

Niektóre osobliwości algorytmów genetycznych na przykładzie zagadnień logistycznych

Certain peculiarities of the genetic algorithms based on logistic issues

Anna Gryko-Nikitin

Politechnika Białostocka, Wydział Zarządzania, Katedra Informatyki Gospodarczej
i Logistyki

Abstract

The aim of the following article is to systematize the concepts connected with modern genetic algorithms together with presenting examples of their applications in logistic issues. Using appropriate tools for solving logistic problems is a basic condition of development and/or endurance on the market with an intensive competition. According to the author, modern genetic algorithms are effective optimizing, ransacking and teaching tools which may have a broader use in logistics. This paper also discusses the matter of genetic algorithm's parameters as a factor determining the effectiveness of their application in logistic issues. The weak and strong points of genetic algorithms in the context of their practical usage has also been pointed out. Presented article is based on literature studies concerning modern concepts of genetic algorithms and their applications.

Keywords: *genetic algorithms, logistic system, optimization.*

Wstęp

Algorytmy genetyczne należą do grupy narzędzi optymalizujących, przeszukujących i uczących, które są wykorzystywane w wielu różnych zagadnieniach łącznie z zagadnieniami logistycznymi. Mocną stroną algorytmów genetycznych jest ich prostota połączona z efektywnością. Szeroki wachlarz zastosowań algorytmów genetycznych jest immanentną konsekwencją wykorzystywania w pracy algorytmu jedynie okrojonej informacji o zadaniu i przekazywaniu tej informacji następnym pokoleniom w postaci informacji o „wygranej” (Goldberg 1995).

W logistyce, rozumianej jako „proces sterowania przepływem dóbr od producenta surowców, poprzez wszystkie fazy produkcji i handlu, do końcowego odbiorcy w taki sposób, aby żądany towar znalazł się w odpowiedniej ilości, miejscu i czasie wystąpienia popytu, przy minimalnych kosztach” (Ficoń 2008), algorytmy genetyczne są wykorzystywane przede wszystkim jako narzędzie wspomagające procesy optymalizacji (np. zadanie transportowe, problem komiwojażera), bądź jako instrument wspomagający proces podejmowania decyzji taktyczno-operacyjnych np. prognozowanie popytu (Chodak 2002).

Na przestrzeni lat algorytmy genetyczne ulegały różnym transformacjom. Współczesne algorytmy nie przypominają klasycznego algorytmu genetycznego opracowanego

w 1975 r. przez Hollanda. Dlatego w pracy podjęto próbę usystematyzowania pojęć związanych ze współczesnymi algorytmami genetycznymi korzystając z przykładów wykorzystania tego narzędzia w zagadnieniach logistycznych.

Prosty algorytm genetyczny

Algorytmy genetyczne należą do klasy algorytmów stochastycznych, polegających na losowym przeszukiwaniu przestrzeni możliwych rozwiązań. Kierunek poszukiwań określany jest przy wykorzystaniu probabilistycznych reguły. Zasada działania klasycznych algorytmów genetycznych (AG) została zaczerpnięta z natury i naśladuje takie procesy jak dobór naturalny i dziedziczenie. Rozwiązanie optymalne wyznaczone jest na drodze ewolucji grupy początkowych propozycji rozwiązań (Goldberg 1995). W ogólnym schemacie wykorzystania algorytmów genetycznych przy rozwiązywaniu rzeczywistych problemów można wyróżnić dwie fazy: fazę wstępną, polegającą na sprecyzowaniu problemu i dostosowaniu go do terminologii używanej w AG oraz utworzeniu początkowej populacji oraz fazę II – umownie nazwaną fazą poszukiwania rozwiązań (Chodoń, Kwaśnicki 2002). Fazę wstępną rozpoczyna ustalenie zakresu zmienności i dokładności rozwiązań. W kolejnym kroku należy zdefiniować ograniczenia i określić reprezentację osobników. Każdy osobnik jest pewnym ciągiem kodowym zwanym chromosomem reprezentującym określone wartości zmiennych decyzyjnych. Najprostszym sposobem odwzorowania jest kodowanie binarne polegające na przypisaniu poszczególnym genom wartości 0 lub 1. Kolejnym etapem tej fazy jest zdefiniowanie funkcji przystosowania, określenie rozmiaru populacji, sposobu i prawdopodobieństwa krzyżowania oraz mutacji oraz określenie kryterium stopu.

