 1995
27. Bradshaw Jeffrey M., An Introduction to Software Agents, *Software Agents*, AAI Press/The MIT Press 1997, URL: <http://agents.umbc.edu/introduction/01-Bradshaw.pdf>
28. Bruner Jerome, *Actual Minds, Possible Worlds*, Cambridge, MA: Harvard University Press 1986
29. Bryant Kay, ALEES: an agricultural loan evaluation expert system, *Expert Systems with Application*, Vol. 21, No. 8, Elsevier Science Ltd. 2001
30. Bui Tung X., *Co-op: A group decision support system for cooperative multiple criteria group decision making*. New York, Springer 1987
31. JPMorgan Introduces MorganRisk, *New ASP Gauges Portfolio Risk For Buyside Clients Using Internet*, *Business Wire*, June 11, Boston 2003
32. Cairo Osvaldo, KAMET: A comprehensive methodology for knowledge acquisition from multiple knowledge sources, *Expert Systems with Application*, Vol.14, Elsevier Science Ltd. 1998
33. Cale Jr., Eriksen E.G. & E., Factors affecting the implementation outcome of a mainframe software package: a longitudinal study, *Information & Management*, Volume 26, 1994
34. Cardenosa Jesus, Carlos Juarez, Guillermo Pastor, An intelligent system for problem analysis in organizations, *Expert Systems with Application*, Vol. 15, Pergamon, Elsevier Science Ltd. 1998
35. Cassandras, Christos G., *Discrete Event Systems: Modeling and Performance Analysis*, Irwin and Aksen Associates, Boston 1993
36. Cempel Czesław, *Nowoczesne Zagadnienia Metodologii i Filozofii Badań*, Poznań 2001, URL: <http://neur.am.put.poznan.pl/skrypty.htm>
37. Cempel Czesław, *Modele systemów i ich zachowanie*, Politechnika Poznańska 2003, URL: <http://neur.am.put.poznan.pl/is/>
38. Chander Prabhakar G., Shinghal Rajjan, Radhakrishnan Thiruvengadam., Performance, Assessment and incremental evaluation of rule-based systems in plant, *Proceedings of AAI'97 Verification and Validation of Knowledge-Based Systems Workshop*, AAI Press, Menlo Park CA 1997
39. Chang Yih-Long, *WinQSB: Software and Manual, Version 2.0*, John Wiley & Sons 2002
40. Chen Zhengxin, Knowledge acquisition assisted by CLIPS programming, *Engineering Applications of Artificial Intelligence*, Vol. 12, Pergamon, Elsevier Science Ltd. 1999
41. Chen Mu-Chen, Huang Shih-Hsien, Credit scoring and rejected instances reassigning through evolutionary computation techniques, *Expert Systems with Application*, Vol. 24, Pergamon, Elsevier Science Ltd. 2003
42. Churchman Charles West, *The Design of Inquiring Systems: Basic Concepts of Systems and Organization*, Basic Books, New York 1971
43. Coats Pamela K., Fant L. Franklin, Recognizing financial distress patterns using a neural network tool, *Financial Management*, Vol. 22, No. 3, Dorsey, Edmister and Johnson, Bankruptcy prediction using artificial neural systems, WP University of Mississippi, School of Business 1993
44. Codasyl A modern appraisal of decision tables, *Report of the Decision Table Task Group*, ACM New York 1982
45. Cohen Paul R., Lieberman, Mark D., A Report on FOLIO: An Expert Assistant for Portfolio Managers, (Bundy A. red.) *Proceedings of the 8th International Joint Conference on Artificial Intelligence*, Karlsruhe, William Kaufmann 1983
46. Cohen Paul R., *Heuristic reasoning about uncertainty, An artificial Intelligence Approach*, *Research Notes in Artificial Intelligence*, London, Pitman Publishing 1983

47. Cooper Robin, The rise of activity-based costing – Part Three: how many cost drivers do you need, and how do you select them? *Journal of Cost Management*, Vol. 2, Iss. 4, Thomson RIA 1989
48. Cosmic Solutions, ProAct 2000, <http://www.cosmic-solutions.com/software/proact/index.htm>
49. Courtney James F., Decision Making and knowledge management in inquiring organizations: toward a new decision-making paradigm for DSS, *Decision Support Systems*, Vol. 31, Elsevier Science B.V. 2001
50. Czarny Tadeusz, Metody identyfikacji systemów produkcyjnych. Skrypt, Politechnika Wroclawska 1979
51. D'Agostini Giulio, Subjective probability and Bayes' theorem, La Sapienza University, Roma 2003, URL: <http://www-zeus.roma1.infn.it/~agostini/cern/node31.html>
52. Davenport, T.H., Process Innovation, Harvard Business School Press, Boston, MA 1993
53. Davidson Clive, The race for the missing credit risk solution, *Risk Magazine*, February 2003, Vol.16, No. 2, Incisive Media Plc 2003
54. Davis Randall, Expert systems: where we are? And where do we go from here?, *The Artificial Intelligence Magazine*, Vol. 3, No. 2, AAAI, Menlo Park 1982
55. Delbecq Andrew L., Van de Ven Andrew H., Gustafson David H., Group Techniques for Program Planning, a Guide to Nominal Group Technique and Delphi Processes, Glenview, Scott Foresman and Company 1975
56. Dembski William A., No Free Lunch. Why Specified Complexity Cannot be Purchased Without Intelligence, Lanham, Maryland: Rowman and Littlefield Publishers 2002
57. Dennis Alan R., Carte Traci A., Kelly Gigi G., Breaking the rules: success and failure in groupware-supported business process reengineering, *Decision Support Systems*, Vol. 36, Elsevier Science B.V. 2003
58. DeSanctis, Gerardine, Gallupe Brent R., A Foundation for the Study of Group Decision Support Systems, *Management Science*, May, Vol. 33, No 5, 1987
59. Deutschle Susan, Probability model calculates future worth of estate, *Charlotte Business Journal*, December 3, American City Business Journals Inc. 2001, URL: <http://charlotte.bizjournals.com/charlotte/stories/2001/12/03/focus3.html>
60. Dos Santos Rodrigues L., Holsapple Clyde W., A framework for Designing Adaptive DSS Interfaces, *Decision Support Systems* 5(1), 1989
61. Drucker Ferdynand Peter, *Praktyka zarządzania*, Czytelnik, Warszawa 1994
62. Duch Włodzisław, Fascynujący świat programów komputerowych. Sztuczna inteligencja, Wydawnictwo NAKOM, Poznań 1997, URL: www.phys.uni.torun.pl/~duch/book-fsk.html
63. Duchessi Peter, O'Keefe Robert M., Understanding Expert Systems Success and Failure, *Expert Systems with Application*, Vol. 9, No. 2, Elsevier Science Ltd. 1995
64. Duffy Maureen Nevin, Palisade Takes Add-in Science to New Heights, *Derivatives Strategy*, August 1996, DerivativesStrategy.com
65. Durkin John, Expert System, Design and Development, Prentice Hall, Englewood Cliffs 1994
66. Dutta Soumitra, Shekhar Shashi, Bond rating: A non-conservative application of neural networks, *Proceedings of IEEE International Conference on Neural Networks*, San Diego 1988
