

Roland Jachimowski¹

Wydział Transportu, Politechnika Warszawska

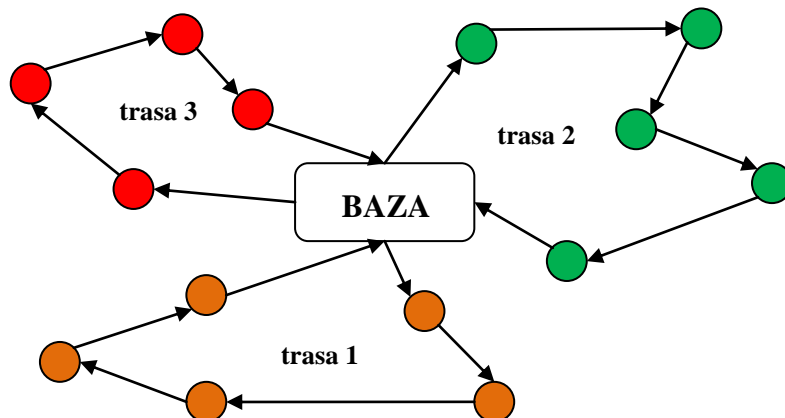
Zastosowanie algorytmów heurystycznych do rozwiązywania problemu układania tras pojazdów

1. WPROWADZENIE

Szybki rozwój wymiany handlowej, stale rosnące wymagania odnośnie poziomu obsługi klienta sprawiają, iż logistyka jest obecnie uważana za jedno z najważniejszych narzędzi racjonalizacji działań, obniżania kosztów i podwyższania konkurencyjności [4]. Ograniczone zasoby środków transportu, silna konkurencja na rynku przewozów wymaga przemyślanego planowania wykorzystania dostępnego taboru. Planowanie to jest szczególnie trudne w przypadku dystrybucji towarów, kiedy pojedynczy pojazd dostarcza ładunki w jednym cyklu do wielu odbiorców. W tej sytuacji planowanie to polega na określeniu kolejności obsługi odbiorców i w sposób znaczący może przyczynić się do skrócenia długości trasy pokonywanej przez pojazd. Obok długości trasy pojazdu kryterium planowania może być chociażby czas realizacji trasy. Tak więc, tego typu planowanie transportu staje się problemem natury kombinatorycznej.

W badaniach operacyjnych problem planowania trasy pojazdu dokonującego zwózki bądź rozwózki ładunków nosi miano problemu układania tras pojazdów (ang. Vehicle Routing Problem) i wywodzi się od powszechnie znanego problemu komiwojażera [2]. W ogólnym ujęciu problem układania tras pojazdów polega na obsłużeniu przez pojazd każdego dostawcy/odbiorcy tylko jeden raz i powrocie do bazy skąd rozpoczęta została trasa. Kryterium optymalności jest w tym miejscu długość trasy, przy czym znane są odległości pomiędzy wszystkimi odbiorcami/dostawcami oraz odległości pomiędzy tymi dostawcami/odbiorcami a bazą.

Schematycznie strukturę problemu układania tras pojazdów przedstawiono na rysunku 1.



Rys. 1. Struktura problemu układania tras pojazdów

Źródło: opracowanie własne

W literaturze istnieje wiele różnych odmian problemu układania tras pojazdów, w zależności od warunków, na jakich pojazdy mogą realizować swoje trasy. Warunki te dotyczą chociażby określonej ładowności pojazdów, liczby baz, z jakich pojazdy obsługują odbiorców, okien czasowych odbiorców, możliwości jednoczesnej zwózki i rozwózki, konieczności zakończenia trasy w bazie, czy też wielkości zapotrzebowania odbiorców wyrażonego rozkładem prawdopodobieństwa [12].

¹rjach@wt.pw.edu.pl

Niniejszy artykuł jest podsumowaniem badań nad algorytmami heurystycznymi stosowanymi do rozwiązywania problemu układania tras pojazdów.

2. ALGORYTMY DLA PROBLEMU UKŁADANIA TRAS PRZEWOZU

2.1. Heurystyki

Z uwagi na fakt, iż liczba rozwiązań dopuszczalnych problemu układania tras pojazdów rośnie wykładniczo względem liczby danych, rozwiązywanie nawet nie dużych przypadków wymaga stosowania algorytmów przybliżonych. Wśród algorytmów opisanych w literaturze przedmiotu stosowanych do rozwiązywania problemu VRP wyróżnić można m.in.:

- heurystyki konstrukcyjne,
- heurystyki lokalnej optymalizacji,
- metaheurystyki (algorytmy genetyczne, roju cząstek, symulowane wyżarzanie, sieci neuronowe).