Fazę II składającą się z oceny osobników, procesu reprodukcji oraz zastosowania operatorów genetycznych, rozpoczyna wygenerowanie losowej populacji oraz jej ocena pod względem przystosowania. Kolejnym krokiem tej fazy jest operacja selekcji osobników do puli rodzicielskiej. Znalezienie dobrego rozwiązania w zadawalającym czasie w dużej mierze zależy od zastosowanej metody selekcji. Osobniki z bieżącej populacji wybierane są do nowego pokolenia względem funkcji dopasowania. Kryterium wyboru zazwyczaj dobierane jest w taki sposób, aby preferować osobniki z wyższą wartością wskaźnika przystosowania.

Jedną z częściej stosowanych metod selekcji jest wybór losowy z powtórzeniami. W metodzie tej prawdopodobieństwo znalezienia się w populacji potomnej danego osobnika jest proporcjonalne do wskaźnika przystosowania ciągów kodowych. Równoważne jest to wymarcie osobników najsłabszych przy jednoczesnym premiowaniu osobników „lepiej” przystosowanych poprzez wprowadzanie przez nich większej liczby potomków. Jest to zgodne z twierdzeniem o schematach, według którego: „krótkie, niskiego rzędu i oceniane powyżej średniej schematy uzyskują wykładniczo rosnącą liczbę łańcuchów w kolejnych pokoleniach” (Michalewicz 1999).

Mankamentem tej metody jest zmniejszenie różnorodności populacji co może skutkować przedwczesną zbieżnością do lokalnego optimum. Przykładowa realizacja algorytmiczna omawianej operacji polega na symulacji ruletki, gdzie każdemu osobnikowi z populacji odpowiada sektor o rozmiarze proporcjonalnym do jego przystosowania. Sektor przypisany poszczególnym chromosomom na tarczy ruletki określany jest zazwyczaj jako stosunek funkcji oceniającej i -osobnika do sumy funkcji oceniających wszystkich osobników. Wielkość tą zazwyczaj wyraża się procentowo w następujący sposób (Goldberg 1995):

$$v(ch_i) = p_s(ch_i) \cdot 100\%$$

przy czym:

$$p_s(ch_i) = \frac{F(ch_i)}{\sum_{j=1}^N F(ch_j)}$$

gdzie:

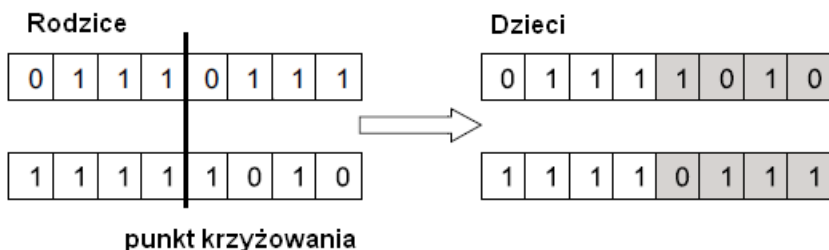
$p_s(ch_i)$ - prawdopodobieństwo selekcji chromosomu ch_i ,

$F(ch_i)$ - wartość funkcji przystosowania chromosomu i -tego,

N - liczebność populacji.

Wyselekcjonowani rodzice podlegają operacji krzyżowania. Operacja krzyżowania jest transformacją wieloargumentową, pozwalającą z osobników rodzi-

cielskich otrzymać osobniki potomne, które wejdą do następnego pokolenia. W klasycznym algorytmie genetycznym krzyżowanie przebiega w dwóch etapach. Na początku należy określić z pewnym zadanym prawdopodobieństwem p_c , które chromosomy będą poddane operacji krzyżowania, poczym wybrane chromosomy należy losowo sparować. W następnym kroku losuje się punkt, w którym dokonuje się wymiany materiału genetycznego (rys. 1).



