

Rola algorytmów genetycznych w procesie optymalizacji wyznaczania tras przejazdu

Konrad Gauda

Problematyka artykułu dotyczy możliwości wykorzystania algorytmów genetycznych w procesie optymalizacji wyznaczania tras przejazdu. Opisana jest idea algorytmów genetycznych oraz zasada ich działania. Ukazany jest także przebieg wyznaczania optymalnej trasy z wykorzystaniem aplikacji Visual Genetic.

Słowa kluczowe: algorytmy genetyczne, optymalizacja, transport

Wstęp

Algorytm genetyczny (AG) jest narzędziem przeszukującym przestrzeń alternatywnych rozwiązań problemu w celu wyszukania rozwiązań najlepszych. Znajdują one zastosowanie tam, gdzie nie jest dobrze określony lub poznany sposób rozwiązania problemu, ale znany jest sposób oceny jakości rozwiązania [1, 3]. W transporcie czy logistyce znanym przykładem jest problem komiwojażera, gdzie należy znaleźć najkrótszą drogę łączącą wszystkie miasta, tak by przez każde miasto przejść tylko raz. W terminologii grafów miasta są wierzchołkami grafu, a trasy pomiędzy nimi to krawędzie z wagami. Waga krawędzi najczęściej odpowiada odległości pomiędzy miastami połączonymi tą krawędzią. Trasa komiwojażera jest cyklem przechodzącym przez każdy wierzchołek grafu dokładnie jeden raz – jest to tzw. cykl Hamiltona (rys.1.).



Rys.1. Przykładowa trasa komiwojażera – cykl Hamiltona

Rozwiązanie problemu komiwojażera polega na jak najszybszym odwiedzeniu wszystkich lokalizacji (miast) i powrocie do punktu startu. Każdą lokalizację można odwiedzić tylko raz. Inaczej mówiąc należy znaleźć minimalny cykl Hamiltona w pełnym grafie ważonym (tab.1.).

Liczba lokalizacji	Liczba rozwiązań	Graf
2	1 (A-B-A)	
3	2 (A-B-C-A; A-C-B-A)	
4	6 (A-B-C-D-A; A-C-D-B-A; A-C-B-D-A; A-D-C-B-A; A-B-D-C-A; A-D-B-C-A)	
10	362880 (n-1)! Wykładnicza klasa złożoności obliczeniowej	

Tab.1. Liczba rozwiązań w zależności od liczby destynacji

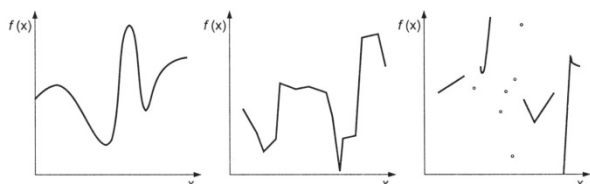
Ocena jakości proponowanej trasy jest błyskawiczna, natomiast znalezienie optymalnego rozwiązania kwalifikuje się do problemów trudnych (wykładnicza klasa złożoności obliczeniowej) [1, 3]. Z problemem tym bardzo dobrze radzą sobie algorytmy genetyczne, które wykorzystują probabilistyczne podejście i dzięki temu szybkość ich działania jest niewspółmiernie większa w stosunku do algorytmów klasycznych. Do podstawowych cech algorytmów genetycznych można zaliczyć następujące:

- AG odwzorowuje naturalne procesy ewolucji zachodzące w czasie (ocenia tysiące generacji w kilka sekund);
- celem tych procesów jest maksymalne dopasowanie osobników do istniejących warunków życia;
- rolę środowiska spełnia tu funkcja oceniająca (tzw. funkcja celu);
- AG łączy w sobie zasadę przeżycia najlepiej przystosowanych osobników z systematyczną, choć zrandomizowaną (losową) wymianą informacji;
- pomimo elementu losowości AG nie błądzi przypadkowo lecz wykorzystuje efektywnie przeszłe doświadczenia (dziedziczenie).

1. Zasada działania algorytmu genetycznego

Zasada działania algorytmów genetycznych oparta jest na mechanizmach doboru naturalnego i dziedziczności.