Idea algorytmu konstrukcji tras polega na łączeniu ze sobą kolejnych węzłów bądź łuków aż do uzyskania rozwiązania dopuszczalnego, czyli rozwiązania, które może być bardzo dalekie od optymalnego. Kolejne węzły wybierane są na podstawie określonego kryterium (zazwyczaj koszt dołączenia kolejnego punktu do trasy pojazdu) [3].

Z kolei heurystyki lokalnej optymalizacji polegają na zastępowaniu aktualnie rozpatrywanego rozwiązania nowym, o lepszym z punktu widzenia kryterium optymalizacji układem tras. W najprostszym przypadku pierwsze rozwiązanie dopuszczalne jest generowane chociażby metodą zachłanną. Generowanie kolejnych rozwiązań dopuszczalnych polega na wymianie jakiegoś atrybutu albo kombinacji atrybutów w rozwiązaniu poprzednim. Kryterium wyboru rozwiązania sprawdza jakość aktualnego rozwiązania i jeżeli jest ono lepsze od rozwiązania poprzedniego, zastępuje ono rozwiązanie poprzednie i staje się rozwiązaniem bazowym dla kolejnej zmiany atrybutów [11].

2.2. Metaheurystyki

W niniejszym artykule porównano pod kątem jakości oraz wydajności trzy wybrane algorytmy metaheurystyczne dla problemu układania tras pojazdów (algorytm genetyczny, algorytm roju cząstek oraz algorytm symulowanego wyżarzania). Stąd też w dalszej części tego rozdziału przedstawiono krótką ich charakterystykę (szczegółowy zapis omawianych poniżej algorytmów można znaleźć w pracach [6][5][7]).

Jednym z najpopularniejszych algorytmów metaheurystycznych wykorzystywanych w problemie układania tras pojazdów jest algorytm genetyczny. Działanie algorytmu polega w pierwszej kolejności na wylosowaniu pewnej populacji początkowej. Następnie populacja poddawana jest selekcji a najlepsze osobniki są w dalszej kolejności krzyżowane ze sobą. Po etapie krzyżowania przeprowadzana jest mutacja polegająca na wprowadzaniu niewielkich zmian losowych do nowopowstałych osobników. W ten sposób powstaje kolejne (nowe) pokolenie osobników i algorytm powraca do etapu poddawania tych osobników selekcji. Jeżeli reprodukcja i mutacja kolejnego pokolenia nie poprawia wyniku rozwiązania, to uzyskane wcześniej rozwiązanie jest rozwiązaniem najlepszym. W taki sposób algorytm genetyczny może w zasadzie działać w nieskończoność. Dlatego też wprowadza się ograniczenia dotyczące maksymalnej liczby iteracji algorytmu. Kryterium zakończenia działania algorytmu w przypadku wyznaczania tras pojazdów oprócz czasu trwania obliczeń mogą być również uzyskane długości tras, czy chociażby brak poprawy wyniku przez pewną ilość iteracji [10].

Do budowy porównywanego w niniejszym artykule algorytmu genetycznego posłużono się ścieżkową reprezentacją osobników, krzyżowane zrealizowano metodą OX, a mutacji dokonano z wykorzystaniem algorytmu 2-opt. Szczegółową charakterystykę algorytmu zamieszczono w pracy [7].

Kolejnym algorytmem, porównywanym w niniejszym artykule jest algorytm symulowanego wyżarzania. Algorytm symulowanego wyżarzania (*ang. Simulated Annealing*) swoją ideę czerpie z procesu schładzania stopów metali. Otóż cząsteczki substancji znajdującej się w stanie ciekłym