67. Dutta Amitava, Integrating AI and optimization for decision support: a survey, *Decision Support Systems*, Vol. 18, Elsevier Science B.V. 1996
68. Dyer Robert F., Forman Ernest H., Group Decision support with the Analytic Hierarchy Process, *Decision Support Systems*, Vol. 9, Elsevier Science Ltd. 1992
69. Dylan Wiliam, Types of Research in Mathematics Education, *Psychology of Mathematics Education (PME) Annual General Meeting* 1999
70. Edvinsson Leif, Malone Michael S., *Intellectual Capital: Realizing Your Company's True Value by Finding Its Hidden Brainpower*, HarperBusiness 1997
71. Edwards Alex, Connell, N.A.D., *Expert Systems in Accounting*, Prentice Hall / ICA, Hertfordshire 1989

72. Eierman Michael A., Niederman Fred, Adams Carl, DSS theory: A model of constructs and relationships, Decision Support Systems 14, Elsevier Science Publishers B. V., Amsterdam, The Netherlands 1995
73. Einhorn Hillel J., Hogarth Robin M., Behavioral decision theory: processes of judgment and choice, Annual Review of Psychology, Vol. 32, Iss. 1, University of Chicago 1981
74. Ellis Clarence A., Gibbs Simon J., Rein Gall L., Groupware: The Research and Development Issues, MicroElectronics and Computer Technology Corporation Report, February 1989
75. Eom Sean B., The Intellectual Development and Structure of Decision Support Systems (1991-1995), Omega – International Journal of Management Science, Vol. 26, No. 5, Elsevier Science Ltd. 1998
76. Eriksson Henrik, Models for knowledge-acquisition tool design, Knowledge acquisition, Vol. 6, No. 1, Academic Press Ltd., London 1994
77. Expertise2Go, Expertise2Go Web-Enabled Expert Systems 2001, <http://www.expertise2go.com>
78. Fairchild Alea, Reengineering and Restructuring the Enterprise: A Management Guide for the 21 century, Computer Technology Research 1998
79. Famili Fazel, Wei-Min Shen, Weber Richard, Simoudis Evangelos, Data Preprocessing and Intelligent Data Analysis, Intelligent Data Analysis, Vol. 1, Iss. 1-4, Elsevier Science B.V. 1997
80. Farmer J. Doayne, Sidorovich John J, Predicting Chaotic Time Series, Physical Review Letters, Vol.59, The American Physical Society 1987
81. Fausett Laurene, Fundamentals of Neural Networks, New York, Prentice Hall 1994
82. Feigenbaum Edward A., The Art of Artificial Intelligence: Themes and Case Studies of Knowledge Engineering, Proceedings of the Fifth International Conference on Artificial Intelligence 1997, pp. 1014-1029
83. Finch David J., Lees Patrick F., Hybrid Knowledge-Based System for Chemical Incident Management, Expert Systems with Application, Vol. 12, Iss. 3, Elsevier Science Ltd. 1997
84. Findeisen Władysław, Iastrebov A., Lande R., Lindsay J., Pearson M., (Quade E.S., red.), A Sample Glossary of Systems Analysis, prepared for the Preliminary Version of the Handbook of Applied Systems Analysis, IIASA, 1978
85. Fischer Gerhard, Human-Computer Interaction Software: Lessons Learned, Challenges Ahead, IEEE Software January/February, Vol. 6, No. 1, 1989
86. Fisher Ronald A., The Design of Experiments, Edinburgh, Oliver and Boyd, 1935
87. Fishman George S., Symulacja komputerowa, pojęcia i metody, PWE. Warszawa 1981
88. Frawley William J., Piatetsky-Shapiro Gregory, Matheus Christopher J., Knowledge discovery in databases: an overview, AI Magazin, Fall 1992, AAAI
89. Gabryś Bogdan, Bargiela Andrzej, Neural networks based decision support in presence of uncertainties, Journal of Water Resources Planning and Management, Vol. 125, No. 5, American Society of Civil Engineers 1999
90. Galindo Jorge, Tamayo Pablo, Credit Risk Assessment using Statistical and Machine Learning Methods as an Ingredient for Risk Modeling of Financial Intermediaries, Computing in Economics and Finance Volume 31, Society for Computational Economics 1997
91. Genesereth Michael R., Nilsson Nils J., Logical foundations of artificial intelligence, San Mateo, CA, Kaufmann 1987
92. Geoffrion Arthur M., FW/SM: A prototype structured modeling environment, Management Science, Vol. 37, No. 12, 1991
93. Giarratano Joseph C., Riley Gary D., Expert Systems: Principles and Programming, Course Technology, 3rd edition, 1998
94. Gibson Michel, Information Technology for Managers, College of Technology, University of Houston 2004, URL: http://www.tech.uh.edu/gibson/IT_for_Managers/IT_for_Mgrs_PPT.htm
95. Gilbert Don, Aparicio Manny, Atkinson Betty, Brady Steve, Ciccarino Joe, Grosf Benjamin, O'Connor Pat, Osisek Damian, Pritko Steve, Spagna Rick, Wilson Les, The role of intelligent agents in the information infrastructure, IBM Intelligent Agent Strategy, IBM Corporation 1995

96. Glover Fred, P. Kelly James, Laguna Manuel, *New Advances And Applications Of Combining Simulation And Optimization*, Proceedings of the 28th Winter Simulation Conference, ACM Press New York 1996
97. Głowiński Cezary, *Sztuka wysokiego składowania*, PC Kurier, 12/2000, Wydawnictwo Lupus
98. Goldberg David E., *Genetic algorithms in search, optimization and machine learning*, Addison-Wesley Professional 1989
99. Goonatikale Suran, Khebbal Sukhdev, *Intelligent Hybrid Systems: Issues, Classifications and Future Directions*, (Goonatilake, Khebbal, red.), *Intelligent hybrid systems*, John Wiley & Sons 1995
100. Gorry, G.A., Morton, M.S., *A Framework for Management Information Systems*, Sloan Business Review, Vol. 13, Iss. 1, MIT Sloan School of Management 1971
101. Green Mark, *A Survey of Three Dialogue Models*, ACM Transactions On Graphics, Vol. 5, No. 3, 1999
102. Green Marc, *Toward a Perceptual Science of Multidimensional Data Visualization: Bertin and Beyond*, ERGO/GERO Human Factors Science 1999, URL: http://abstract.cs.washington.edu/~maneesh/cs558/papers/Green_Towards.doc
103. Griffin Ricky W., *Podstawy zarządzania organizacjami*, PWN, Warszawa 1997
104. van Groningen Charles N., Paddock Robert A., *Smart bridge: Expert Bridge Classification System*, Expert Systems with Application, Vol. 13, No. 2, Pergamon, Elsevier Science Ltd. 1997
105. Guimaraes Tor, Igbaria Magid, Lu Ming-Te, *The Determinants of DSS Success: An Integrated Model*, Decision Sciences 23(2), Blackwell Publishing 1992
106. Guimaraes Tor, Yoon Youngohc, Clevenson Aaron, *Empirically testing ES success factors in business process reengineering*, International Journal of Production Economics, Vol. 50, Elsevier B.V. 1997
107. Gupta Kalyan Moy, Montazemi Ali R., *A connectionist approach for similarity assessment in case-based reasoning systems*, Decision Support Systems, Vol. 19, Elsevier Science B.V. 1997