Mechanizmy te polegają na przetrwaniu osobników najlepiej dostosowanych w danym środowisku, podczas gdy osobniki gorzej przystosowane są eliminowane. Z kolei te osobniki, które przetrwają przekazują informację genetyczną swoim potomkom. Krzyżowanie informacji genetycznej otrzymanej od „rodziców” prowadzi do sytuacji, w której kolejne pokolenia są przeciętnie coraz lepiej przystosowane do warunków środowiska, mamy więc do czynienia ze swoistym procesem optymalizacji [2, 4, 5]. Sposób działania algorytmów genetycznych nieprzypadkowo przypomina zjawisko ewolucji biologicznej, ponieważ ich twórca John Henry Holland właśnie z biologii czerpał inspiracje do swoich prac. Zainteresowały go cechy naturalnej ewolucji, a przede wszystkim fakt, że ewolucja zachodzi na chromosomach, a nie na żywych istotach. Zastosował on ciąg bitów (zer i jedynek) zamiast chromosomu i dokonywał symulacyjnej ewolucji na populacjach chromosomów. Znając funkcję przystosowania (kryteria oceny osobników) można za pomocą algorytmów genetycznych znaleźć jej optima [1]. Przy czym funkcja, której ekstremum jest poszukiwane nie musi być funkcją gładką, ciągłą, ani jednomodalną (jak ma to miejsce w przypadku metod analitycznych) (rys.2.).



Rys.2. Przykłady funkcji których ekstremum może być poszukiwane za pomocą algorytmów genetycznych [3]

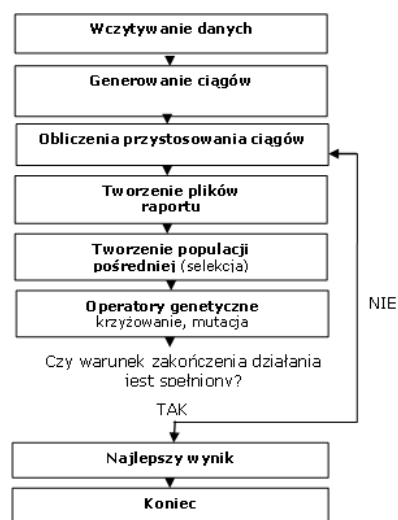
Na klasyczny algorytm genetyczny składają się następujące kroki:

- inicjacja, czyli wybór początkowej populacji chromosomów (pierwsze pokolenie). Inicjacja polega na losowym wyborze żądanej liczby chromosomów reprezentowanych przez ciągi binarne określonej długości (np. liczba populacji $n=6$);
- ocena przystosowania chromosomów polegająca na obliczeniu wartości funkcji przystosowania (celu) dla każdego chromosomu z tej populacji. W przypadku szukania najkrótszej trasy im niższa jest wartość tej funkcji, tym lepsza „jakość” chromosomu;
- sprawdzenie warunku zatrzymania, który zależy od konkretnego zastosowania algorytmu i może wystąpić gdy:
 - uzyskamy wartość maksymalną znaną wcześniej np. $F(x)=0,01$
 - nie poprawia się uzyskiwana wartość (np. jeśli przez trzy generacje wynik się nie zmienia),
 - po określonej liczbie generowanych populacji np. przetwarzanie ma trwać 6 generacji;
- selekcja chromosomów polegająca na wybraniu na podstawie obliczonych wartości funkcji przystosowania, tych chromosomów które będą brały udział w tworzeniu potomków do następnego pokolenia. Najczęściej stosowana jest metoda ruletki. Jak łatwo się domyślić ciągi uznane za „lepsze” mają więcej szans na znalezieniu się w tym pokoleniu i mogą być do niego skopiowane nawet kilkakrotnie, podczas gdy ciągi „gorsze” (o wyższej wartości

funkcji przystosowania) mogą nie dostać się do tego pokolenia w ogóle. Zazwyczaj pokolenie pośrednie ma taki sam rozmiar jak pokolenie pierwsze;

- zastosowanie operatorów genetycznych w stosunku do pokolenia pośredniego:
 - krzyżowanie polega na tym, iż kojarzymy chromosomy w pary „rodziców”, następnie wybieramy losowo punkt krzyżowania wewnątrz ciągu oraz zamieniamy fragmenty ciągów miejscami tak, że powstaje nowe „potomstwo” (z dwóch ciągów – rodziców powstają dwa nowe ciągi – potomstwo). Głównym celem jest wymiana informacji między ciągami tak aby wykorzystując najlepsze cechy ciągów – rodziców, można było utworzyć jeszcze lepsze ciągi – potomstwa,
 - mutacja polega na zmianie wartości jakiejś pozycji w ciągu binarnym (np. z zera na jeden). Celem stosowania mutacji jest wprowadzenie modyfikacji w ciągach, powodującej rozpoczęcie przeszukiwania przestrzeni rozwiązań przez algorytm genetyczny w nowych, do tej pory nie badanych obszarach;
- utworzenie nowej populacji. Chromosomy otrzymane w wyniku działania operatorów genetycznych na chromosomy tymczasowej populacji rodzicielskiej wchodzi w skład nowej populacji.