poruszają się szybko i chaotycznie. Wraz ze stopniowym spadkiem temperatury maleje energia cząsteczek, cząsteczki zaczynają poruszać się coraz wolniej i jednocześnie dążą do uzyskania położenia, w którym osiągną minimum potencjału. Jeżeli schładzanie cieczy jest procesem powolnym, to cząsteczki osiągną położenie, w którym będą miały minimum potencjału i utworzą stałą strukturę krystaliczną. W przeciwnym przypadku, gdy schładzanie przeprowadzone zostanie szybko, cząsteczki mogą nie zdążyć osiągnąć owego minimum potencjału i w stanie stałym będą poustawiane chaotycznie. Dążenie cząsteczek do osiągnięcia minimum potencjału ma swoje podobieństwo w poszukiwaniu minimalnej trasy w problemie komiwojażera. Analogię tę widać, jeśli aktualne położenie danej cząsteczki odniesiemy do jednego z możliwych rozwiązań dopuszczalnych problemu komiwojażera. W przypadku problemu VRP algorytm rozpoczyna swoje działanie od wylosowania dowolnego rozwiązania dopuszczalnego (permutacji miast) i obliczenia wartości funkcji kryterium dla tego rozwiązania. W kolejnym kroku algorytmu badane jest sąsiedztwo danego rozwiązania dopuszczalnego. Następnie wybierane jest jedno z wyznaczonych sąsiedztw i obliczana jest dla niego wartość funkcji kryterium. Jeżeli wartość funkcji kryterium dla tego sąsiedztwa jest większa bądź równa wartości permutacji początkowej, to przyjmowana jest jako nowe rozwiązanie. Jeśli wartość ta jest mniejsza od wartości permutacji poprzedniej to jako nowe rozwiązanie jest ona przyjmowana z pewnym prawdopodobieństwem, które bezpośrednio zależy od temperatury regulującej proces akceptacji gorszych rozwiązań [9]. Szczegółową charakterystykę algorytmu oraz kolejne kroki jego działania zamieszczono w pracy [5].

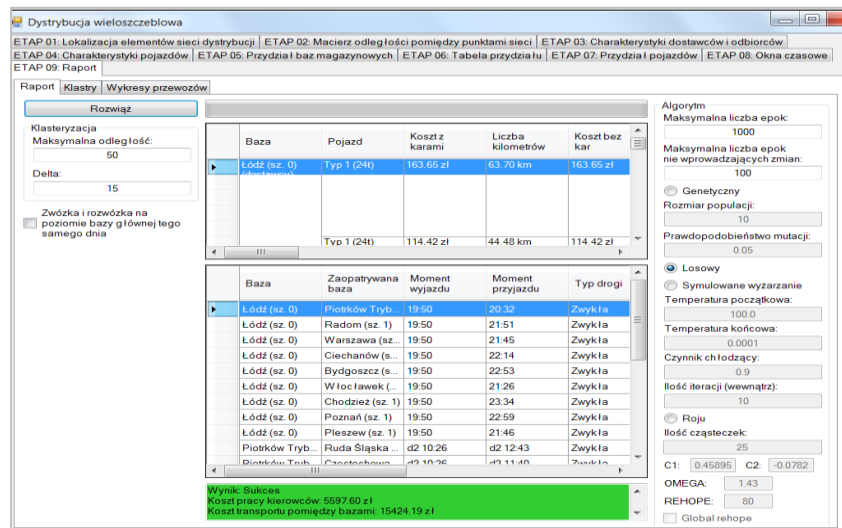
Ostatnim algorytmem porównywanym w niniejszym artykule jest algorytm roju cząstek (*ang. Particle Swarm Optimization PSO* [8]). Algorytm bazuje na komunikacji pomiędzy członkami stada ptaków (członkami roju), ich przekazywaniu sobie informacji oraz zachowaniu całego stada. Każdy osobnik należący do danego stada traktowany jest jako pojedyncza cząstka. Podobnie jak w przypadku algorytmu symulowanego wyżarzania tak i tu cząstki poruszają się w kierunku nowych położzeń poszukując optimum (minimum w przypadku problemu VRP). W swym przemieszczaniu cząstki śledzą przywódcę stada, którym jest aktualnie najlepsze rozwiązanie problemu, jednocześnie przyspieszając i zmieniając kierunek za każdym razem, kiedy takie rozwiązanie zostanie znalezione. O zmianie kierunku i swojej prędkości osobniki podejmują decyzję na podstawie swojego sąsiada, przy czym dzięki komunikacji w stadzie, zachowuje ono swoją dynamikę. W każdej iteracji algorytmu cząstki podążają za najlepszym rozwiązaniem (przywódcą stada). Szczegółową charakterystykę algorytmu oraz kolejne kroki jego działania zamieszczono w pracy [6].

3. PRZYKŁAD OBLICZENIOWY

Badania opisanych w poprzednim rozdziale algorytmów dokonano z wykorzystaniem autorskiej aplikacji komputerowej „Dystrybucja Wieloszczeblowa” [7]. Program umożliwia układanie tras pojazdów w systemach o dowolnej liczbie szczebli dystrybucji. Oprócz algorytmu genetycznego, symulowanego wyżarzania i roju cząstek, w aplikacji zaimplementowano dodatkowo algorytm losowy. Okno modułu obliczeniowego aplikacji przedstawiono na rysunku 2.

Testy jakościowe oraz wydajnościowe algorytmu przedstawiono na przykładzie obliczeniowym składającym się z 98 punktów nadania ładunków, 559 punktów odbioru ładunku oraz 21 terminali przeładunkowych zlokalizowanych na trzech szczeblach dystrybucji.