Źródło: opracowanie własne na podstawie: Goldberg (1995).

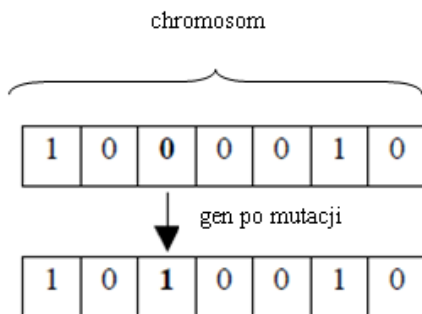
Rys. 1. Operacja krzyżowania binarnego dla kanonicznego algorytmu genetycznego

Krzyżowanie proste może prowadzić do tego, że niektóre ciągi kodowe nigdy nie zostaną osiągnięte, dlatego też opracowano inne bardziej efektywne sposoby rekombinacji genów takie jak: krzyżowanie dwu-punktowe, wielopunktowe, równomierne. Ciekawą modyfikacją operatora krzyżowanie jest krzyżowanie równomierne. Ten rodzaj krzyżowania wprowadza pojęcie losowego wzorca dziedziczenia. Wzorec wskazuje, które geny dziedziczone są od pierwszego rodzica, a które od drugiego.

Operacją wtórną względem krzyżowania jest operacja mutacji, polegająca na losowej zamianie genów. Operacja ta zależy od przyjętej reprezentacji osobnika. Dla elementarnego algorytmu genetycznego, przy binarnej reprezentacji chromosomu $\Omega = \{0,1\}^n$, dla każdego genu w chromosomie podejmuje się decyzję o jego modyfikacji z pewnym prawdopodobieństwem p_m . W zależności od podjętej decyzji może dojść do negacji bitu bądź zamiany bitu na losowy (Arabas 2001).

Jedną z częściej stosowanych sposobów mutacji jest mutacja równomierna (rys. 2). Każdy gen w chromosomie ma równe szanse na to, by ulec procesowi mutacji, zgodnie z prawdopodobieństwem jej wystąpienia. W celu dokonania mutacji dla każdego genu losuje się liczbę z przedziału (0, 1) i jeśli wylosowana licz-

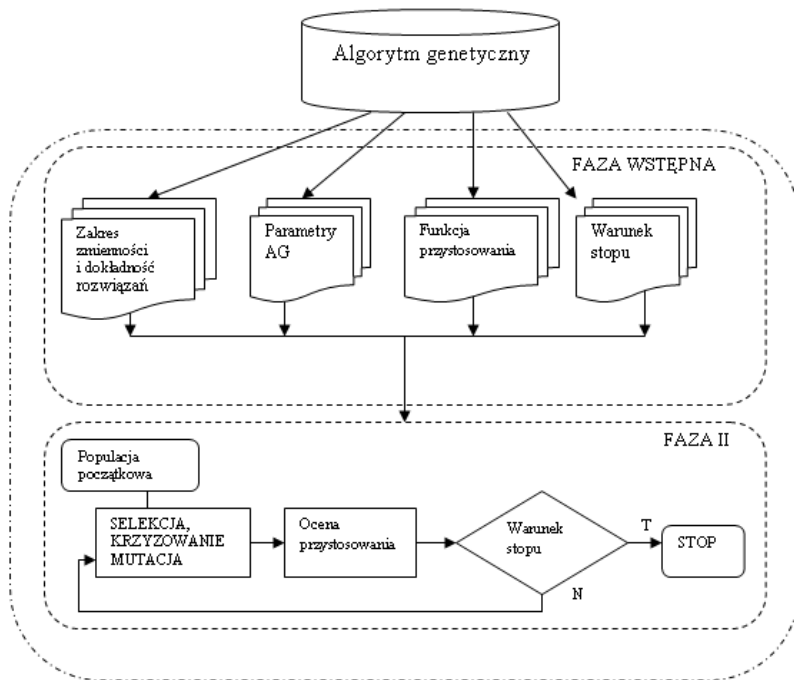
ba jest mniejsza lub równa prawdopodobieństwu mutacji, to wartość danego genu ulega zmianie.



