B A D A N I A O P E R A C Y J N E I D E C Y Z J E

Nr 3–4 2005

Radosław JADCZAK*

ROZWIĄZYWANIE ZAGADNIEŃ UKŁADANIA TRAS POJAZDÓW Z WYKORZYSTANIEM ALGORYTMÓW EWOLUCYJNYCH

W artykule poruszono zagadnienie układania tras pojazdów, znane także w literaturze jako problem wielu komiwojażerów. Zagadnienie to, należące do problemów *NP*-zupełnych, można łatwo sformułować, jednak znalezienie jego rozwiązania optymalnego jest bardzo trudne. Zaprezentowano możliwość wykorzystania algorytmów ewolucyjnych, które – w przeciwieństwie do klasycznych metod z dużą efektywnością – przeszukują przestrzeń rozwiązań rozpatrywanego zagadnienia.

Słowa kluczowe: *problem układania tras pojazdów, algorytmy ewolucyjne*

Wstęp

Wiele przedsiębiorstw produkcyjnych i handlowych stoi przed problemem właściwej organizacji dystrybucji swoich wyrobów do sieci klientów. W badaniach operacyjnych problem ten jest określany mianem zagadnienia układania tras pojazdów (ang. *Vehicle Routing Problem*). Cechą charakterystyczną problemów układania tras pojazdów jest łatwość sformułowania konkretnego problemu, w przeciwieństwie do jego rozwiązania. Stąd też zagadnienia te interesują wielu badaczy, czego wynikiem jest znaczna liczba propozycji algorytmów poszukiwania najlepszych rozwiązań.

Interesującą alternatywę dla dotychczas najbardziej popularnych metod rozwiązywania zadań układania tras pojazdów stanowią algorytmy ewolucyjne. Są to metody przeznaczone przede wszystkim do rozwiązywania zadań optymalizacji, jednak wydają się być szczególnie użyteczne w przypadku zagadnień o charakterze kombinatorycznym.

* Katedra Badań Operacyjnych, Uniwersytet Łódzki, ul. Rewolucji 1905 r. 41, 90-214 Łódź, rjadczak@pai.net.pl

W pierwszej i drugiej części niniejszej pracy przedstawiono odpowiednio: sformułowanie problemu układania tras pojazdów oraz znane dotychczas metody jego rozwiązywania. Następnie zaprezentowano algorytm ewolucyjny dla zadania układania tras pojazdów. W ostatniej części pracy o charakterze empirycznym przedstawiono wyniki, uzyskane za pomocą algorytmu ewolucyjnego, będące podstawą do rozwiązania kilku zadań testowych.

1. Sformułowanie problemu układania tras pojazdów

Najczęściej podejmowanym w literaturze zagadnieniem układania tras pojazdów jest problem jednego komiwojażera, znany także pod nazwą problemu podróżującego sprzedawcy (ang. *Traveling Salesman Problem*). Definicję zagadnienia, przedstawionego po raz pierwszy przez W.R. Hamiltona w 1859 roku, można podać na przykładzie pracy przedstawiciela handlowego w sposób następujący. Przedstawiciel handlowy pewnego przedsiębiorstwa wyjeżdża ze swojego biura, aby odwiedzić pewną zadaną liczbę swoich klientów. Musi odwiedzić wszystkich klientów i dokładnie jeden raz, a następnie powrócić do biura. Kolejność wizyt u poszczególnych klientów jest dowolna, jednak celem przedstawiciela handlowego jest wybranie takiej trasy, aby była ona jak najkrótsza.

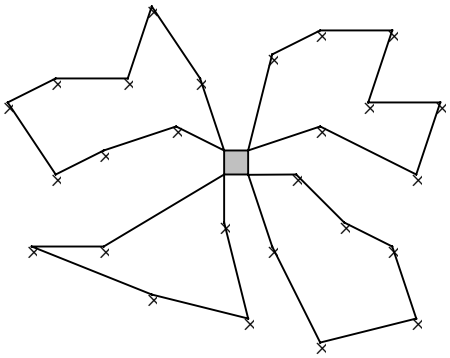
Przedstawiony problem komiwojażera ma charakter kombinatoryczny i należy do problemów *NP*-zupełnych [11]. Pomimo bardzo prostej definicji zagadnienia znalezienie optymalnego rozwiązania (najkrótszej trasy) przy znacznej liczbie klientów, których musi odwiedzić przedstawiciel handlowy, jest niezwykle pracochłonne. Jeżeli założymy, że odległość pomiędzy klientem A a klientem B jest inna, niż pomiędzy klientem B a klientem A, to przedstawiciel handlowy odwiedzając tylko 10 klientów może wybrać jedną spośród ponad 3,6 miliona tras. Dla zadanej liczby N odwiedzanych klientów liczba możliwych do pokonania tras wynosi $N!$

Problem jednego komiwojażera jest szczególnym przypadkiem zagadnienia układania tras pojazdów, nazywanego także problemem wielu komiwojażerów, którego sformułowanie można przedstawić w następujący sposób. Załóżmy, że rozpatrujemy pewne przedsiębiorstwo (zakład produkcyjny, hurtownia), które musi każdego dnia dostarczyć do sieci swoich klientów pewne jednorodne dobro o określonej ilości. Przedsiębiorstwo dysponuje własnym taborem transportowym przystosowanym do dystrybucji dobra, w którym każdy pojazd ma ograniczoną ładowność. Zadaniem przedsiębiorstwa jest dostarczenie do wszystkich klientów żądanej ilości dobra w taki sposób, aby spełnione były łącznie następujące warunki (rys. 1):

- każdy klient może być odwiedzony tylko przez jeden pojazd, który dostarczy całkowitą żądaną ilość dobra;

• ładowność każdego użytego pojazdu nie może zostać przekroczona;

• suma kosztów (lub długości) tras, pokonanych przez wszystkie użyte w tym celu pojazdy, musi być jak najmniejsza.



Rys. 1. Przykład rozwiązania dopuszczalnego zadania układania tras pojazdów
Ź r ó d ł o: Opracowanie własne.