Realizację obliczeń aplikacja rozpoczyna od przydziału wszystkich punktów nadania i odbioru do klastrów. Realizowane jest to za pomocą algorytmu klasteryzacji, którego kolejne kroki zamieszczonego w pracy [1]. Następnie wyznaczane są trasy pojazdów obsługujących dostawców i odbiorców na poziomie każdego klastra. Z uwagi na dość znaczny zakres tych danych, nie zostały one w niniejszym artykule przedstawione. Nie mniej jednak szczegółowo dane do tego przykładu przedstawiono w pracy [7].



Rys. 2. Moduł obliczeniowy aplikacji

Źródło: opracowanie własne

Układanie tras pojazdów za pomocą poszczególnych algorytmów heurystycznych i późniejsze przeprowadzanie testów jakościowych i wydajnościowych wymagało określenia parametrów tych algorytmów. Badania algorytmów dokonano zmieniając wartości ich parametrów odnoszące się do maksymalnej liczby iteracji algorytmu i wielkości początkowej populacji rozwiązań dopuszczalnych. Pozostałe parametry algorytmów nie były zmieniane (parametry niezmienniane scharakteryzowano poniżej wartością liczbową). W związku z tym, parametry niezbędne dla przeprowadzenia badań to:

Algorytm losowy

- liczba iteracji

Algorytm genetyczny

- rozmiar populacji początkowej,
- prawdopodobieństwo mutacji – 0,05,
- maksymalna liczba epok algorytmu (maksymalna liczba iteracji),
- maksymalna liczba epok nie wprowadzających zmian, po których następuje zatrzymanie algorytmu – 100

Algorytm symulowanego wyżarzania

- temperatura początkowa,
- temperatura końcowa,
- czynnik chłodzący – 0,9.

Algorytm roju cząstek

Parametry dobierano doświadczeniowo wykonując 100 losowań parametrów C1, C2 i OMEGA z zakresu (-1,9;1,9) oraz parametru REHOPE z zakresu (1;500).

- ilość cząstek,
- C1 – 0,45895 (określa jak bardzo cząsteczka polega na swoich wcześniejszych rozwiązaniach),
- C2 – -0,0782 (określa jak bardzo cząsteczka polega na rozwiązaniach swoich sąsiadów),
- OMEGA – 1,43 (określa jak bardzo cząsteczka polega na swoim rozwiązaniu z poprzedniej iteracji),
- REHOPE – 100 (określa ilość iteracji niepolepszających najlepszego rozwiązania znalezionego przez daną cząstkę. Po przekroczeniu wartości tego parametru prędkość i pozycja cząstki zostaje ponownie zainicjalizowana losową pozycją, a jej prędkość wyzerowana).

W wyniku zastosowania powyższych algorytmów uzyskano bliskie optymalnym plany przewozów od dostawców do odbiorców z wykorzystaniem terminali przeładunkowych.

4. TESTOWANIE ALGORYTMÓW PLANOWANIA PRZEWOZÓW

4.1. Testy jakościowe algorytmów

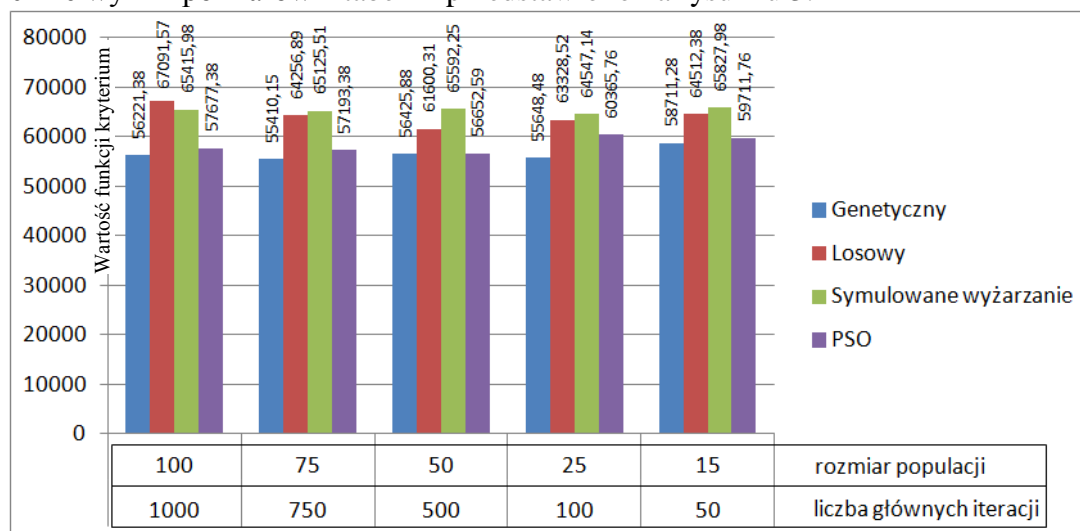
Testy jakościowe algorytmów polegały na określeniu wartości funkcji kryterium optymalizowanego problemu dla różnych parametrów algorytmów, przy czym parametry te w tego typu testach określają wielkość przestrzeni potencjalnych rozwiązań problemu. Testy jakościowe przeprowadzono dla tych samych parametrów poszczególnych algorytmów odnoszących się do ilości głównych iteracji algorytmu oraz w przypadku algorytmu genetycznego i roju cząstek dodatkowo dla parametrów odnoszących się do liczby dopuszczalnych rozwiązań bazowych (wielkości populacji). Wyniki pomiarów dla przykładu obliczeniowego przedstawiono w tabeli 1.