Źródło: opracowanie własne na podstawie: Michalewicz (1999).

Rys. 2. Operacja mutacji

Mutacja, mimo że jest operacją o drugorzędnym znaczeniu, pozwala na wprowadzenie nowego materiału genetycznego zwiększając tym samym różnorodność w populacji. Utrzymanie różnorodności w populacji jest kluczowe dla efektywnego przetwarzania algorytmu genetycznego, gdyż wraz ze wzrostem podobieństwa osobników, prawdopodobieństwo znalezienia nowych rozwiązań zazwyczaj spada. Po wykonaniu mutacji populacja składająca się z nowych osobników podlega ocenie. Wszystkie kroki fazy II zostają powtórzone dopóki nie osiągnie się kryterium stopu. Faza II zostaje zakończona w momencie, gdy zostanie znalezione satysfakcjonujące rozwiązanie lub nastąpił warunek końca algorytmu. Schemat działania algorytmu genetycznego przedstawia rysunek 3.



Źródło: opracowanie własne na podstawie: Goldberg (1995), Chodak i Kwaśnicki (2002).

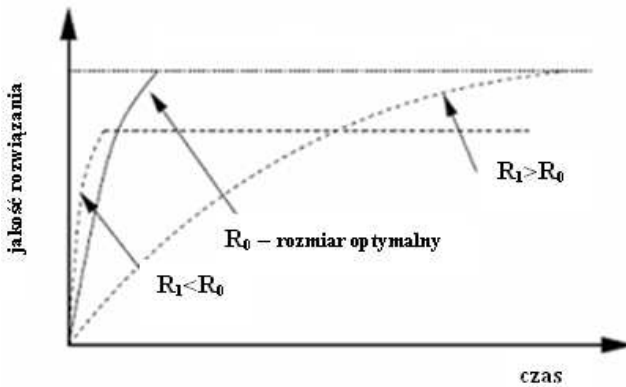
Rys. 3. Algorytm genetyczny – schemat działania

Osobliwości algorytmów genetycznych

Osobliwą cechą algorytmów genetycznych jest to, że otrzymane wyniki zależą w dużym stopniu od obranych parametrów. Proces poszukiwania najlepszego rozwiązania w algorytmach genetycznych jest, więc uzależniony od efektów prac na etapie fazy wstępnej. Źle przeprowadzona faza I wpływa na obniżenie jakości otrzymanego wyniku oraz na czas potrzebny na jego otrzymanie. W literaturze tematu można znaleźć wiele prac poświęconych zagadnieniom fazy wstępnej. Są to prace traktujące m.in. o sposobach doboru parametrów (Srinivas, Patnaik 1998) jak i również wykazujące interakcje pomiędzy nimi (Deb, Agrawal 1998).

Istotnym parametrem algorytmów genetycznych jest wielkość populacji. Rozmiar populacji determinuje między innymi jakość otrzymanego rozwiązania

oraz czas potrzebny na jego otrzymanie. Rysunek 4 przedstawia wpływ rozmiaru populacji na jakość rozwiązania i czas potrzebny na jego otrzymanie.



Źródło: opracowanie własne na podstawie: Lobo i Goldberg (2004).