W tak sformułowanym zagadnieniu układania tras pojazdów występują dwa problemy optymalizacyjne:

• podział zbioru wszystkich odbiorców dobra na rejony, z których każdy zostanie przypisany do jednego pojazdu, jakim dysponuje przedsiębiorstwo;

• wyznaczenie kolejności dostaw w ramach rejonu, przyporządkowanego do każdego użytego przez przedsiębiorstwo pojazdu.

Zagadnienie układania tras dla pojazdów stanowi punkt wyjścia do formułowania innych problemów, będących modyfikacją zadania podstawowego. Wśród różnych odmian problemu układania tras pojazdów można wyróżnić: zadanie układania tras pojazdów z oknami czasowymi, zadanie układania tras pojazdów z dostawą i odbiorem dobra, stochastyczny problem układania tras pojazdów, czy problem układania tras pojazdów z uzupełnieniami ładunku pojazdu.

2. Metody rozwiązywania zadań układania tras pojazdów

W ciągu ostatnich prawie pięćdziesięciu lat badań nad zagadnieniem układania tras pojazdów zaproponowano wiele metod jego rozwiązania. Podstawowy podział rozróżnia metody dokładne i przybliżone (heurystyczne). Wśród tych pierwszych, pozwalających dokładnie ustalić rozwiązanie optymalne, na uwagę zasługują przede

wszystkim metody realizujące strategię podziału i ograniczeń. Należy do nich procedura J.D. Little'a [12] dla zadania jednego komiwojażera oraz metoda EAAM L.J. Jasińskiego [7] dla problemu wielu komiwojażerów. Ze względu na dużą pracochłonność, bardzo długi czas poszukiwania rozwiązania optymalnego oraz efektywność tych metod tylko w przypadku zadań o niewielkich rozmiarach (małej liczbie punktów dostaw), znacznie większą popularność zyskały sobie metody heurystyczne.

W przeciwieństwie do metod dokładnych, algorytmy przybliżone nie dają pewności uzyskania rozwiązania optymalnego. Uzyskiwane rozwiązanie jest bliskie optymalnemu oraz akceptowalne przez decydenta z punktu widzenia czasu jego osiągania, który jest nieporównywalnie krótszy niż w przypadku metod dokładnych. Dotychczas zaproponowane metody heurystyczne można sklasyfikować w trzech grupach [8], [9]:

- metody konstrukcyjne;
- metody dekompozycyjne;
- metody wzrostu.

Do metod konstrukcyjnych zalicza się zaproponowane algorytmy, które jednocześnie przyporządkowują odbiorców do poszczególnych pojazdów oraz ustalają kolejność dostaw przez dany pojazd. Wśród metod realizujących tą strategię poszukiwania rozwiązania optymalnego można wyróżnić: algorytm *savings* G. Clarke'a i J.W. Wrighta [2], modyfikację tej metody zaproponowaną przez H. Paessensa [15], a także algorytm R.H. Mole'a i S.R. Jamesona [14].

Dwa problemy optymalizacyjne przedstawione w pierwszej części niniejszej pracy, czyli przydział odbiorców do pojazdów oraz kolejność dostaw przez poszczególne pojazdy, są rozpatrywane oddzielnie w przypadku zastosowania metod dekompozycyjnych. Strategia ta może być realizowana w dwóch kierunkach: najpierw rejon dostaw, potem kolejność lub odwrotnie – najpierw kolejność, potem rejon dostaw. W drugim przypadku celem jest budowa jednej niedopuszczalnej trasy dla wszystkich zaopatrywanych przez dostawcę odbiorców, by następnie podzielić ją na mniejsze podtrasy obsługiwane przez poszczególne pojazdy. Reprezentantami tych metod są algorytmy: *sweep* B.E. Gilletta i L.R. Millera [4], M. Fishera i R. Jaikumara [3], w którym wykorzystywane jest zagadnienie uogólnionego przydziału, czy algorytm BF-WOT A. Całczyńskiego [1].

Ostatnia grupa metod heurystycznych to algorytmy wzrostu, znane także pod nazwą metod lokalnej optymalizacji. Opierają się na strategii poszukiwania rozwiązania optymalnego przez zastępowanie aktualnie rozpatrywanego rozwiązania nowym, reprezentującym lepszy układ tras. Przekształcenie aktualnie rozpatrywanego zbioru tras pojazdów w inny może się odbywać w trojaki sposób: przez wymianę podciągów punktów obsługi pomiędzy trasami poprzez skrzyżowanie krawędzi dwóch tras; przez bezpośrednią wymianę punktów obsługi między trasami; przez przeniesienie punktu obsługi z jednej trasy do drugiej.

Odrębną klasę podejść do rozwiązywania zadań układania tras pojazdów stanowią metody oparte na algorytmach ewolucyjnych, sieciach neuronowych oraz algorytmach mrówkowych.

3. Algorytm ewolucyjny dla zadania układania tras pojazdów

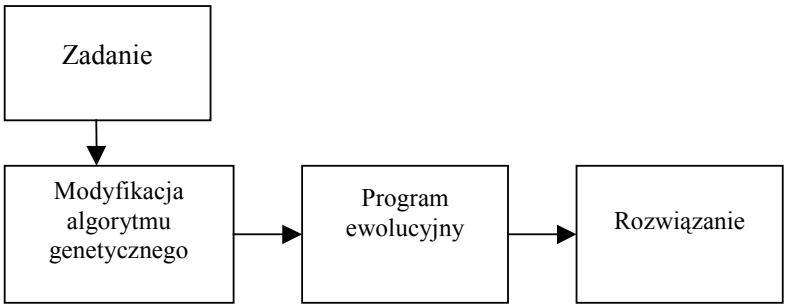
Wśród badaczy zajmujących się problemami optymalizacji coraz większym zainteresowaniem cieszą się algorytmy ewolucyjne, których działanie opiera się na procesach ewolucyjnych zachodzących wśród organizmów żywych. Duże zainteresowanie tymi metodami wynika między innymi z następujących cech, jakimi się one charakteryzują:

- algorytmy ewolucyjne operują nie na rzeczywistych rozwiązaniach badanego problemu, lecz na ich zakodowanych postaciach (chromosomach);
- poszukiwanie rozwiązań optymalnych opiera się przede wszystkim na funkcji przystosowania chromosomów, którą jest zazwyczaj bezpośrednio funkcja kryterium badanego problemu;
- algorytmy ewolucyjne nie wymagają dogłębnej znajomości optymalizowanego problemu, lecz jedynie podstawowej wiedzy na jego temat;
- bez względu na charakter badanego problemu optymalizacyjnego działanie większości algorytmów ewolucyjnych polega na przeprowadzaniu operacji selekcji, mutacji i krzyżowania, w pojedynczej iteracji nazywanej pokoleniem;
- w każdym pokoleniu algorytm rozpatruje nie jedno rozwiązanie, lecz zbiór chromosomów, który nosi nazwę populacji.