Tabela 1. Wyniki pomiarów wartości funkcji kryterium w funkcji liczby głównych iteracji algorytmu i liczby dopuszczalnych rozwiązań bazowych

Liczba iteracji	1000	750	500	100	50	Wartość funkcji kryterium [zł]
Wielkość populacji	100	75	50	25	15	
Genetyczny	56221,38	55410,15	56425,88	55648,48	58711,28	
Losowy	67091,57	64256,89	61600,31	63328,52	64512,38	
Symulowane wyżarzanie	65415,98	65125,51	65592,25	64547,14	65827,98	
PSO	57677,38	57193,38	56652,59	60365,76	59711,76	

Źródło: opracowanie własne

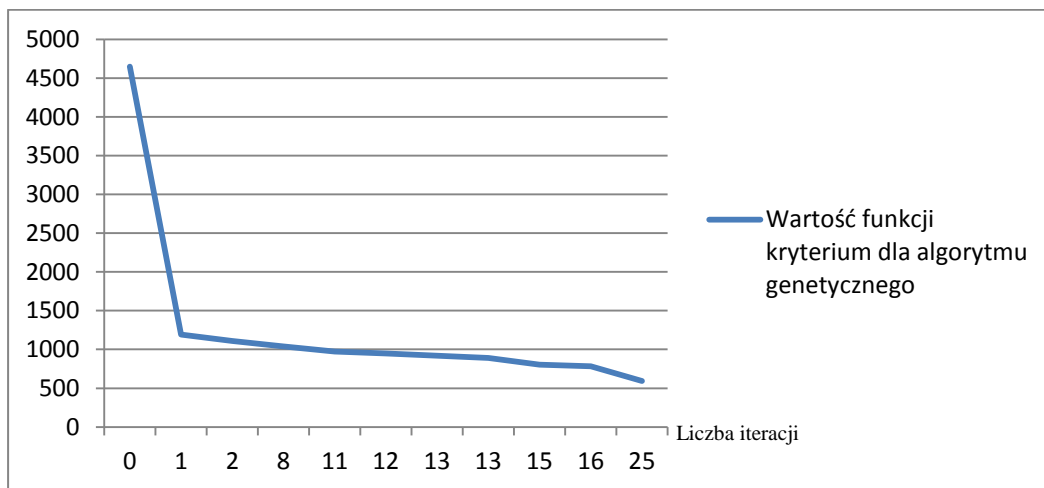
Graficznie wyniki pomiarów z tabeli 1 przedstawiono na rysunku 3.



Rys. 3. Wyniki pomiarów wartości funkcji kryterium w funkcji liczby głównych iteracji algorytmu i liczby dopuszczalnych rozwiązań bazowych