108. Hanczar Paweł, *Program do tworzenia i przeprowadzania gier symulacyjnych SymServ v.1.0*, Symulacja Systemów Gospodarczych, Warszawa, WSPiZ, Politechnika Wroclawska 2000
109. Harmon Paul, King David, *Expert Systems in Business*, New York, Wiley 1985
110. Heshemi Ray R., Le Blanc Louis A., Rucks Conway T., Rajaratnam A., *A hybrid intelligent system for predicting bank holding structures*, European Journal of Operational Research, Vol. 109, Iss. 2, Elsevier Science B.V. 1998
111. Hayes-Roth Frederick, Waterman Donald A., Lenat Douglas B., *Building Expert Systems*, Reading, MA, Addison-Wesley 1983
112. Hayes-Roth Barbara, *Blackboard systems* In Shapiro Stuart C. (red.) *Encyclopedia of artificial intelligence*, str. 73-80 New York, Wiley 1990
113. Hedberg Sara Reese, *The Data Gold Rush*, BYTE 1995 No 10, URL: <http://www.byte.com/art/9510/sec8/art7.htm#>
114. Heizer Jay, Render Barry, *Operations Management 6th ed.* Prentice Hall, New Jersey 2001
115. Heracleous Loizos, *Better than the Rest: Making Europe the Leader in the Next Wave of Innovation and Performance*, Long Range Planning - International Journal of Strategic Management, February, Cass Business School, London 1998
116. Ho Yu-Chi, Lee Jonathan T., *Study Guide for ES205, Decision Analysis* (aktualnie niedostępny)
117. Ho Yu-Chi, Pepyne David, *A Simple Explanation of the No Free Lunch Theorem*, Proceedings of the 2001 Conference on Decision and Control, Journal of Optimization Theory and Applications, Vol. 115, Iss. 3, New York, Plenum Publishing Corporation 2002
118. Holland John H., *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*, The MIT Press 1992
119. Holsapple Clyde W., Singh Meenu., *The knowledge chain model*, 3rd Annual Conference of the Southern Association for Information Systems (SAIS), Atlanta 2000
120. Horngren Charles T., Foster George, Datar Srikant M., *Cost accounting: a managerial emphasis*, New Jersey: Prentice Hall 1997
121. Hurwicz Mike, *Take Your Data to the Cleaners*, BYTE, Jan 1997

122. Hutchinson James M., Lo Andrew, Poggio Tomaso, A non-parametric Approach to Pricing and Hedging Derivative Securities Via Learning Networks, *The Journal of Finance*, Vol. 49, No. 3, Blackwell Publishing 1994
123. Igbaria Magid, Guimaraes Tor, Empirically testing the outcomes of user involvement in DSS development, *Omega International Journal Management Science*, Vol. 22, Elsevier Science Ltd. 1994
124. Ignizio James P., *Introduction to Expert Systems*, McGraw-Hill, New York 1991
125. Extend 5 Professional Simulation tools, Version 5 User's Guide, ImagineThat Inc. 2002, URL: <http://www.imaginetthatinc.com>
126. Inmon William, *Building the Data Warehouse*, QED Information Sciences, Wellesley, MA 1992
127. Inselbert Alfred, Dimsdale Bernard, Parallel coordinates: a tool for visualizing multi-dimensional geometry, *Proceedings of Visualization '90*, pp. 361-378, Oct. 1990
128. J.P.Morgan, *Introduction to CreditMetrics*, New York, J.P. Morgan & Co. Inc. 1997, URL: <http://www.mbs.umd.edu/Finance/hunal/courses/bmgt745/CrMxintr.pdf>
129. Jajuga Krzysztof, *Value at Risk*, Rynek Terminowy 9/3, 2000
130. Jakubczyc Janina A., Matouk Kamal, Owoc Mieczysław, Sukcesy i porażki systemów inteligentnych, *Nowoczesne technologie informacyjne w zarządzaniu*, AE Wrocław 2002
131. Jakubowski Tomasz, *Zarządzanie wiedzą w firmach konsultingowych*, *Gazeta IT* nr 7, 2002, URL: http://www.gazeta-it.pl/archiwum/git07/zarzadzanie_wiedza_kons.html
132. Jelonek Jacek, Krawiec Krzysztof, Słowiński Roman, Rough set reduction of attributes and their domains for neural networks, *Computational Intelligence*, Vol. 11, Iss. 2, Blackwell Publishing 1995
133. Jensen Paul A., Bard Jonathan F., *Operations Research Models and Methods*, Texas University, Austin 2004, URL: <http://www.me.utexas.edu/~jensen/ORMM/models/unit/simulate/index.html>
134. Jo Hong-Kyu, Han In-Goo, Integration of Case-Based Forecasting, Neural Network, and Discriminant Analysis for Bankruptcy Prediction, *Expert Systems with Application*, Vol. 11, Iss. 4, Elsevier Science Ltd. 1996
135. Johannsen Gunnar, Knowledge-based design of human-machine interfaces, *Control Engineering Practice*, Vol. 3, No. 2, Pergamon, Elsevier Science Ltd. 1995
136. Jorysz H.R., Vernadat Francois, CIM-OSA Part 2: information view, *International Journal of Computer Integrated Manufacturing*, Vol. 3, Nos. 3 and 4, Taylor and Francis 1990
137. Kacprzyński Bogdan, *Planowanie eksperymentów. Podstawy matematyczne*, WNT Warszawa 1974
138. Kahn Kenneth B., Adams Marjorie E., Sales Forecasting As A Knowledge Management Process, *The Journal Of Business Forecasting*, Winter 2000-2001
139. Kahneman Daniel, Slovic Paul, Tversky Amos, *Judgment Under Uncertainty: Heuristics and Biases*, Cambridge University Press, Cambridge 1982
140. Kangassalo Hannu, *Foundations of conceptual modeling: a theory construction view*, *Information Modeling and Knowledge Bases*, Kangassalo H. i inni (red.), IOS Pres, Amsterdam 1990
141. Kant Immanuel, *Krytyka czystego rozumu*, Wydawnictwo Antyk 2001
142. Kapłan Marek, Krok Arkadiusz, Najważniejsze sygnały problemów firmy, *Tygodnik Finansowy*, Invest.Com 2005, URL: <http://www.tf.pl>
143. Kasik David J., A User Interface Management System, *Computer Graphics*, Vol.16, No. 3, 1982
144. Keen Peter G.W., Scott Morton Michael S., *Decision Support Systems: An Organizational Perspective*, Addison-Wesley, Reading, MA 1978
145. Keeney Ralph L., Raiffa Howard, *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*, John Wiley and Sons, New York 1976
146. Kilgour Gray, Kilgour Richard, *Frequently Asked Questions: Hybrid Systems*, 1997, URL: <http://divcom.otago.ac.nz/COM/INFOSCI/SMRL/people/andrew/publications/faq/hybrid/hybrid.htm>
147. Kim Hee-Woong, Kim Young-Gul, Dynamic process modeling for BPR: A computerized simulation approach, *Information & Management* 32, Elsevier Science Ltd. 1997

148. Kim Kyoung-Jae, Han Ingoo, Application of a hybrid genetic algorithm and neural network approach in activity-based costing, *Expert Systems with Application*, Vol. 24, Iss. 1, Elsevier Science Ltd. 1997
149. Kitts B., Edvinsson L., Beding T., Intellectual capital: from intangible assets to fitness landscapes, *Expert Systems with Application*, Vol. 20, Pergamon, Elsevier Science Ltd. 2001