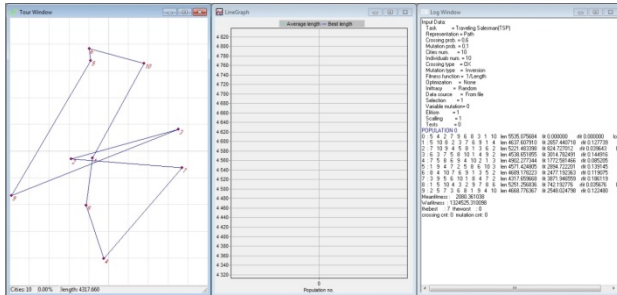
Nowa populacja staje się populacją bieżącą w danej iteracji algorytmu genetycznego. Praca algorytmów genetycznych polega więc na cyklicznym tworzeniu nowych pokoleń, ocenie ciągów kodowych stanowiących te pokolenia i przetwarzaniu ciągów za pomocą operatorów genetycznych. Teoretycznie każde następne pokolenie powinno zawierać ciągi kodowe, których średnia wartość funkcji przystosowania jest „lepsza” niż średnia wartość funkcji przystosowania z poprzedniego pokolenia. Dlatego też prace algorytmu genetycznego można zakończyć wówczas, kiedy kolejne pokolenia nie różnią się od siebie średnią wartością funkcji przystosowania (celu). Można także arbitralnie wyznaczyć liczbę pokoleń, bez zwracania uwagi na zachowanie średniej wartości funkcji celu i zakończyć pracę algorytmu genetycznego po osiągnięciu wymaganej liczby pokoleń (warunek zatrzymania). Schemat blokowy algorytmu genetycznego pokazany jest na rysunku 3.



Rys.3. Schemat blokowy działania algorytmu genetycznego [3]

2. Wykorzystanie aplikacji Visual Genetic do wyznaczania optymalnej trasy przejazdu

Do obliczenia optymalnej trasy składającej się z 10 miast (ponad 360 tys. możliwych rozwiązań) wykorzystana została aplikacja Visual Genetic. Układ miast wraz z wygenerowaną populacją początkową chromosomów ukazany jest na rysunku 4.



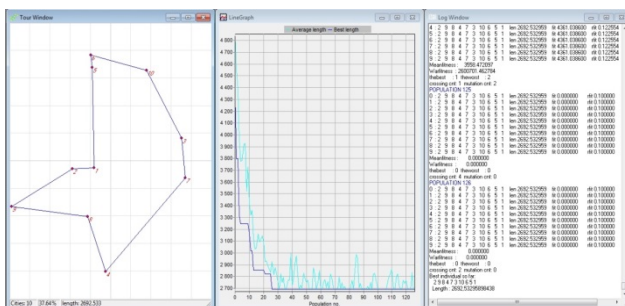
Rys.4. Zrzut ekranowy ukazujący losowy przebieg trasy między 10 miastami oraz wygenerowaną początkową populacją chromosomów

Chromosomami są binarne ciągi kodowe czyli odpowiednio spreparowane dane dotyczące poszukiwanych wartości zmiennych dla których wartość funkcji przystosowania powinna osiągnąć ekstremum (minimum). Każdy chromosom składa się z określonej liczby genów. Każdy gen w naszym przypadku to zakodowane współrzędne jednego z miast. Założone parametry algorytmu genetycznego ukazane są w tabeli 2.

Tab. 2. Parametry wejściowe algorytmu genetycznego

Liczba miast	Funkcja celu	Liczba chromosomów w populacji	Prawdopodobieństwo		Warunek stopu
			krzyżowania	mutacji	
10	$F(x)=x-1$	10	0,6	0,1	100 populacji

Dla założonych warunków wejściowych przebieg optymalnej trasy został obliczony po wygenerowaniu 126 populacji (rys.5.).



Rys.5. Zrzut ekranowy ukazujący optymalny przebieg trasy oraz wizualizację funkcji celu i strukturę poszczególnych populacji chromosomów

Optymalna (najkrótsza) trasa przejazdu wygląda następująco: 2-9-8-4-7-3-10-6-5-1 (długość: 2692,5) i jest krótsza od trasy początkowej (populacja 0) o 1625,2 jednostki czyli o 37% . Łatwo

sobie wyobrazić jakie pociąga to za sobą korzyści w postaci krótszego czasu przejazdu czy mniejszego zużycia paliwa.

Można oczywiście próbować jeszcze zoptymalizować działanie samego algorytmu poprzez zmianę liczby chromosomów w populacji, prawdopodobieństwa zastosowania operatorów genetycznych czy innego określenia warunku zatrzymania pracy algorytmu genetycznego. Przykładowo wzrost liczby chromosomów w populacji wpłynął podczas symulacji na:

- wyraźne zwiększenie czasu pracy algorytmu,
- spadek liczby populacji po której uzyskuje się rozwiązanie . Ponadto wraz ze wzrostem wielkości populacji nie zaobserwowano znacznego wpływu na „jakość” znalezionego rozwiązania. Wydaje się, że najbardziej optymalnym działaniem jest przyjęcie liczby chromosomów w populacji w przedziale 10 – 20. Przeanalizowaną wielkość populacji ukazuje tabela 3.