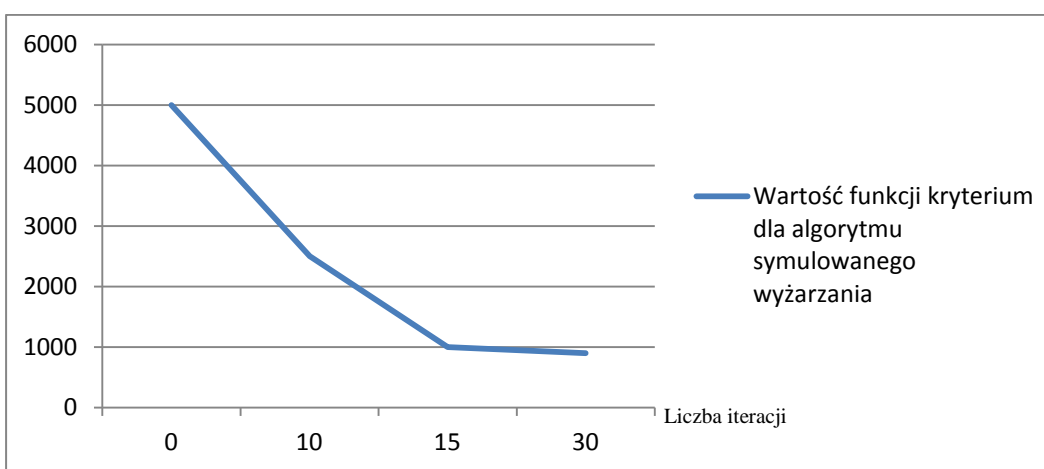
Źródło: opracowanie własne

Ponadto w celu przedstawienia zachowania się algorytmu w kolejnych jego iteracjach, dla klastra nr 1 obsługiwanego przez pierwszy pojazd centralnego terminala przeładunkowego na rysunkach 4-6 przedstawiono wartości funkcji kryterium dla każdego z algorytmów metaheurystycznych w funkcji kolejnych ich iteracji.



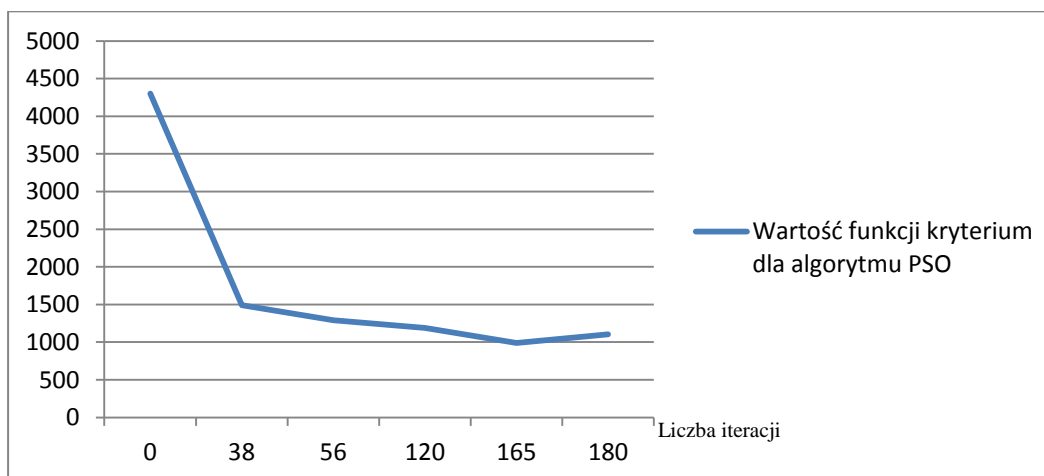
Rys.4. Zmiany wartości funkcji kryterium w funkcji kolejnych iteracji algorytmu genetycznego dla klastra nr 1

Źródło: opracowanie własne



Rys. 5. Zmiany wartości funkcji kryterium w funkcji kolejnych iteracji algorytmu symulowanego wyżarzania dla klastra nr 1

Źródło: opracowanie własne



Rys. 6. Zmiany wartości funkcji kryterium w funkcji kolejnych iteracji algorytmu PSO dla klastra nr 1

Źródło: opracowanie własne

Dla algorytmu losowego z uwagi na każdorazowe losowanie trasy pojazdu, wykresu zmiany wartości w funkcji kolejnych iteracji nie przedstawiono.

4.2. Analiza testów jakościowych

Analiza rysunku 3 pod kątem uzyskiwanych przez algorytmy końcowej wartości funkcji kryterium wskazuje, iż zdecydowanie najlepsze wyniki zwraca algorytm genetyczny. Nieco gorsze jakościowo wyniki uzyskano przy zastosowaniu algorytmu roju cząstek PSO. Wartość średnia wyników dla algorytmu genetycznego uzyskanych dla różnych ilości iteracji i rozmiaru populacji jest o 3,14% niższa od wartości średniej dla algorytmu PSO, 13,5% niższa od wartości średniej dla algorytmu symulowanego wyżarzania oraz o 11,9% niższe od wartości średniej dla algorytmu losowego.

Z powyższego zestawienia wynika, iż najslabsze jakościowo wyniki zwracał algorytm symulowanego wyżarzania. Wynika to głównie z trudności w ustalaniu wartości czynnika chłodzącego i tym samym prawdopodobieństwa, z jakim gorsze rozwiązanie jest na danym etapie algorytmu przyjmowane za bazowe. Źródła literaturowe podają [9], iż właściwe ustalenie temperatury oraz czynnika chłodzącego umożliwia uzyskiwanie wyników bliskich wynikom chociażby dla algorytmu genetycznego.