150. Kleiber Michał, Modelowanie i Symulacja Komputerowa - Moda czy Naturalny Trend Rozwoju Nauki, *Nauka*, Nr 4, 1999
151. Kobsa Alfred, Generic User Modeling Systems, *User Modeling and User-Adapted Interaction* 11: 49-63, Kluwer Academic Publisher 2001
152. Kopaliński Władysław, Słownik wyrazów obcych i zwrotów obcojęzycznych, PRO-media CD – De Agostini Polska 2001, URL: <http://www.slovník-online.pl/>
153. Korzeń Zbigniew, Informatyczne systemy badań symulacyjnych i wizualizacji w logistyce – trendy rozwojowe i perspektywy nowych wdrożeń, Politechnika Wrocławska 2001, URL: <http://www.logistyka.net.pl/logistyka/docs/2001/03/06/1653223/Ref-8a.doc>
154. Koza John R., 1992, Genetic programming: On the programming of computers by means of natural selection, Cambridge MA: MIT Press 1992
155. Krawiec Krzysztof, Neural Networks in Decision Support, Politechnika Poznańska 1998, URL: <http://www-idss.cs.put.poznan.pl/~krawiec/nwd/nn-wd98.html>
156. Krippendorff Klaus, Web Dictionary of Cybernetics and Systems, Principia Cybernetica Web 1986, URL: <http://pespmc1.vub.ac.be/ASC/indexASC.html>
157. Kuutti Ilmo, Impacts of product modelling on ship design and production planning reduced direct and indirect costs of design, 9th International Conference on Computer Applications in Shipbuilding, Yokohama 1997, URL: http://www.napa.fi/Standard_features/iccas_97.htm
158. Kuziak Katarzyna, Koncepcja wartości zagrożonej VaR (Value at Risk), Statsoft 2004, URL: <http://www.statsoft.pl/czytelnia/finanse/pdf/kuziak.pdf>
159. Laskey Kathryn Blackmond, Decision Support Systems Engineering, George Mason University 2003, URL: http://ite.gmu.edu/~klaskey/SYST542/DSS_Unit1.pdf
160. Law Duh Foong Daved, Foong Sew Bun, Kwan Shee Eng Jeremiah, An integrated Case-Based Reasoning Approach for Intelligent Help Desk Fault Management, *Expert Systems with Application*, Vol. 13, No. 4, Elsevier Science Ltd. 1997
161. Law Averill M., McComas Michael G., Building valid models: how to build valid and credible simulation models, Proceedings of the 2001 Winter Simulation Conference, ACM Press, Arlington, VA 2001
162. Lawrence John A., Pasternack Barry A., Applied Management Science, 2nd edition, John Wiley and Sons 2002
163. Lee Jae K., Trippi Robert R., Chun Seok C., Kim Hyun Soo, K-FOLIO: Integrating the Markowitz Model with a Knowledge Based System, *Journal of Portfolio Management*, Vol. 17, No. 1, Institutional Investor, Inc. Journals Group 1990
164. Lee Jae Kyu, Kwon Suh B., ES: An Expert Systems Development Planner Using a Constraint and Rule-Based Approach, *Expert Systems with Application*, Vol. 9, Iss. 1, Elsevier Science Ltd. 1995
165. Lee Shie-Jue., Wu Chih-Hung, CLXPRT: A Rule-Based Scheduling System, *Expert Systems with Application*, Vol. 9, No. 2, Elsevier Science Ltd. 1995
166. Lee Tian-Shyug, Chiu Chih-Chou, Lu Chie-Jie, Chen I-Fei, Credit scoring using the hybrid neural discriminant technique, *Expert Systems with Application*, Vol. 23, Pergamon, Elsevier Science Ltd. 2002
167. Lees Brian, Kumar B., Mathew A., Corchado Juan M., B. Sinha, Pedreschi Remo, A hybrid case-based neural network approach to scientific and engineering data analysis, Proceedings of the Eighteenth Annual International Conference of the British Computer Society Specialist Group on Expert Systems, Cambridge 1998
168. Leigh William, Purvis Russell, Ragusa James M., Forecasting the NYSE composite index with technical analysis, pattern recognizer, neural network, and genetic algorithm: a case study in romantic decision support, *Decision Support Systems*, Vol. 32, Elsevier Science B.V. 2002

169. Leonard-Barton D., Sviokla J.J., Putting Expert Systems to Work, Harvard Business Review, March-April 1988
170. Leonard-Barton Dorothy, Wellsprings of knowledge, Boston MA: Harvard Business School Press 1995
171. Liang Ting-Peng, User Interface Design for Decision Support Systems: A Self Adaptive Approach, Information and Management, Vol. 12, Iss. 4, Elsevier Science Publishers BV, North-Holland 1987
172. Liebowitz Jay and Letsky Christine, Developing Your First Expert System, CD ROM. International Society for Intelligent Systems, Rockville, MD 2000
173. Liebowitz Jay, Expert Systems: a short introduction, Engineering Fracture Mechanics, Vol. 50, No. 5/6, Elsevier Science Ltd. 1995
174. Liebowitz Jay, Knowledge Management (Knowledge Sharing): A Quick Tutorial and A Look at KM Activities at GSFC 1998, URL: http://seacd.gsfc.nasa.gov/SE_Seminar/ses_data_2002/020205_Liebowitz.ppt (obecnie niedostępne)
175. Liebowitz Jay, Suen C.Y., Developing knowledge management metrics for measuring intellectual capital, Journal of Intellectual Capital, Vol. 1, No. 1, MCB University Press 2000
176. Liebowitz Jay, Knowledge management and its link to artificial intelligence, Expert Systems with Application, Vol. 20, Pergamon, Elsevier Science Ltd. 2001
177. Lin Chn-Teng, Neural fuzzy control systems structure and parameter learning, World Scientific Co. Ltd 1994
178. Podstawowe metody identyfikacji obiektów, (Czarny T., red.) Metody identyfikacji systemów produkcyjnych. Skrypt, Politechnika Wrocławska 1979
179. Lucas Peter, Gaag Linda V., Principles of Expert System, Reading MA, Addison-Wesley 1992
180. Lyneis J.M., Pugh A.L., Automated vs. 'hand' calibration of system dynamics models: An experiment with a simple project model, Proceedings of the 1996 Int. System Dynamics Conference, G.P. Richardson and Sterman Jim D. (red.), System Dynamics Society, Cambridge, MA 1996
181. Łazar Dariusz, Smolak M., Hurtownie danych, Digital Forum 19, 1996
182. Malhotra Yogesh, Business Process Redesign: An Overview, IEEE Engineering Management Review, Vol. 26, No. 3, 1998, URL: <http://www.brint.com/papers/bpr.htm>
183. Malhotra Yogesh, Expert Systems for Knowledge Management: Crossing The Chasm Between Information Processing and Sense Making, Expert Systems with Application, Vol. 20, Iss. 1, Pergamon, Elsevier Science Ltd. 2001
184. Mannino Michael V., Koushik Murlidhar V., The cost-minimizing inverse classification problem: a genetic algorithm approach, Decision Support Systems, Vol. 29, Iss. 3, Elsevier Science B.V. 2000
185. Mańczak Kazimierz, Technika planowania eksperymentu, Warszawa, WNT 1976
186. Markowitz Harry M., Portfolio Selection: Efficient Diversification of Investments (2nd. ed.), Blackwell Publisher 1991
187. Martin J., Dialog człowieka z maszyną cyfrową, WNT Warszawa 1976