Podstawowym i jednocześnie najbardziej znanym algorytmem ewolucyjnym jest klasyczny algorytm genetyczny [5]. Rozwiązania badanych problemów optymalizacyjnych, którymi są zazwyczaj liczby rzeczywiste, są kodowane w postaci ciągu zer i jedynek. Do tak sformułowanego kodowania binarnego wykorzystywane są operatory krzyżowania z jednym lub kilkoma punktami cięcia (wymiana podciągów zer i jedynek pomiędzy chromosomami). Mutacja polega natomiast na zmianie zera w jedynkę (lub odwrotnie) wybranego lub wybranych elementów (genów) ciągu kodowego. W klasycznych algorytmach genetycznych wykorzystywana jest metoda selekcji ruletkowej.

Nie zawsze jest możliwe zastosowanie kodowania binarnego w celu reprezentacji rozwiązań badanego problemu optymalizacyjnego. Dotyczy to zwłaszcza zadań o charakterze kombinatorycznym, do których należy problem komiwojażera oraz problem układania tras pojazdów. Zastosowanie innego sposobu kodowania niż binarne pociąga za sobą także zmianę sposobu krzyżowania i mutacji w taki sposób, aby powstające nowe chromosomy reprezentowały rozwiązania dopuszczalne. Poza tym zaletą

algorytmów ewolucyjnych jest ich duża elastyczność, polegająca na możliwości zastosowania innych sposobów selekcji, jak również dodaniu innych procedur obok krzyżowania i mutacji. Takie dokonywanie modyfikacji klasycznego algorytmu genetycznego w literaturze poświęconej algorytmom ewolucyjnym nosi nazwę programowania ewolucyjnego (rys. 2).

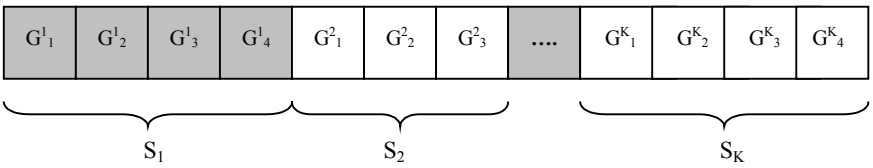


Rys. 2. Schemat zastosowania programu ewolucyjnego
Ź r ó d ł o: [13]

Aby do rozwiązania każdego zadania optymalizacyjnego zastosować algorytm ewolucyjny, należy najpierw określić sposób reprezentacji rozwiązania w postaci chromosomu, a także podstawowe parametry algorytmu.

Do rozwiązania przedstawionego w pierwszej części niniejszej pracy podstawowego zadania układania tras pojazdów zaproponowano program ewolucyjny PEWKOM (Program Ewolucyjny Wielu Komiwojażerów), w którym zastosowano permutacyjny sposób kodowania chromosomów.

Załóżmy, że dostawca ma dostarczyć dobro do N odbiorców, korzystając w tym celu z K dostępnych pojazdów. Oznacza to, że każdy chromosom będzie miał postać wektora składającego się z N genów. Każdy gen będzie reprezentować jednego odbiorcę. Jeśli oznaczymy przez G_i^k i-ty kolejny gen chromosomu, przyporządkowany do k-tego pojazdu, to postać chromosomu możemy przedstawić jak na rys. 3.



Rys. 3. Schemat ciągu kodowego (chromosomu)
dla zadania wielu komiwojażerów programu PEWKOM
Ź r ó d ł o: Opracowanie własne.

Przydział poszczególnych genów (odbiorców) do pojazdów będzie się odbywać począwszy od pierwszego genu chromosomu. Wybór pojazdu natomiast, który będzie odwiedzał kolejne punkty, będzie uwarunkowany jego ładownością. Jako pierwsze wybierzemy do obsługi punktów pojazdy o największych możliwościach transportowych.

Zgodnie z rysunkiem 3, pierwsza część chromosomu (podciąg genów: G_1^1, G_2^1, ..., G_4^1) będzie przyporządkowana do pierwszego pojazdu S_1 (o największej ładowności spośród wszystkich pojazdów S_1, S_2, …, S_k), ponieważ jego ładowność nie pozwala na obsłużenie kolejnego punktu reprezentowanego przez gen G_1^2 W związku z tym gen $G^2{}_1$ wejdzie w skład drugiej części chromosomu: G_1^2, G_2^2, ..., G_3^2, przyporządkowanej do drugiego pojazdu S_2. W ten sposób zostaną przyporządkowane wszystkie punkty obsługi do kolejnych pojazdów.