Porównując zmiany zachowania się algorytmów w funkcji kolejnych ich iteracji zauważono, iż ponownie najlepszy okazał się być algorytm genetyczny. W analizowanym klastrze już po pierwszej iteracji wartość funkcji kryterium dla algorytmu genetycznego osiągnęła poziom podobny do końcowych wartości dla pozostałych algorytmów. W algorytmie symulowanego wyżarzania, bliską końcowej wartości funkcji kryterium algorytm osiągnął już w 15 jego iteracji. Zdecydowanie dłużej wartości końcowej poszukiwał algorytm roju. Najlepszą uzyskaną wartość funkcji kryterium algorytm osiągnął w 165 iteracji.

4.3. Testy wydajnościowe algorytmów

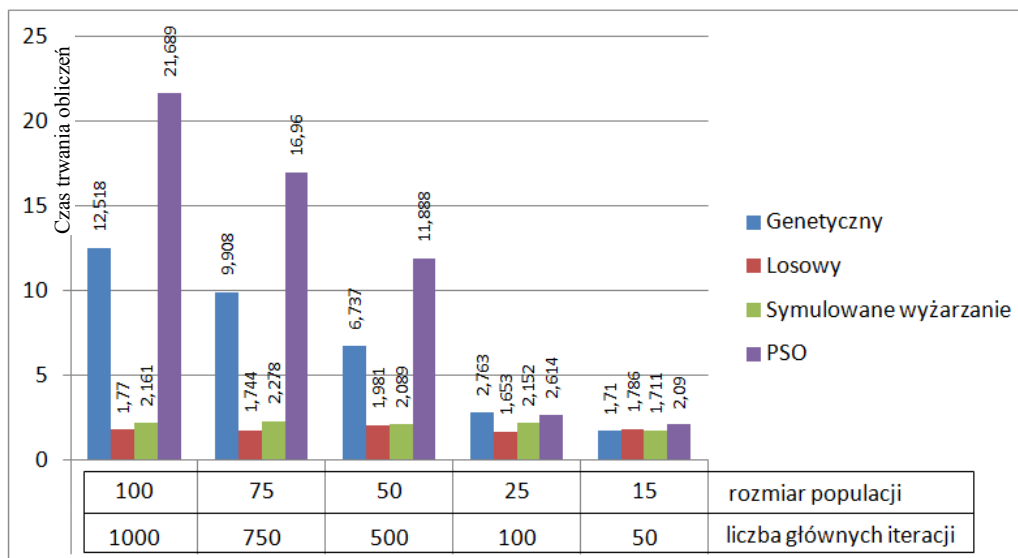
Testy wydajnościowe algorytmów polegały na określeniu czasu trwania obliczeń dla różnych parametrów algorytmów. Testy wydajnościowe podobnie jak jakościowe przeprowadzono dla tych samych parametrów poszczególnych algorytmów odnoszących się do ilości głównych iteracji algorytmu oraz wielkości populacji. Wyniki pomiarów dla przykładu obliczeniowego zamieszczonego w tabeli 2.

Tabela 2. Wyniki pomiarów czasu trwania obliczeń w funkcji liczby głównych iteracji algorytmu i liczby dopuszczalnych rozwiązań bazowych

Liczba iteracji	1000	750	500	100	50	Czas obliczeń [s]
Wielkość populacji	100	75	50	25	15	
Genetyczny	12,518	9,908	6,737	2,763	1,710	
Losowy	1,770	1,744	1,981	1,653	1,786	
Symulowane wyżarzanie	2,161	2,278	2,089	2,152	1,711	
PSO	21,689	16,960	11,888	2,614	2,090	

Źródło: opracowanie własne

Graficznie wyniki pomiarów z tabeli 2 przedstawiono na rysunku 7.



Rys. 7. Wyniki pomiarów czasu trwania obliczeń w funkcji liczby głównych iteracji algorytmu i liczby dopuszczalnych rozwiązań bazowych

Źródło: opracowanie własne

4.4. Analiza testów wydajnościowych

Analiza testów wydajnościowych wskazuje na to, iż lepsze w porównaniu z innymi algorytmami wyniki uzyskane przez algorytm genetyczny i PSO okupione zostały dłuższym czasem. Wynika to głównie z rozmiarów populacji początkowych rozwiązań bazowych tych algorytmów. Należy jednak zwrócić jednocześnie uwagę na fakt, iż mimo zdecydowanie dłuższego czasu poszukiwania rozwiązania dla algorytmu genetycznego i PSO, czas ten w dalszym ciągu dla dość dużego przykładu obliczeniowego wynosił zaledwie kilkanaście sekund. Tak więc, zdecydowanie ważniejsza jest w tym miejscu jakość uzyskiwanego rozwiązania niż czas poszukiwania tego rozwiązania.