188. Matsatsinis Nikolaos F., Doumpos, M., Zopounidis Constantin, Knowledge Acquisition and Representation for Expert Systems in the Field of Financial Analysis, Expert Systems with Application, Vol.: 12, Iss.:2, Elsevier Science Ltd. 1997
189. Matsatsinis Nikolaos F., Samaras Andreas P., Brand choice model selection based on consumers' multicriteria preferences and experts' knowledge, Computers & Operations Research 27 (2000), Pergamon, Elsevier Science Ltd. 2000
190. McConnelly Keith, Introduction to IDEF Modeling: Function and Information Modeling , URL: <http://www.defenselink.mil/c3i/bpr/bprcd/0066.htm>
191. Mechtov A.I., Moshkovich H.M., Knowledge Acquisition Tool for Case-Based Reasoning Systems, Expert Systems with Application, Vol. 9, No. 2, 1995
192. Medsker Larry R., Hybrid intelligent systems, Kluwer Academic Publishers, Boston 1995

193. Mejía-Lavalle Manuel, Rodríguez-Ortiz Guillermo, Obtaining expert system rules using data mining tools from a power generation database, *Expert Systems with Application*, Vol. 14, Elsevier Science Ltd. 1998
194. Metropolis Nicholas, Ulam Stanisław, The Monte Carlo method, *Journal of the American Statistical Association*, Vol. 44 (247), 1949
195. Metzler Douglas P., Martincic Cynthia J., QUE:Explanation through Exploration, *Expert Systems with Application*, Vol. 15, Pergamon, Elsevier Science Ltd. 1998
196. Meyer Marc H., Curley Kathleen Foley, An Applied Framework for classifying the complexity of knowledge-based systems, *MIS Quarterly*, Vol. 15, 1991
197. Meyer Ralph, Guessoum Ahmed, Abduction and conceptual filters for pragmatic reasoning in dialogue systems, Jokinen Kristina (red.) *Proceedings of the XIVth Scandinavian Conference of Linguistics, Workshop on Pragmatics in Dialogue Management*, Publications of the University of Göteborg 1998
198. Michalewicz Zbigniew, *Genetic algorithms + data structures = evolution programs* (second edition), Berlin: Springer-Verlag 1992
199. Michalski Ryszard S., Mozetic Igor, Hong Jiarong, Lavrac Nada, The multipurpose incremental learning system AQ15 and its testing application on tree medical domains, *Proceedings of the national conference on artificial intelligence*, San Mateo, CA: Morgan Kaufmann 1986
200. Miłosz Marek, *Metodologia projektowania technicznego*, Lublin 1998, URL: <http://pluton.pol.lublin.pl/~marekm/metodol/index.htm>
201. Minsky Marvin, *A Framework for Representing Knowledge*, (Winston P., red.) *The Psychology of Computer Vision*, McGraw-Hill 1975, URL: <http://web.media.mit.edu/~minsky/papers/Frames/frames.html>
202. Monostori Laszlo, Connectionist and neuro-fuzzy techniques in intelligent manufacturing, *Proceedings Of the First World Congress on Intelligent Manufacturing Processes & Systems*, Vol. II, Mayaguez, Porto Rico 1995
203. Monostori Laszlo, Szelke Elizabeth, Kadar Botond, *Management of Changes and Disturbances in Manufacturing Systems*, *Annual Reviews in Control*, Vol. 22, Pergamon Press 1998, Elsevier Science Ltd.
204. Monostori Laszlo, *Subsymbolic and Hybrid Artificial Intelligence Techniques in Financial Engineering*, *ERCIM NEWS, Special: Financial Mathematics*, No. 38, European Research Consortium for Informatics and Mathematics 1999, URL: http://www.ercim.org/publication/Ercim_News/enw38/monostori2.html
205. Mookerjee Vijay S., Chaturvedi Alok R., *A Blackboard Control Architecture for Model Selection and Sequencing*, *European Journal of Information Systems*, Vol. 2, No. 1, Elsevier Science B.V. 1993
206. Moore James S., *An Expert System Approach to Graduate School Admission Decisions and Academic Performance Prediction*, *Omega International Journal Management Science*, Vol. 26, Iss. 5, Elsevier Science Ltd. 1998
207. Moore Jeffrey H., Weatherford Larry R., *Decision Modeling with Microsoft Excel*, Prentice Hall 2001
208. Morris William T., *Management Science*, Prentice-Hall, Englewood Cliffs 1968
209. Mulawka Jan J., *Systemy eksperckie*, WNT Warszawa 1996
210. Myers Brad A., *User Interface Tools: Introduction and Survey*, *IEEE Software* '89, Vol. 6, No. 1
211. Myerson Roger B., *Probability Models for Economic Decisions*, Duxbury Press (w przygotowaniu), URL: <http://home.uchicago.edu/~rmyerson/prbook.htm>
212. Najim Kaddour, Rusnák Anton, Mészáros Alojz, Fikar Miroslav, *Constrained Long-Range Predictive Control Based on Artificial Neural Networks*, *International Journal of Systems Science*, Vol. 28, No. 12, Taylor and Francis Ltd 1997
213. Nedović Ljubica, Devedžić Vladan, *Expert Systems in Finance – a Cross-section of the Field*, *Expert Systems with Application*, Vol. 23, No. 1, Elsevier Science Ltd. 2002, URL: [http://fon.fon.bg.ac.yu/~devedzic/Expert Systems with Application2002-1.pdf](http://fon.fon.bg.ac.yu/~devedzic/Expert%20Systems%20with%20Application2002-1.pdf)

214. von Neumann John, Morgenstern Oskar, Theory of Games and Economic Behavior, Princeton, NJ: Princeton University Press 1953
215. Nęcka Edward, Inteligencja. Geneza, struktura, funkcje, Gdańskie Wydawnictwo Psychologiczne, Gdańsk 2003
216. Nguyen Dat-Dao, On A Unified Framework For Building Neural Expert Systems, Conference Proceedings Of Western Decision Sciences Institute, Reno, Nevada 1998
217. Niederliński Antoni, Regułowe systemy eksperckie, Wydawnictwo Pracowni Komputerowej Jacka Skalmierskiego, Gliwice 2000
218. Nielsen Francois, Formal Organizations & Bureaucracy Decision Making Processes, University of Northern Carolina 2001, URL: <http://www.unc.edu/~nielsen/socil10/nm11/m11004.gif>
219. Nonaka Ikujiro, The knowledge creating company, Harvard Business Review 1991, November-December, Vol. 96
220. Nowak Wojciech, Aktualny stan zastosowań sztucznej inteligencji w praktyce i nauczaniu rachunkowości, Rachunkowość, 297-299, nr XLVII, 1996
221. O'Keefe Robert M., Artificial Intelligence and the management science practitioner: Expert systems and MS/OR methodology (good news and bad), Interfaces, Vol. 18, Iss. 6, The Institute of Management Sciences 1988
222. Odum Howard T., Ecological and General Systems: An Introduction to Systems Ecology, University Press of Colorado, Revised edition 1994
223. Odum Howard T., Odum Elisabeth C., Modeling for all Scales: An Introduction to System Simulation, Academic Press 2000
224. OLAP Council 1997, Definitions, URL: <http://dssresources.com/glossary/olaptrms.html>
225. Oliva Rogelio, Model Calibration as a Testing Strategy for System Dynamics Models, European Journal of Operational Research, Vol. 151, No. 3, Elsevier Science BV 2003