Wartością funkcji przystosowania chromosomu $f(Ch)$ jest suma długości tras, jaką pokonują użyte pojazdy. Długość trasy T_k pojazdu S_k można zapisać następująco:

$$T_k = \sum_{i=1}^{l-1} d(G_i^k, G_{i+1}^k) + d(B, G_1^k) + d(G_l^k, B),$$

gdzie:

$d(G_i^k, G_{i+1}^k)$ – odległość (koszt) przejazdu pojazdu pomiędzy kolejnymi odbiorcami reprezentowanymi w chromosomie przez geny G_i^k i G_{i+1}^k;

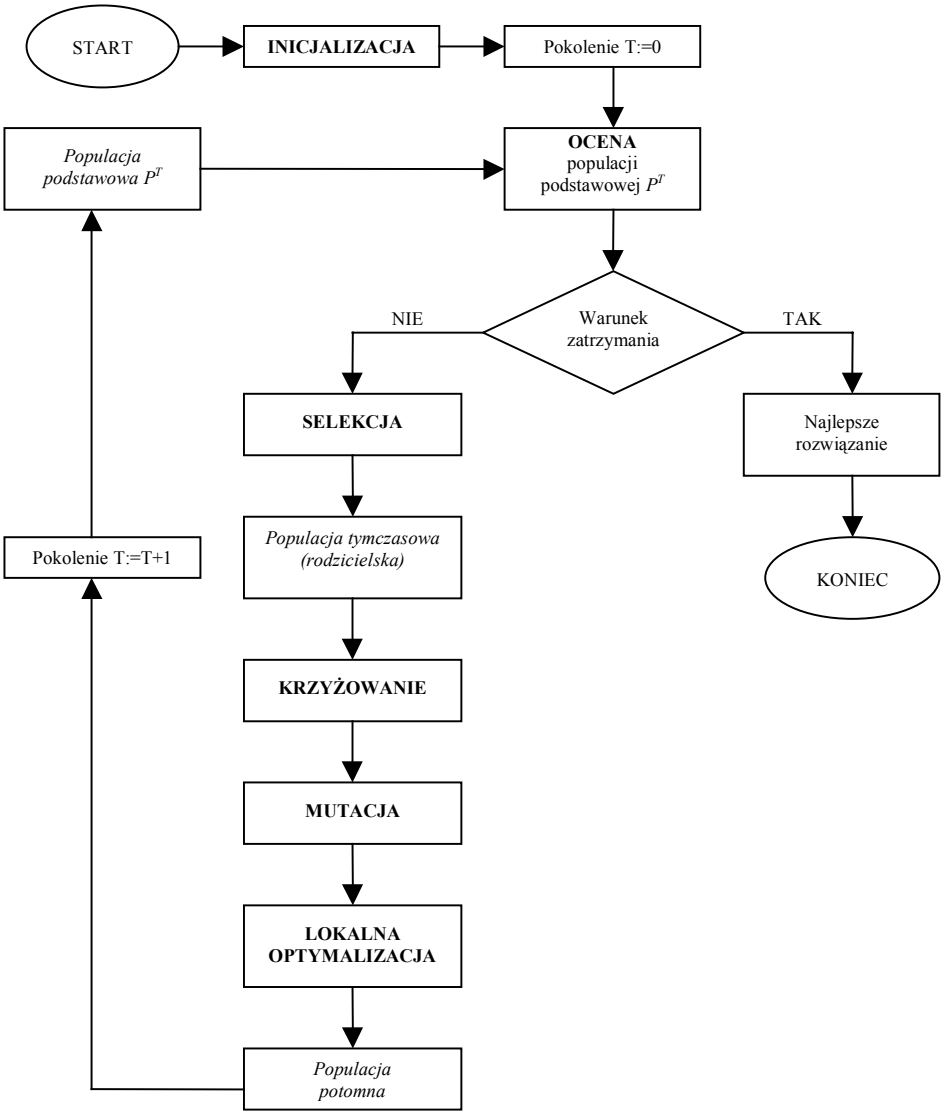
$d(B, G_1^k)$ – odległość (koszt) przejazdu pojazdu pomiędzy dostawcą (bazą) a odbiorcą reprezentowanym w chromosomie przez gen G_1^k;

$d(G_l^k, B)$ – odległość (koszt) przejazdu pojazdu pomiędzy odbiorcą reprezentowanym w chromosomie przez gen G_l^k a dostawcą (bazą).

Następnie należy określić podstawowe parametry działania algorytmu ewolucyjnego. Do parametrów tych należy liczebność populacji, która w każdym pokoleniu jest niezmienna. Natomiast zadana na początku liczba pokoleń stanowi kryterium stopu programu ewolucyjnego.

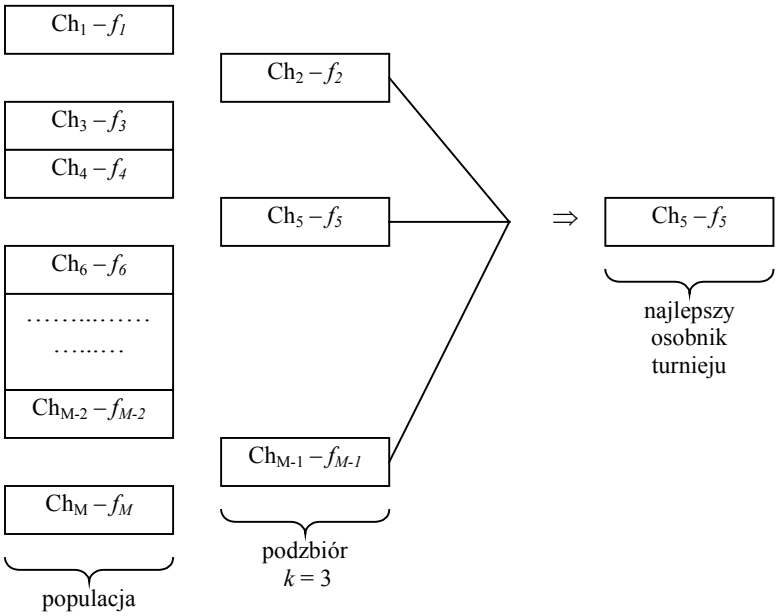
W zaprezentowanym na rysunku 4 programie ewolucyjnym korzysta się zarówno z procesu krzyżowania chromosomów, jak i ich mutacji. Jednak nie zawsze te operatory genetyczne są wykorzystywane. Częstotliwość użycia krzyżowania i mutacji wyznaczają dwa bardzo istotne parametry programu ewolucyjnego, odpowiednio: prawdopodobieństwo zajścia procesu krzyżowania oraz prawdopodobieństwo zajścia procesu mutacji.

Zdefiniowanie sposobu kodowania rozwiązań zadania układania tras pojazdów oraz określenie podstawowych parametrów stanowi etap inicjalizacji prezentowanego programu ewolucyjnego, którego schemat działania przedstawiono na rysunku 4.