5. WNIOSKI

Opracowana aplikacja komputerowa "Dystrybucja wieloszczelbowa" umożliwi szybkie zaplanowanie procesu przewozowego w systemach o dowolnej liczbie szczebli oraz dowolnej liczbie pośredników na każdym ze szczebli. Dzięki zaimplementowanym w aplikacji algorytmom możliwe jest obserwowanie zmian zachodzących w wynikach uzyskiwanych dla różnych przykładów obliczeniowych przy wykorzystaniu tych właśnie algorytmów. Tym samym aplikacja rozwijana o kolejne algorytmy umożliwić będzie analizę oraz testowanie działania tych algorytmów, co pozwoli na poszukiwanie algorytmów coraz lepszych jakościowo i wydajnościowo.

Streszczenie

Artykuł jest podsumowaniem badań nad algorytmami heurystycznymi stosowanymi do rozwiązywania problemu układania tras pojazdów. W teoretycznej części artykułu scharakteryzowano problem układania tras oraz algorytmy przybliżone wykorzystywane do jego rozwiązywania. W części praktycznej wybrane algorytmy zaimplementowano w autorskiej aplikacji komputerowej. Na przykładzie obliczeniowym testowano szybkość wykonywania obliczeń przez te algorytmy oraz jakość zwracanych przez nie rozwiązań.

Słowa kluczowe: problem układania tras pojazdów, algorytmy, wydajność algorytmów, jakość algorytmów

The use of the heuristic algorithms for the Vehicle routing problem

Abstract

The article is a summary of research on heuristic algorithms used to solve the Vehicle routing problem. In the theoretical part of the article the Vehicle routing problem and heuristic algorithms used to solve it were characterized. In the practical part of the article, the selected heuristic algorithms were implemented in the author's computer application. For the given example, the computing speed of these algorithms and the quality of the solutions they return were tested.

Keywords: Vehicle routing problem, algorithms, algorithms' efficiency, algorithms' quality

Praca naukowa finansowana w ramach grantu dziekańskiego nr 504013291160 pt. „Wspomaganie decyzji w stochastycznym problemie planowania transportu. Część II” oraz w ramach Pracy Statutowej Zakładu Logistyki i Systemów Transportowych Wydziału Transportu Politechniki Warszawskiej.

LITERATURA

- [1] Ambroziak T., Jachimowski R.: Algorytm klasteryzacji w zastosowaniu do problem trasowania pojazdów, Logistyka 2/2012.
- [2] Całczyński A.: Metody optymalizacyjne w obsłudze transportowej rynku, PWE, Warszawa 1992.
- [3] Clarke G., Wright J.: Scheduling of vehicles from a central depot to a number of delivery points. Oper. Res. 12, 568-581. (1964).
- [4] Coyle J., Bardi C., Langley J.: Zarządzanie logistyczne, Polskie Wydawnictwo Ekonomiczne, Warszawa 2007.
- [5] Jachimowski R. Kłodawski M.: Simulated annealing algorithm for the multi-level vehicle routing problem, Logistyka 4/3013.
- [6] Jachimowski R.: Algorytm metaheurystyczny dla problemu układania tras pojazdów, Logistyka 6/2014
- [7] Jachimowski R.: Metoda wyznaczania optymalnych planów dostaw w hierarchicznym systemie dystrybucji: praca doktorska, OWPW, Warszawa 2012.
- [8] Kennedy, J. & Eberhart, R.C. (1995). Particle swarm optimization, Proceedings of the IEEE International Conference on Neural Networks, Vol. 4, pp. 1942-1948, ISBN: 0780327683, Perth, Western Australia November 1995.
- [9] Kirkpatrick, S., Gellat, C.D., and Vecchi, M.P., Optimization by simulated annealing, Science 220, (1983), 671-680.
- [10] Michalewicz Z.: Algorytmy genetyczne+struktury danych=programy ewolucyjne, Wydawnictwa Naukowo-Techniczne, Warszawa 1999.
- [11] Sysło M., Narsingh D., Kowalik J.: Algorytmy optymalizacji dyskretnej z programami w języku Pascal, PWN, Warszawa 1993.
- [12] Toth P., Vigo D.: The Vehicle Routing Problem, SIAM 2002.