226. Encyklopedia WIEM 2005, ONET.PL, URL: <http://portalwiedzy.onet.pl>
227. Ortega Enrique, Computer Minimodels and Simulation Exercises for Science and Social Science, University of Campinas 2005, URL: <http://www.unicamp.br/fea/ortega/>
228. Palisade @RISK 2005, URL: <http://www.palisade-europe.com/html/software.html>
229. Paluszyński Witold, Problemy z wnioskowaniem opartym na logice, Politechnika Wrocławska 2003, URL: sequoia.ict.pwr.wroc.pl/~witold/aiuwr/nonmonotonic_s.pdf
230. Pańkowska Małgorzata, Zastosowanie systemów informatycznych informatycznych bazą wiedzy w planowaniu strategicznym w banku, Bank i Kredyt, Grudzień 1995
231. Paredis Jan, Co-evolutionary constraint satisfaction, Proceedings of the Third Conference on Paralell Problems from Nature, Springer Verlag 1994
232. Pasewark William R., Welker Robert B., A Vroom-Yetton evaluation of subordinate participation in budgetary decision making, Journal of Management Accounting Research Volume 2, Management Accounting Section of the American Accounting Association 1990
233. Patzak Gerold, Systemtechnik - Planung komplexen innovativer Systeme, Grundlagen, Methoden, Techniken, Springer Verlag 1982, Berlin
234. Pepyne David L., Cassandras Christos G., Optimal Control of Hybrid Systems in Manufacturing, Proceedings of the IEEE, Vol. 88, No. 7, IEEE 2000, URL: <http://vita.bu.edu/cgc/Published/PROCprint0800.pdf>
235. Polanyi Michael, The Tacit Dimension, Routledge and Kegan Paul Ltd, London 1967
236. Powell David J., Skolnick Michael, Tong Shi Shing, 1995, A unified approach for engineering design, Intelligent Hybrid Systems, Goonatilake S., Treleaven P. (red.), John Wiley, London 1995
237. Power Daniel, A Framework for Computerized Decision Support, University of Northern Iowa 2001, URL: <http://www.uni.edu/dss/courses/150080/slides/slides2001/index2001.htm>
238. Prerau David S., Developing and managing expert systems, New York, Addison Wesley 1990
239. Principia Cybernetica Web, Web Dictionary of Cybernetics and Systems 1986, URL: <http://pespmc1.vub.ac.be/ASC/indexASC.html>
240. Prusak Laurence, The knowledge advantage, Strategy and Leadership, Vol. 24, No. 2, Emerald 1996

241. Qi Min, Financial applications of artificial neural networks, (Maddala G.S., Rao C.R. red.), Handbook of Statistics. Statistical Methods in Finance, Vol. 14, North-Holland Elsevier Science Publishers, Amsterdam 1996
242. Yu Qian, Xiuxi Li, Yanrong Jiang, Yanqin Wen, An expert system for real-time fault diagnosis of complex chemical processes, Expert Systems with Application, Vol. 24, Elsevier 2003
243. Quah Tong-Seng, Tan Chew-Lim, Raman Krishnamurthy S., Srinivasan Bobby, Towards integrating rule-based expert systems and neural networks, Decision Support Systems, Vol. 17, Elsevier Science Ltd. 1996
244. Quinlan John Ross, Learning efficient classification procedures and their application to chess and games, Machine Learning: an Artificial Intelligence Approach, Vol. 1, Tioga, Palo Alto CA 1983
245. Quinlan J. Ross, Learning logical definitions from relations, Machine Learning, Vol. 5, Kluwer Academic Publishers 1990
246. Radościński Edward, Systemy informatyczne w dynamicznej analizie decyzyjnej, PWN 2001
247. Radhaakrishna Rao C., Modele liniowe statystyki matematycznej, PWN 1982
248. Rao Lei, Goldsman David, Owen Larry, Development And Application Of A Validation Framework For Traffic Simulation Models, Proceedings Of The Winter Simulation Conference, ACM Press New York 1998
249. Rathnam Sukumar, Mannino Michael V., Tools for building the human-computer interface of a decision support systems, Decision Support Systems, Vol. 13, Elsevier Science B.V., North-Holland 1995
250. Ringuest Jeffrey L., A chi-square static for validating simulation-generated responses, Computers and Operations Research. Volume 13, Iss. 4, Elsevier Science Ltd. 1986
251. Roos Johan, Roos Goran, Dragonetti Nicola Carlo, Edvinsson Leif, Intellectual Capital: Navigating in the New Business Landscape, New York University Press 1998
252. Rosch Eleanor H., Principles of Categorization, Cognition and Categorization, Erlbaum, Hillsdale 1978
253. Ruggiero Murray A. jr., How to Build an Artificial Trader, Futures: Management Commodities & Options, Vol. 23, September, Cedar Falls, Iowa 1994
254. Rutkowski Ireneusz P., Zastosowanie hybrydowego inteligentnego systemu informacyjnego w procesie rozwoju strategii marketingowej, Świat marketingu, Styczeń 2002, Katedra Strategii Marketingowych Akademii Ekonomicznej w Poznaniu,
URL: http://www.swiatmarketingu.pl/publikacje/publ2001_1/i_rutkowski_1.htm
255. Rynkiewicz Piotr, Informatyka w controllingu, Trzymać rękę na pulsie, PCkurier 1/2002, Wydawnictwo Lupus
256. Sage Andrew P., Decision support systems engineering, New York, Wiley 1991
257. Salchenberger Linda M., Cinar E. Mine, Lash Nicholas, Neural Networks: a new tool for predicting thrift failures, Decision Sciences, Vol. 23, Blackwell Publishing 1992
258. Sankar Chetan S., Ford Nelson F., Bauer Michael, A DSS user interface model to provide consistency and adaptability, Decision Support Systems, Vol. 13, Elsevier Science B.V., North-Holland 1995
259. Sant Tom, Four Types of Decision Makers: How to Reach Them, Company Coach 2003, <http://www.companycoach.com/MTM/Messages16.htm>
260. Sarle Warren S., What is backprop, ai-faq/neural-nets 2002,
URL: ftp://ftp.sas.com/pub/neural/FAQ2.html#A_backprop
261. Savolainen Tapani, Salminen Veikko, Rough modeling methodology for plant projects, Workshop on CIM in Process and Manufacturing Industries, Helsinki 1992, The Finnish Society of Automation
262. Allan Schick, Gawthrop, Louis C. (red), The Administrative Process and Democratic Theory, Houghton Mifflin 1970
263. Scrivener Stephen A.R., Wilson Paul, Computer-Supported Cooperative Work: The Multimedia and Networking Paradigm, Unicom Applied Information Technology, Brookfield, Vermont 1994
264. Selig William John, Johannes James D., Reasoning Visualization in Expert Systems: The Applicability of Algorithm Animation Techniques, Proceedings of the Third International

- Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems, Charleston, SC 1990
265. Shah Shantilal, Data Visualization, Norfolk State University 1999
URL: <http://www.cs.nsu.edu/others/research/shah/>
 266. Shannon R.E., Systems Simulation: The Art and Science, Prentice-Hall, Englewood Cliffs, NJ, 1975