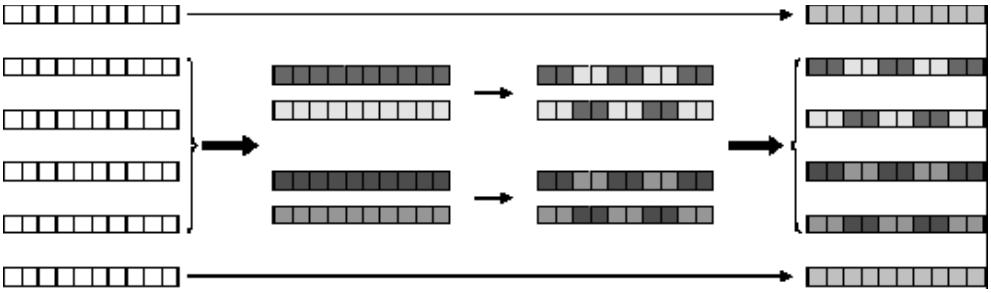
Rys. 4. Schemat działania programu ewolucyjnego PEWKOM
Ź r ó d ł o: Opracowanie własne.

Na początku każdej iteracji programu ewolucyjnego oceniana jest populacja chromosomów. Ocena polega na ustaleniu dla każdego z nich wartości funkcji przystosowania. Wartość ta jest wykorzystywana w pierwszej fazie działania algorytmu, czyli w selekcji. Celem selekcji jest wyłonienie z populacji ocenianej populacji rodzicielskiej o takiej samej liczebności chromosomów. Schemat selekcji przedstawiono na rysunku 5.



Rys. 5. Proces selekcji
Ź r ó d ł o: Opracowanie własne.

Z populacji ocenianej wybierany jest w sposób losowy podzbiór *k* chromosomów, z których w populacji rodzicielskiej umieszczany jest chromosom o najlepszej wartości funkcji przystosowania. W selekcji tej wartość *k* jest parametrem, który stanowi o naporze selekcyjnym w prezentowanym algorytmie ewolucyjnym. Im większa wartość *k*, tym większy jest napór selekcyjny. Proces ten należy powtórzyć tyle razy, ile wynosi liczebność populacji ocenianej, a każdy wybór chromosomu nosi nazwę turnieju.

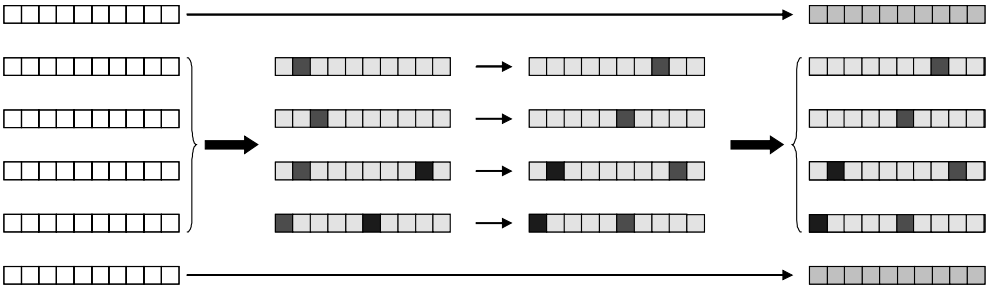


Rys. 6. Proces krzyżowania
Ź r ó d ł o: Opracowanie własne.

Chromosomy populacji rodzicielskiej w następnym etapie poddawane są operacji krzyżowania. Ponieważ nie wszystkie chromosomy mogą zostać poddane krzyżowaniu, co wynika z przyjętego na początku prawdopodobieństwa zajścia krzyżowania, część z nich zostanie więc umieszczona bezpośrednio w nowej populacji tymczasowej. Pozostałe chromosomy zostaną poddane krzyżowaniu, które polega na wymianie materiału genetycznego pomiędzy losowo dobranymi parami (rys. 6).

Do zaprezentowanego sposobu kodowania chromosomów wykorzystano krzyżowanie cykliczne [13], w wyniku którego powstają dwa nowe chromosomy umieszczane w populacji tymczasowej. W krzyżowaniu cyklicznym każdy gen wraz z jego pozycją w ciągu kodowym pochodzi od jednego z pary krzyżowanych chromosomów.

Populacja tymczasowa utworzona częściowo z chromosomów, które nie przeszły procesu krzyżowania oraz z chromosomów utworzonych w wyniku krzyżowania cyklicznego zostaje poddana operacji mutacji. Mutacja, podobnie jak krzyżowanie zachodzi z zadanym na początku prawdopodobieństwem. Podobnie jak w krzyżowaniu mutacja nie może prowadzić do tworzenia chromosomów reprezentujących rozwiązanie niedopuszczalne. Proces mutacji może polegać na przemieszczeniu genów lub ich wymianie (rys. 7).

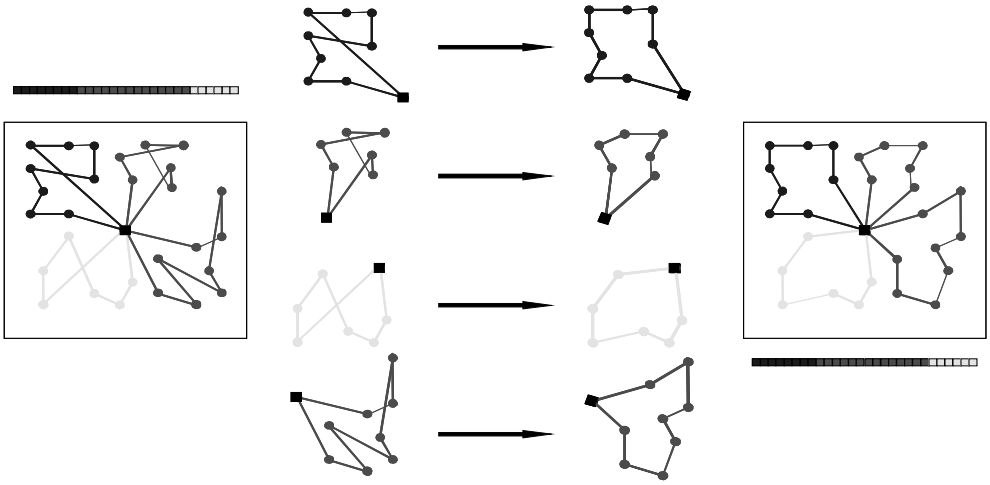


Rys. 7. Proces mutacji
Ź r ó d ł o: Opracowanie własne.