Tab. 3. Liczba chromosomów w populacji podczas procesu optymalizacji parametrów algorytmu

Liczba miast	Funkcja celu	Liczba chromosomów w populacji	Prawdopodobieństwo		Warunek stopu
			krzyżowania	mutacji	
10	$F(x)=x-1$	2	0,6	0,1	500 populacji
		20			
		50			
		100			
		250			

Z kolei stopniowe zwiększanie prawdopodobieństwa wystąpienia krzyżowania i mutacji do wartości 0,8/0,3 sprawia, iż znajdowane przez algorytm trasy są coraz krótsze (tab.4.). Powyżej tych wartości „jakość” rozwiązań nieznacznie się obniża.

Tab. 4. Przeanalizowane współczynniki prawdopodobieństwa operatorów genetycznych w procesie optymalizacji pracy algorytmu

Liczba miast	Funkcja celu	Liczba chromosomów w populacji	Prawdopodobieństwo		Warunek stopu
			krzyżowania	mutacji	
10	$F(x)=x-1$	10	0,1	0,1	500 populacji
			0,7	0,2	
			0,8	0,3	
			0,9	0,4	
			1,0	0,5	

Warto także zaznaczyć, że operacje zwiększania prawdopodobieństwa wystąpienia operatorów genetycznych nie wpływają istotnie ani na czas pracy algorytmu ani na liczbę populacji po wygenerowaniu których znalezione jest rozwiązanie.

Podsumowanie

Algorytmy genetyczne radzą sobie z większością trudności, jakie napotykanne są przy stosowaniu algorytmów klasycznych. Mają one wyjątkową zdolność łatwej adaptacji i mogą być stosowane przy rozwiązywaniu złożonych nieliniowych i wielowymiarowych

problemów inżynierskich. Wydaje się, że w niedalekiej przyszłości rola technik opartych na algorytmach genetycznych będzie wzrastać, a tym samym należy oczekiwać, że aplikacje bazujące na zasadach ewolucyjnych osiągną coraz większą popularność [1]. W transporcie mogą być wykorzystywane do bardzo szybkiego wyznaczania optymalnych tras przejazdu z uwagi na to, że stosują probabilistyczne, a nie deterministyczne reguły wyboru, przy czym istotny jest dobór właściwych parametrów ich pracy (przede wszystkim liczby chromosomów w populacji oraz prawdopodobieństwa mutacji i krzyżowania). Ponadto algorytmy te prowadzą poszukiwania, wychodząc nie z pojedynczego punktu, lecz z pewnej ich populacji. Ma to istotne znaczenie także z uwagi na to, że sytuacje na drodze mogą się zmieniać w czasie (być dynamiczne), a to z kolei pociąga za sobą konieczność szybkiego otrzymywania korzystnych rozwiązań.

Bibliografia

1. Figielska E.: Algorytmy ewolucyjne i ich zastosowania. Zeszyty Naukowe Warszawskiej Wyższej Szkoły Informatyki, Zeszyt 1, 2006
2. Fiori de Castro H, Cavalca K.: Availability optimization with genetic algorithm. International Journal of Quality & Reliability Management, 20 (2003) 847-863
3. Knosala R.: Zastosowanie metod sztucznej inteligencji w inżynierii produkcji. Wydawnictwa Naukowo-Techniczne, 2002
4. Król A., Metody sztucznej inteligencji w projektowaniu harmonogramu przewozów dla dynamicznych potrzeb transportowych. AUTOBUSY – Technika, Eksploatacja, Systemy Transportowe 2013, nr 3.
5. Mc Call J.: Genetic algorithms for modelling and optimization. Journal of Computational and Applied Mathematics, 184 (2005) 205-222
6. Orłowski C., Lipski J., Loska A.: Informatyka i komputerowe wspomaganie prac inżynierskich. Polskie Wydawnictwa Ekonomiczne, 2012
7. Pecolt S., Głowiński S., Optymalizacja sterowania silownikiem elektromagnetycznym przy wykorzystaniu algorytmów genetycznych. AUTOBUSY – Technika, Eksploatacja, Systemy Transportowe 2013, nr 3.

Autor:

Dr **Konrad Gauda** – Wyższa Szkoła Ekonomii i Innowacji w Lublinie, Wydział Transportu i Informatyki.

The role of genetic algorithms in the process of optimization determining driving routes

The problem of the article concerns the possibility of using genetic algorithms in the optimization process of determining driving routes. Described is the idea of genetic algorithms and the principle their activities. Shown is also the process of optimizing exemplary route with the use of Visual Genetic application.

Key words: genetic algorithms, optimization, transport