 267. Shao Yuan Pu, The infusion of Expert Systems In Banking: An Explanatory Study, Expert Systems with Application, Vol. 12, No. 4, Elsevier Science Ltd. 1997
 268. Shao Jianmin, Application of an artificial neural network to improve short-term road ice forecasts, Expert Systems with Application, Vol. 14, Elsevier Science Ltd. 1998
 269. Shepherd Anne. Interactive implementation: promoting acceptance of expert systems, Journal of Computers, Environment, and Urban Systems, Vol. 21, No. 5, Elsevier Science Ltd. 1997
 270. Shim J.P., Warkentin Merrill, Courtney James F., Power Daniel J., Sharda Ramesh, Carlsson Christer, Past, present, and future of decision support technology, Decision Support Systems Volume 33, Elsevier Science B.V. 2002
 271. Shocken S., Ariav G., Neural networks for decision support systems: problems and opportunities, Decision Support Systems, Vol. 11, Elsevier Science Ltd. 1994
 272. Sierocki Robert, Wybrane aspekty wdrożenia technologii wielowymiarowej analizy danych OLAP we wspomaganiu controllingu przedsiębiorstwa, Nowoczesne technologie informacyjne w zarządzaniu, AE Wrocław 2002
 273. Silva Ricardo Bezerra de Andrade, Ludermir Teresa Bernarda, Neural network methods for rule induction, Proceedings of the 1999 International Joint Conference on Neural Networks, Washington DC, URL: <http://www.cs.cmu.edu/~rbas/downloads/ijcnn2099.pdf>
 274. Simoff Simeon J., Sudweeks Fay, The Metaphor of the Face as an Interface for Communicating Non-Quantitative Information, Proceedings 1st Australasian User Interface Conference (AUIC'2000), IEEE Press, URL: www.it.murdoch.edu.au/~sudweeks/papers/metaphor.pdf
 275. Simon Herbert A., The New Science of Management Decision, Harper Brothers, New York 1960
 276. Simon Herbert A., Reason in Human Affairs, Stanford University Press 1983
 277. Simon Alan, Better Clients, Better Decisions, BYTE January 1997,
URL: <http://www.byte.com/art/9701/sec7/art3.htm>
 278. Skowronek-Mielczarek Anna, Instrumenty podejmowania decyzji i planowania, (Strużycki M., red.) Zarządzanie przedsiębiorstwem, Difin, Warszawa 2002
 279. Smith Wayne K., Time Out, Using Visible Pull Systems to Drive Process Improvement, John Wiley & Sons, January 1998
 280. Sprague Ralph H., Carlson Eric D., Building Effective Decision Support Systems, Prentice Hall, Englewood Cliffs, NJ 1982
 281. Stanojevic Mladen, Vranes Sanja, Velasevic Dusan, Using Truth Maintenance Systems: A Tutorial, IEEE Intelligent Systems and their applications, Vol. 9, No. 6, IEEE, Inc. 1994
 282. StatSoft Polska, Planowanie doświadczeń. Statistica,
URL: <http://www.statsoft.pl/download/help6/Experimental.chm>
 283. Steyer Jean-Philippe, Rolland Damien, Bouvier Jean-Claude, Moletta René, Hybrid fuzzy neural network for diagnosis - application to the anaerobic treatment of wine distillery wastewater in a fluidized bed reactor, Water Science and Technology Volume 36, No. 6-7, IWA Publishing 1997
 284. Stockdale Peggy, Psychology of Employee Relations, Southern Illinois University, Carbondale 1999, URL: <http://www.siu.edu/departments/cola/psycho/psyc323/chapt12/sld034.htm>
 285. Use of Artificial neural networks and Fuzzy logic for integrated water management, STOWA i IHE Delft Hydroinformatics Joint Research Project 2000, URL: <http://www.stowa-nm.ihe.nl/>
 286. Strużycki Marian, Istota i struktura otoczenia przedsiębiorstwa, (Strużycki M., red.) Zarządzanie przedsiębiorstwem, Difin, Warszawa 2002
 287. Subramanian Girish H., Yaverbaum Gayle J., Brandt Stephen J., An Empirical Evaluation of Factors Influencing Expert Systems Effectiveness, Journal of Systems Software, Vol. 38, Elsevier Science Publishing Co., Inc., New York 1997

288. Summers Barbara, Riding Allan L., Networks That Learn and Credit Evaluation, ICSB Small Business Advancement National Center, University of Central Arkansas 1996
289. Sveiby Karl Erik, The New Organizational Wealth: Managing and Measuring Intangible Assets, Berrett-Koehler Publishers Inc. 1997
290. Swain James J., Simulation goes mainstream, OR/MS Today October 1997
291. Swain James J., Power Tools for Visualization and Decision-Making, ORMS Today, February 2001
292. Swain James, Simulation Software Survey, OR/MS Today, Institute for Operations Research and the Management, August, Lionheart Publishing 2003,
URL: <http://lionhrtpub.com/orms/surveys/Simulation/Simulationmain.html>
293. Szapiro Tomasz (praca zbiorowa), Decyzje menedżerskie z Excelem, Polskie Wydawnictwo Ekonomiczne 2000
294. Szapiro George G., A Search for Hidden Relationships: Data Mining with Genetic Algorithms, Computational Economics, Vol. 10, Kluwer Academic Publisher 2000
295. Thomas Lyn C., Crook Jonathan N. & Edelman David B., Credit Scoring and Credit Control, Proceedings of the International Conference. Edited with Crook Jonathan N., Edelman David B., Oxford University Press 1992
296. Thomsen Erik, OLAP Solutions. Building Multidimensional Information Systems, John Wiley & Sons, New York 1997
297. Todd Peter, Benbasat I., An Experimental Investigation Of The Impact Of Computer Based Decision Aids On Processing Effort, MIS Quarterly, Vol. 16, No. 3, 1992
298. DuToit S.H.C., Steyn A.G.W., Stumpf R.H., Graphical Exploratory Data Analysis, Springer-Verlag, New York, 1986
299. Trippi Robert, Turban Efraim, The Impact of parallel and neural computing on managerial decision making, Journal of Management Information Systems, Vol. 6, No. 3, Bentley College 1989
300. Turban Efraim, Decision Support Systems and Expert Systems: Managerial Perspectives, MacMillan, New York 1988
301. Turban Efraim, Expert Systems And Applied Artificial Intelligence, MacMillan Publishing Company 1993
302. Turban Efraim, Decision Support and Expert Systems. Management Support Systems, MacMillan Publishing Company, New York 1993
303. Turban Efraim, Rainer Rex Kelly, Potter Richard E., Introduction to Information Technology, John Wiley & Sons 2003,
URL: http://www.uni.edu/dss/courses/150080/slides/slides2001/Chapter10_files/frame.htm
304. Tutaj Rafał, Wizualizacja przemysłowa – Aktualne trendy i wymagania, Prace naukowe ITMiA Politechniki Wrocławskiej, Nr 67, Seria: Konferencje Nr 29, Automatykacja Produkcji'97, Wrocław 1997
305. Tyran Craig K., George Joey F., The Implementation of Expert Systems: A Survey of Successful Implementation, Data Base, Vol. 24, Iss. 1, 1993
306. Tzafestas Spyros G., Stamou Giorgos B., An improved neural network for fuzzy reasoning implementation, Mathematics and Computers in Simulation, Vol. 40, Special Issue on neural networks/neural computing, Elsevier Science Publishers B.V. 1996