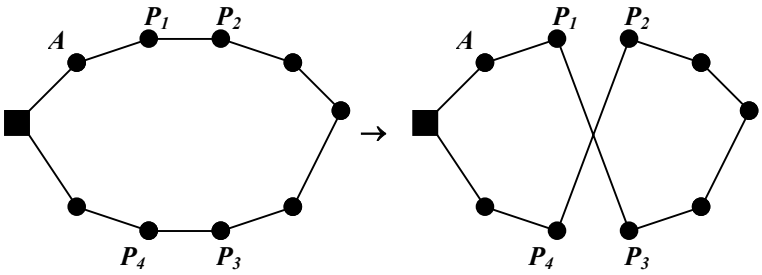
Pierwszy sposób jest mutacją, w której wybierany jest losowo jeden element chromosomu, a następnie zostaje wstawiony na losowo wybraną pozycję. Wymiana genów natomiast polega na wylosowaniu dwóch pozycji chromosomu, a następnie zamianie miejscami zajmujących je genów. Zarówno przemieszczenie, jak i wymiana mogą dotyczyć nie tylko jednego genu, ale także pewnego ich podciągu.

Ostatnim etapem działania prezentowanego programu ewolucyjnego jest zastosowanie procedury lokalnej optymalizacji. Polega ona na uruchomieniu prostej metody heurystycznej, której celem jest poprawa dotychczas uzyskanego rozwiązania, przedstawianego przez dany chromosom populacji. Heurystykę stosuje się w odniesieniu nie do zakodowanej postaci rozwiązania, czyli chromosomu, ale do jego fenotypu. W zdekodowanej postaci osobnika jest pewien zbiór tras, które – rozpatrywane od-

dzielnie – są w miarę możliwości poprawiane. Oznacza to, że za każdym razem rozpatrywane jest zadanie jednego komiwojażera w celu sprawdzenia, czy istnieje szansa znalezienia korzystniejszej kolejności obsługi przydzielonych do danego pojazdu punktów (rys. 8).



Rys. 8. Proces lokalnej optymalizacji
Ź r ó d ł o: Opracowanie własne.



Rys. 9. Przykład przekształcenia aktualnej trasy w nową
Ź r ó d ł o: Opracowanie własne

W tym celu wykorzystano uproszczoną wersję metody Lina–Kernighana [11], stosowanej do rozwiązywania zadań jednego komiwojażera. W aktualnie rozpatrywanej trasie pojazdu sprawdzane są kolejne pary połączeń pomiędzy sąsiednimi punktami. Zakładając, że rozpatrywane są dwie krawędzie oznaczające przejazdy pomiędzy punktami: P_1 i P_2 oraz P_3 i P_4 (rys. 9), dokonano przerwań aktualnej trasy pomiędzy wskazanymi parami punktów, a następnie stworzono nową trasę przez konstrukcję

nowych połączeń: P_1 z P_3 oraz P_2 i P_4. Należy w tym miejscu zauważyć, że krawędzie starej trasy pomiędzy punktami P_2 i P_3 pozostaną bez zmian, zmieni się natomiast kolejność odwiedzania punktów na odwrotną. Jeżeli nowa trasa pojazdu jest korzystniejsza od poprzedniej, to należy zastąpić nią starą.

Przedstawiony proces poszukiwania lepszego rozwiązania dla jednego pojazdu może się odbywać dla wszystkich krawędzi wchodzących w skład trasy z wyjątkiem połączenia pomiędzy bazą a pierwszym punktem obsługi w trasie (punkt A na rys. 9). W przedstawionym sposobie kodowania rozwiązań wielkość popytu (podaży) tego punktu powodowała konieczność rozpoczęcia trasy nowego pojazdu. Zmiana krawędzi na inną podczas lokalnej optymalizacji mogłaby doprowadzić do tego, że pierwszym punktem rozpatrywanej trasy mógłby być taki, który po ponownym zakodowaniu rozwiązania wszedłby w skład trasy pojazdu obsługującego punkty reprezentowane przez kilka wcześniejszych genów. Spowodowałoby to, iż nowy chromosom po dokonaniu lokalnej optymalizacji przedstawiałby inny przydział punktów do pojazdów, a tym samym zupełnie inny harmonogram tras. Założenie to wynika z faktu, że przyporządkowanie określonej struktury genów w chromosomie do rozwiązania rzeczywistego, czyli fenotypu, musi być wzajemnie jednoznaczne.

4. Wyniki obliczeń

Zaprezentowany algorytm ewolucyjny został wykorzystany do rozwiązania kilku zadań testowych, a także w badaniach empirycznych dotyczących zwózki mleka w spółdzielni mleczarskiej [6]. Przykłady te przedstawiają problemy wielu komiwojażerów, odnoszących się do zadań układania tras dla pojazdów z ograniczoną ładownością. W zadaniach tych nie jest wyspecyfikowana konkretna liczba pojazdów, jakimi dysponuje dostawca, lecz jedynie ich minimalna liczba wraz z maksymalną ładownością każdego z nich. Przyjęto także, iż wszystkie pojazdy mogą przewozić maksymalnie taką samą ilość dobra. Pojazdy w każdym z przykładów mają do obsłużenia kilkudziesięciu odbiorców (od 30 do 50), z którymi jest związana pewna wartość oznaczająca popyt. Dla każdego przykładu testowego podane jest również znane rozwiązanie optymalne (R^*) – optymalna łączna długość tras pojazdów.

W tabelach 1 i 2 zaprezentowano wyniki uzyskane przy wykorzystaniu zaproponowanego programu ewolucyjnego PEWKOM. Dla porównania przedstawiono także wyniki otrzymane przy zastosowaniu dwóch metod heurystycznych. Obie heurystyki oparte są na strategiach przeszukiwania przestrzeni rozwiązań przedstawionych w pierwszej części artykułu. Pierwsza z metod – algorytm SAVINGS realizuje strategię konstrukcyjną, podchodzącą w sposób kompleksowy do omawianego problemu. Druga metoda heurystyczna – algorytm dwufazowy (2-F) opiera się na strategii wyznaczającej naj-

pierw rejon dostaw dla dostępnych pojazdów, by następnie ustalać optymalną kolejność ich obsługi w ramach każdego środka transportu indywidualnie.