Rys. 4. Wpływ rozmiaru populacji na jakość rozwiązania i czas

Zaniżenie rozmiaru populacji prowadzi co prawda do skrócenia czasu obliczeń, ale odbywa się to kosztem obniżenia jakości rozwiązania. Zawyżenie rozmiaru populacji prowadzi wprawdzie do poprawy jakości rozwiązania, ale czas potrzebny na jego otrzymanie znacznie się wydłuża. Kompromis może stanowić rozwiązanie polegające na manipulowaniu wielkością populacji poprzez uwzględnienie procesu starzenia się osobników (Arabas; Michalewicz; Mulawka 1994). W pracy pt. *GAVaPS-a genetic algorithm with varying population size* autorzy zaproponowali algorytm genetyczny ze zmienną liczebnością populacji. Rozmiar populacji zastąpiono wiekiem osobnika oraz maksymalnym czasem przeżycia. Zaproponowane podejście pozwala na redukowanie rozmiaru populacji w momencie znalezienia otoczenia optimum, skracając czas obliczeń.

Na uwagę zasługują również kwestia odwzorowania chromosomów. Osobliwości chromosomu zostaną przybliżone na przykładzie prognozowania popytu oraz zagadnienia komiwojażera.

W prostym algorytmie genetycznym chromosomy reprezentowane są przez binarne ciągi kodowe. Oznacza to, że chromosom składa się z genów przyjmujących jedną z dwóch wartości 0 lub 1. Liczba genów w chromosomie zależy od wielkości zadania oraz od obranej metody kodowania wartości zmiennej decyzyjnej. Przykładowo w zadaniu polegającym na znalezieniu maksimum pewnej funkcji $f(x)$, gdzie x może przyjmować wartości od 0 do 15 zakładając kodowa-

nie za pomocą kodu dwójkowego, chromosom składałby się z 4 genów. Chromosomowi o strukturze 1111 odpowiadałaby wartość x równa 15, zaś chromosomowi o strukturze 0000 – wartość x równa 0.

Ciekawym rozwiązaniem kwestii kodowania jest też podział chromosomu na bloki o stałej długości. Tym sposobem można w chromosomie zakodować dowolną liczbę identyfikowanych parametrów. Ten sposób odwzorowania wartości zmiennych wykorzystano m.in. w pracy pt. *Zastosowanie algorytmów genetycznych w prognozowaniu popytu* (Chodak, Kwaśnicki 2002). W artykule funkcję popytu wyrażono wzorem:

$$D = \frac{C + B * t + A * \sin(\omega * t + \varphi)}{P^e}$$

gdzie:

D – przewidywana sprzedaż;

A – amplituda;

t – czas;

ω – częstość;

φ – przesunięcie fazowe;

C – przesunięcie wzdłuż osi odciętych;

B – współczynnik kierunkowy trendu liniowego;

e – współczynnik elastyczności cenowej.

Przy tak zdefiniowanej funkcji popytu otrzymano do identyfikacji przy pomocy algorytmów genetycznych 6 parametrów (A , C , B , e , φ , ω). Każdemu parametrowi przypisano segment składający się z 10 bitów. Rysunek 5 przedstawia przykładowy chromosom.



Źródło: Chodak i Kwaśnicki (2002).

Rys. 5. Przykładowa struktura chromosomu

Dla zagadnienia prognozowania popytu kodowanie binarne z przedstawionym sposobem odwzorowania zadania okazało się całkowicie wystarczające. Są jednak zadania, dla których kodowanie binarne nie jest odpowiednie. Do tego typu zadań należy zagadnienie komiwojażera.

Zadanie komiwojażera (ang. Travelling Salesman Problem) najogólniej mówiąc polega na optymalizowaniu kosztów podróży przy zadanych kosztach przejazdu pomiędzy miastami. Komiwojażer startując z miasta A musi odwiedzić wszystkie miasta na swoim terytorium dokładnie raz i wrócić do punktu startu, przy czym koszt podróży powinien być minimalny. Liczba możliwych tras przy n -miastach jest rzędu $n!$ (zadanie komiwojażera należy do zadań NP-trudnych tzn. nie da się znaleźć rozwiązania w czasie wielomianowym).