307. Ultsch Alfred, Guimaraes Gabriela, Weber Volker, Self Organizing Feature Maps for Logical Unification, Proceedings of WCES, Lisboa, Januar 1994, URL: <http://www.mathematik.uni-marburg.de/~databionics/de/downloads/papers/ultsch94logic.pdf>
308. Ultsch Alfred, The Integration of Neural Networks with Symbolic Knowledge Processing, (Diday i inni, red.) New Approaches in Classification and Data Analysis, Springer Verlag 1994
309. Ultsch Alfred, The Integration of Connectionist Models with Knowledge-based Systems: Hybrid Systems, Proceedings of the IEEE SMC 98 International Conference, San Diego 1998
310. van Heijst Gertjan, Schreiber A. Th., Wielinga Bob J., Using explicit ontologies in KBS development, International Journal of Human-Computer Studies, Vol. 46, Iss. 2-3, Elsevier Ltd. 1996

311. van Horn, Richard L., Validation of Simulation Results, *Management Science*, Vol. 17, No. 5 January, 1971
312. Vanthienen Jan, Wets Geert, Integration of the Decision Table Formalism with a Relational Database Environment, *Information Systems*, Vol. 20, No. 7, pp. 595-616, Pergamon, Elsevier Sciences Ltd. 1995
313. Venter F.J., Oosthuizen G.D., Ross J.D., Knowledge Discovery in Databases using Lattices, *Expert Systems with Application*, Vol. 13, No. 4, Elsevier Science Ltd. 1997
314. Verrijin-Stuart Alexander, A formal view of interfaces in a multi-user Environment, Proceedings of the IFIP W8.4 Conference on Multi-User Interfaces and Applications, Heraklion Crete, Greece, North-Holland 1990
315. Vroom Victor H., Yetton Philip W., Leadership and Decision Making, University of Pittsburgh Press 1973
316. Wang Huaiqing, Knowledge level analysis of group decision support systems, *Expert Systems with Application*, Vol. 13, No. 2, Elsevier Science Ltd. 1997
317. Ward Matthew O., LeBlanc Jeffrey T., Tipnis Rajeev, N-Land: a Graphical Tool for Exploring N-Dimensional Data, Computer Graphics International Conference in Melbourne, Australia 1997, URL: <http://davis.wpi.edu/~matt/courses/nland/cgi93.html>
318. Watkins Fred A., Fuzzy Logic Record, Session 8PD, Vol. 26, No. 1, Las Vegas Spring Meeting 2000, May 22–24, URL: (obecnie niedostępna)
319. Web Dictionary on Cybernetics and Systems Theory, International Institute for Applied Systems Analysis (IIASA), Principia Cybernetica Web, Brussels i Los Alamos 2000, URL: http://pespmc1.vub.ac.be/ASC/DECISI_THEOR.html
320. Webb Geoffrey I, Wells Jason, Recent progress in machine-expert collaboration for knowledge acquisition, *Artificial Intelligence*, Eight Australian Joint Conference on Artificial Intelligence, World Scientific, Singapore 1995
321. Weiss Sholom M., Kulikowski Casimir A., Computer Systems that Learn: Classification and Prediction Methods from Statistics, Neural Networks, Machine Learning and Expert Systems, San Mateo CA, Morgan Kaufmann Publisher 1991
322. Wermter Stefan, Sun Ron, An overview of Hybrid Neural System, *Hybrid Neural Systems*, Springer, Heidelberg, New York 2000, URL: <http://www.his.sunderland.ac.uk/ps/intro.pdf>
323. Wiggins Lyna L., Kim T.John, Jain Ravinder, Evaluating an Expert System in the field: Experience with the CORA expert system, *Expert systems in environmental planning*, New York: Springer-Verlag 1993
324. Scientific method, URL: http://en.wikipedia.org/wiki/Scientific_method
325. Winton Charles, Discrete Systems Modeling and Simulation, University of North Florida 2001, URL: <http://www.unf.edu/ccec/cis/cwinton/html/cop4300/s01/4300menu.html>
326. Wolpert, David H., Macready William G., The No Free Lunch Theorems For Optimization, *IEEE Transaction on Evolutionary Computation*, Vol. 1, No. 1, 1997
327. Wong Man Leung, An adaptive knowledge-acquisition system using generic genetic programming, *Expert Systems with Application*, Vol. 15, Elsevier Science Ltd. 1998
328. Wright George, Ayton Peter, Eliciting and Modelling Expert Knowledge, *Decision Support Systems*, Vol. 3, No. 1, Elsevier Science B.V. 1987
329. Wrycza Stanisław, *Informatyka dla ekonomistów*, Wydawnictwo Uniwersytetu Gdańskiego 1993
330. Xindong Wu, Explicit schematic information in knowledge representation and acquisition, *Expert Systems with Application*, Vol. 15, Pergamon, Elsevier Science Ltd. 1998
331. Yates Frank, Complex experiments, *Journal of the Royal Statistical Society (Supplement) 2*, 181, 1935
332. Yoon Youngohc, Guimaraes Tor, O'Neal Quinton, Exploring the Factors Associated With Expert Systems Success, *MIS Quarterly*, March 1995
333. Yourdon Edward, *Modern Structured Analysis*. Yourdon Press, Prentice-Hall, Englewood Cliffs, 1989

334. Zabawa Jacek, Kopiński Adam, Niektóre przesłanki stosowania metod i narzędzi systemów eksperckich w pewnych obszarach zarządzania, (red. Abramowicz W.) Business Information Systems'98, Akademia Ekonomiczna w Poznaniu
335. Zabawa Jacek, Radosiński Edward, Interface w hybrydowych inteligentnych systemach wspomaganie decyzji, Symulacja Systemów Gospodarczych Duszniki 2002, Wydawnictwo Politechniki Wrocławskiej
336. Zabawa Jacek, Wybrane techniki poszukiwania optymalnych rozwiązań w symulacyjnej analizie ekonomicznej przedsiębiorstwa, Symulacja systemów gospodarczych. Część II, Oficyna Wydawnicza Politechniki Wrocławskiej 2004
337. Zerka Marek, Zarządzanie ryzykiem na konkurencyjnym rynku energii elektrycznej, Agencja Rynku Energii S.A. 2001, URL: http://cire.pl/pdf.php?plik=/publikacje/Art_Zerka.pdf
338. Zhang Zili, Zhang Chengqi, An Agent-Based Hybrid Intelligent System for Financial Investment Planning, Proceedings of PRICAI 2002, LNAI-2417, Springer, URL: http://www3.cm.deakin.edu.au/~zzhang/publication_files/pricai2002.pdf
339. Zhao Jun, Knight Brian, Nissan Ephraim, Soper Alan, FuelGen: A Genetic Algorithm-based System for Fuel Loading Pattern Design in Nuclear Power Reactors, Expert Systems with Application, Vol. 14, Elsevier Science Ltd. 1998
340. Zieliński Ryszard, Generatory liczb losowych, WNT Warszawa 1972
341. Zlatareva Neli P., A refinement framework to support validation and maintenance of knowledge-based systems, Expert Systems with Application, Vol. 15, Pergamon, Elsevier Science Ltd. 1998
342. Zopounidis Constantin, Doumpos Michael, Matsatsinis Nikolaos F., On the use of knowledge-based decision support systems in financial management: A survey, Decision Support Systems, Vol. 20, Elsevier Science Ltd. 1997