Tabela 1

Całkowite długości wszystkich tras pojazdów najlepszych rozwiązań dla wybranych zadań testowych

Problem	R*	PEWKOM	SAVINGS	2-F
A-n33-k5	661	661	713	723
A-n36-k5	799	799	828	859
A-n39-k5	822	822	902	994
B-n43-k6	742	742	782	863
E-n51-k5	521	532	589	613

Ź r ó d ł o: Opracowanie własne.

Wartości określone dla poszczególnych metod określają całkowitą długość wszystkich tras użytych pojazdów najlepszych rozwiązań (tabela 1).

Ponieważ w przypadku programu ewolucyjnego za każdym razem generowane może być rozwiązanie o innej wartości funkcji celu, w tabeli 2 podane są więc średnie wartości funkcji celu najlepszych rozwiązań, uzyskanych w kolejnych 20 wywołaniach programu ewolucyjnego.

Tabela 2

Całkowite długości wszystkich tras pojazdów dla wybranych zadań testowych. Dla programu ewolucyjnego PEWKOM – średnia z 20 wywołań

Problem	R*	PEWKOM	SAVINGS	2-F
A-n33-k5	661	685,30	713	723
A-n36-k5	799	822,15	828	859
A-n39-k5	822	839,85	902	994
B-n43-k6	742	756,70	782	863
E-n51-k5	521	579,85	589	613

Ź r ó d ł o: Opracowanie własne.

Dla pierwszych czterech zadań program ewolucyjny znalazł rozwiązania o takiej samej wartości funkcji celu, jak przedstawione przez autorów. Jednocześnie można stwierdzić, że program PEWKOM uzyskuje średnio rozwiązania gorsze od podanych przez autorów zadań rozwiązań tylko o kilka procent.

Należy także zwrócić szczególną uwagę na jakość rozwiązań uzyskanych przez algorytm ewolucyjny w stosunku do wyników osiąganych przez pozostałe dwie heurystyki. Jak widać z przedstawionych zestawień, wypadają one nieco gorzej także od średnio najlepszych rozwiązań programu ewolucyjnego.

Zaprezentowane wyniki uzyskano przy ustaleniu następujących parametrów programu ewolucyjnego: liczebność populacji = 200; liczba wygenerowanych pokoleń = 1000; prawdopodobieństwo zajścia krzyżowania = 0,75; prawdopodobieństwo zajścia mutacji = 0,50.

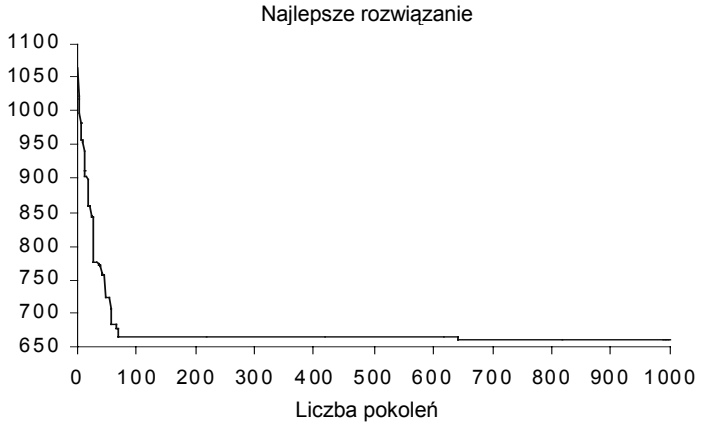
Poniżej przedstawione są wyniki uzyskane w każdym z dwudziestu wywołań algorytmu dla pierwszego zadania testowego. Są to wartości funkcji przystosowania najlepszych chromosomów wygenerowanych przez program ewolucyjny w każdym wywołaniu.

$f_1(\mathrm{Ch}^{\mathrm{best}}) = 683$
$f_2(\mathrm{Ch}^{\mathrm{best}}) = 683$
$f_3(\mathrm{Ch}^{\mathrm{best}}) = 661$
$f_4(\mathrm{Ch}^{\mathrm{best}}) = 676$
$f_5(\mathrm{Ch}^{\mathrm{best}}) = 694$
$f_6(\mathrm{Ch}^{\mathrm{best}}) = 673$
$f_7(\mathrm{Ch}^{\mathrm{best}}) = 693$
$f_8(\mathrm{Ch}^{\mathrm{best}}) = 673$
$f_9(\mathrm{Ch}^{\mathrm{best}}) = 669$
$f_{10}(\mathrm{Ch}^{\mathrm{best}}) = 681$
$f_{11}(\mathrm{Ch}^{\mathrm{best}}) = 704$
$f_{12}(\mathrm{Ch}^{\mathrm{best}}) = 690$
$f_{13}(\mathrm{Ch}^{\mathrm{best}}) = 683$
$f_{14}(\mathrm{Ch}^{\mathrm{best}}) = 675$
$f_{15}(\mathrm{Ch}^{\mathrm{best}}) = 707$
$f_{16}(\mathrm{Ch}^{\mathrm{best}}) = 703$
$f_{17}(\mathrm{Ch}^{\mathrm{best}}) = 699$
$f_{18}(\mathrm{Ch}^{\mathrm{best}}) = 682$
$f_{19}(\mathrm{Ch}^{\mathrm{best}}) = 696$
$f_{20}(\mathrm{Ch}^{\mathrm{best}}) = 681$

Chromosom, który spośród 20 uruchomień algorytmu przedstawia najlepszą kombinację tras dla pojazdów osiągnięty został w trzecim wywołaniu programu ewolucyjnego. Przedstawia on zbiór tras o łącznej długości 661. W każdym kolejnym uruchomieniu programu ewolucyjnego najlepsze uzyskane rozwiązanie przedstawiało inny układ tras, a tym samym o innej łącznej ich długości. Jednak rozwiązania prezentowane przez poszczególne najlepsze chromosomy są do siebie zbliżone pod względem wartości funkcji celu. Odchylenie względne wynosi 1,84%. Na rysunku 10 przedstawiono proces poszukiwania chromosomu, reprezentującego najlepsze rozwiązanie omawianego zadania.