W zagadnieniu komiwojażera chodzi o znalezienie najlepszej permutacji miast. Przy reprezentacji binarnej operacja mutacji równomiernej mogłaby prowadzić do trasy niedopuszczalnej, dlatego dla TSP opracowano inne reprezentacje np. reprezentację wektorową (przyległościową, porządkową, ścieżkową). Najbardziej naturalną reprezentacją dla tego zagadnienia jest reprezentacja ścieżkowa. Ciekawą propozycję modyfikacji algorytmu genetycznego zaproponował Michnowicz w swojej pracy pt. *Problem komiwojażera dla kilku centrów dystrybucji* (Michnowicz 2009). Autor rozważał wykorzystanie algorytmu genetycznego dla rozszerzonego zadania TSP. W rozpatrywanym zadaniu z każdego magazynu wysyłany jest tylko jeden komiwojażer, który odwiedza wszystkich przydzielonych mu odbiorców, po czym wraca do miejsca początkowego. W zaproponowanym algorytmie przetwarzana jest populacja składająca się z jednego osobnika. Do inicjacji początkowego rozwiązania bazowego zaproponowano metodę „minimum macierzy kosztów”.

Podsumowanie

Algorytmy genetyczne to uniwersalne narzędzia optymalizujące, przeszukujące i uczące, które przy zastosowaniu odpowiednio dobranych parametrów pozwalają na otrzymanie optymalnych wyników. Nowe techniki i koncepcję rozszerzające pojęcie prostego algorytmu genetycznego, jak chociażby przytoczone w artykule kodowanie bądź zastosowanie metody „minimum macierzy kosztów” do wygenerowania populacji początkowej, pozwalają na dopasowanie algorytmu do potrzeb konkretnego zadania. Kontrowersje budzi jednak czas wykonania algorytmu oraz problem ze złożonością obliczeniową. Odpowiedzią może być przetworzenie równoległe. Rozpowszechnienie informatyki równoległej oraz rosnące możliwości dostępu do systemów obliczeń współbieżnych sugerują zasadność wykorzystania inherentnej natury algorytmów genetycznych i dalszego rozwijania tej koncepcji. Przetwarzanie równoległe prowadzi niewątpliwie do skrócenia czasu obliczeń.

Pozwala również rozwiązywać zadania o dużej złożoności obliczeniowej, a zagadnienia logistyczne niewątpliwie do takich należą.

Piśmiennictwo

1. Arabas J., 2001. *Wykłady z algorytmów ewolucyjnych*, WNT, Warszawa.
2. Arabas J., Michalewicz Z., Mulawka J., 1994. *GAVaPS-a genetic algorithm with varying population size*. Evolutionary Computation, IEEE World Congress on Computational Intelligence, Proceedings of the First IEEE Conference on 27-29, June 1994, s. 73 – 78.
3. Chodak G., Kwaśnicki W., 2002. *Zastosowanie algorytmów genetycznych w prognozowaniu popytu*. Gospodarka Materialowa & Logistyka, nr. 4.
4. Deb K., Agrawal S., 1998. *Understanding interactions among genetic algorithm parameters*, in: W. Banzhaf, C. Reeves (Eds.), *Foundations of Genetic Algorithms 5*, Morgan Kaufmann, San Francisco, CA, s. 265–286.
5. Ficoń K., 2008. *Logistyka ekonomiczna. Procesy logistyczne*, BEL Studio Sp. z o.o., Warszawa.
6. Goldberg D. E., 1995. *Algorytmy genetyczne i ich zastosowania*. WNT, Warszawa.
7. Lobo F. G., Goldberg D. E., 2004. *The parameter-less genetic algorithm in practice*, Information Sciences 167, s. 217–232.
8. Michnowicz E., 2009. *Problem komiwojażera dla kilku centrów dystrybucji*. Prace Naukowe Politechniki Warszawskiej, Warszawa.
9. Michalewicz Z., 1999. *Algorytmy genetyczne + struktury danych = programy ewolucyjne*. WNT, Warszawa.
10. Rutkowski L., 2005. *Metody i techniki sztucznej inteligencji*. PWN SA, Warszawa
11. Srinivas M., Patnaik L. M. 1994, *Adaptive Probabilities of Crossover and Mutation in Genetic Algorithms*, IEEE Transactions on Systems, Man and Cybernetics 24(4).