Program ewolucyjny, rozpoczynając działanie od losowo wygenerowanych zbiorów tras dla każdego chromosomu, których wartości funkcji przystosowania wynoszą powyżej 1000, już po wygenerowaniu kilkudziesięciu pokoleń populacji znajduje rozwiązanie bardzo bliskie optymalnemu. Pomimo iż liczba generacji została ustalona

na poziomie 1000, można powiedzieć, że od około 100-ego pokolenia algorytm ewolucyjny osiągnął zbieżność. Oznacza to, że w rozpatrywanej przestrzeni rozwiązań algorytm znalazł obszar, w którym może istnieć najlepszy zbiór tras dla pojazdów. Później rozwiązanie uległo już tylko nieznacznej poprawie.



Rys. 10. Wartości funkcji celu najlepszych rozwiązań w kolejnych pokoleniach
Ź r ó d ł o: Opracowanie własne.

W porównaniu ze znanymi metodami przeznaczonymi do rozwiązywania zagadnień układania tras dla pojazdów, procedury oparte na programowaniu ewolucyjnym wydają się bardzo obiecujące. Wynika to z procesów samodostosowania (adaptacji) reguł przeszukiwania przestrzeni rozwiązań do konkretnego zadania oraz poszukiwania optimum globalnego, startując nie z jednego, lecz kilkudziesięciu punktów przestrzeni rozwiązań. Ze względu na swoje niezaprzeczalne atuty, takie jak prostota działania, uniwersalność oraz elastyczność w budowie programów ewolucyjnych, są one niezwykle interesującym narzędziem, które może być wykorzystane w nowych obszarach zastosowań. Jest w pełni uzasadnione prowadzenie dalszych badań nad zastosowaniem algorytmów ewolucyjnych do zadań układania tras dla pojazdów, zwłaszcza polegających na:

- wykorzystaniu programowania ewolucyjnego do rozwiązywania różnych wariantów zadań VRP, uwzględniających wiele dodatkowych założeń występujących w praktyce gospodarczej;
- konstrukcji nowych metod kodowania parametrów zadań w postaci chromosomów oraz ich funkcji przystosowania, uwzględniających dodatkowe założenia;
- budowie nowych, przynoszących lepsze efekty operatorów genetycznych oraz metod selekcji, w tym takich, które uwzględniają dodatkową wiedzę o optymalizowanym problemie.

Bibliografia

[1] CAŁCZYŃSKI A., *Modele i metody ustalania tras przewozów towarowych*, Instytut Handlu Wewnętrznego i Rynku, Warszawa 1979.
[2] CLARKE G., WRIGHT J.W., *Scheduling of vehicles from a central depot to a number of delivery points*, Operations Research, 1964, Vol. 12, s. 568–581.
[3] FISHER M., JAIKUMAR R., *A generalized assignment heuristic for the vehicle routing*, Networks, 1981, Vol. 11, s. 109–124.
[4] GILLET B.E., MILLER L.R., *A heuristic algorithm for the vehicle dispatch problem*, Operations Research, 1974, Vol. 22, s. 340–349.
[5] GOLDERBERG D.E., *Algorytmy genetyczne i ich zastosowania*, WNT, 2003.
[6] JADCZAK R., *Wykorzystanie metod programowania ewolucyjnego do rozwiązania problemu wielu komiwojażerów*, praca doktorska napisana w Katedrze Badań Operacyjnych, w Instytucie Ekonometrii i Statystyki Uniwersytetu Łódzkiego, Łódź 2005.
[7] JASIŃSKI L.J., *Optymalizacja dostawy towarów na zaopatrzenie rynku w warunkach niepewności*, Instytut Rynku Wewnętrznego i Konsumpcji, Warszawa 1987.
[8] LAPORTE G., *The vehicle routing problem: An overview of exact and approximate algorithms*, European Journal of Operational Research, 1992, Vol. 59, Issue 3, s. 345–358.
[9] LAPORTE G., SEMET F., *Classical heuristic for the vehicle routing problem*, [w:] Toth P., Vigo D. *The Vehicle Routing Problem*, Monograph on Discrete Mathematics and Applications, SIAM 2000.
[10] LENSTRA J.K., RINOOY KAN A.H.G., *Complexity of vehicle routing and scheduling problems*, Networks, 1981, Vol. 11, s. 221–227.
[11] LIN S., KERNIGHAN B.W., *An effective heuristic for the traveling salesman problem*, Operations Research, 1973, Vol. 21, s. 498–516.
[12] LITTLE J.D.C., MURTY K.G., SWEENEY D.W., KAREL C., *An algorithm for the traveling salesman problem*, Operations Research, 1963, Vol. 11.
[13] MICHALEWICZ Z., *Algorytmy genetyczne + struktury danych = programy ewolucyjne*, WNT, 1999.
[14] MOLE R.H., JAMESON S.R., *A sequential route-building algorithm employing a generalized savings criterion*, Operations Research Quarterly, 1976, Vol. 27, s. 503–511.
[15] PAESSENS H., *The savings for the vehicle routing problem*, European Journal of Operational Research, 1988, Vol. 34, s. 336–344.

Solving Vehicle Routing Problems with evolutionary algorithms

One of the most known combinatorial problem is a Travelling Salesman Problem. In this paper a extended version is considered – multisalesman problem, which is known in the literature as a vehicle routing problem. It is very easy to define this *NP*-hard problem, but to find a optimal solution is very hard.

In this work application of evolutionary algorithm is presented. These methods are constructed for optimization problems, but in particular they are very useful for combinatorial problems.

In this paper vehicle routing problem is defined and the most known heuristic methods are presented. In the next section, evolutionary program for solving vehicle routing problem is formulated. At the end, the experimental results for some test problems are shown and analyzed.

Keywords: *vehicle routing problem, evolutionary algorithms*