

Nowa implementacja algorytmu mrówkowego wykorzystująca technologie przetwarzania wieloprocesorowego i rozproszonego w systemie nawigacji

A new implementation of an ant algorithm using multiprocessor and distributed computing technologies in navigation system

Daniel Komar¹

Treść. Artykuł ma na celu przybliżenie czytelnikowi problemu wyboru najlepszej trasy podróży pomiędzy dwoma punktami, która będzie minimalizowała liczbę negatywnych czynników wpływających na osobę kierującą pojazdem. Zaprezentowany zostanie nowo zaimplementowany algorytm mrówkowy, który został przystosowany do wykorzystania w pełni możliwości obliczeniowych współczesnych systemów wieloprocesorowych i rozproszonych. Autor przeprowadzając eksperyment w warunkach rzeczywistych, ukaże wyższość opracowanego rozwiązania nad stosowanym obecnie tradycyjnym systemem nawigacji. Przeprowadzone badania wykazały, że wykorzystywana nowa implementacja algorytmu w znacznym stopniu zmniejsza czas przejazdu i liczbę czynników zakłócających mających bezpośredni wpływ na osobę kierującą pojazdem.

Słowa kluczowe: algorytm mrówkowy, nawigacja, czas podróży, warunki drogowe

Abstract. The purpose of this paper is to give reader an understanding of the problem of the best itinerary selection between two points which will minimize the number of negative factors affecting the person driving a vehicle. The author will present an entirely new implementation of an ant algorithm that was adapted in order to make the most of computational capabilities of modern multiprocessor and distributed systems. Having performed experiments in real-world conditions, the author demonstrates, that the new solution is superior to the traditional navigation system which is still used today. The conducted research showed that the new implementation of the algorithm significantly contributed to reduction of journey times and the number of confounding factors which have a direct impact on the person driving a vehicle.

Keywords: ant algorithm, navigation, journey times, driving condition

1. Wstęp

W ostatnich latach nastąpiło znaczne przyspieszenie procesu urbanizacji, powodującego znaczny wzrost zaludnienia miast. Efektem tego procesu jest bardzo duży wzrost liczby pojazdów uczestniczących w ruchu drogowym, co powoduje powstawanie korków drogowych. Kierowcy, co prawda coraz częściej wykorzystują system nawigacji satelitarnej, który często nie jest doskonały, ponieważ nie uwzględnia zmieniających się warunków drogowych na wyznaczonej trasie. Problem wyznaczania nieoptymalnego rozwiązania względem panujących warunków powoduje powstanie związku przyczynowo skutkowego o znacznych rozmiarach. Głównym efektem tego jest negatywny wpływ na sprawność psychofizyczną kierowcy. Przekłada się to na spadek sprawności psychicznej i powoduje błędną percepcję osoby prowadzącej pojazd, co może prowadzić do podejmowania wielu pochopnych i błędnych decyzji w istotnych dla bezpieczeństwa sytuacjach drogowych.

W artykule zostanie przedstawiony algorytm mrówkowy zastosowany w systemie nawigacji, który został zaadaptowany do wyznaczania optymalnej trasy podróży z uwzględnieniem wielu istotnych parametrów zmieniają-

cych się w czasie rzeczywistym. W związku z tym podczas każdorazowego generowania nowego kierunku jazdy pomiędzy dwoma punktami na mapie algorytm stara się dostarczyć jak najlepsze rozwiązanie problemu. Zostaną również przedstawione czynniki, które zostały sklasyfikowane jako odgrywające dużą rolę w ruchu pojazdów.

W dalszej części zostanie zaprezentowany eksperyment przeprowadzony w warunkach rzeczywistych. Przeprowadzona próba była zrealizowana z udziałem dwóch niezależnych systemów nawigacji. Pierwszy wykorzystujący tradycyjne i obecnie stosowane na szeroką skalę rozwiązania oraz drugi wykorzystujący algorytm mrówkowy z uwzględnieniem wielu relewantnych czynników. Przeprowadzone badania mają na celu zbadanie możliwości usprawnienia komunikacji w dużych ośrodkach miejskich.

2. Tło koncepcji

Inspiracją do rozpoczęcia projektu systemu nawigacji wykorzystującego algorytm mrówkowy z uwzględnieniem aktualnie panujących warunków drogowych były psycho-

¹Wydział Informatyki, Wrocławska Wyższa Szkoła Informatyki Stosowanej, ul. Wejherowska 28, 54-239 Wrocław, gordon1x@poczta.fm

logiczne badania kierowców prowadzone przez Zakład Psychologii Transportu Drogowego Instytutu Transportu Samochodowego w Warszawie. Analizowano w nich występujące w ruchu drogowym zakłócające czynniki mające zasadniczy wpływ na kierującego pojazdem. Przyjmując definicję bezpiecznego kierowcy, który ma na celu przeprowadzić bezkolizyjnie pojazd pomiędzy dwoma punktami (początkowym i docelowym) [2]. Kierowca nie mogąc przewidzieć splotu wszystkich czynników może natrafić na bodziec powodujący dezorganizację zamierzonych działań. Zaburzenia takie jak korek drogowy, utrudnienia w ruchu powodują u osób kierujących występowanie indywidualnych stanów emocjonalnych. Bardzo negatywny skutek na kierowcę mają silne emocje, spowodowane zbyt długo występującym zaburzeniem. Wpływają one w znaczący sposób na osłabienie sprawności psychofizycznej. Mocno niepożądanym skutkiem tego może być obniżenie szybkości reakcji, podzielności uwagi i koncentracji. Resultatem tego jest znaczna zmiana zachowania kierowcy, który chce jak najszybciej osiągnąć cel swojej podróży. Zapomina o bezpieczeństwie i wykonuje w wielu przypadkach ryzykowne manewry. Podejmowane przez niego decyzje przestają być racjonalne, co naraża innych uczestników ruchu na kolizję.

Duży wpływ na wybór algorytmu mrówkowego do realizacji zadania nawigacji miały artykuły naukowe i badania przeprowadzone przez Marco Dorigo. W jego pracach można odnaleźć główne zasady jakie spełniać musi algorytm mrówkowy, aby umożliwić rozwiązanie problemu wyszukiwania najkrótszej trasy pomiędzy dwoma punktami [1, 3, 4, 5]. Przedstawione w pracach różne modyfikacje algorytmu wykorzystują podejście probabilistyczne, a w implementacji programowej generator liczb losowych. Powoduje to, że znalezione rozwiązanie niekoniecznie jest najlepsze, ale w stosunku do innych algorytmów istnieje prawdopodobieństwo znalezienia zadowalającego rozwiązania w krótszym czasie.

3. Algorytm mrówkowy

Na potrzeby projektu została opracowana implementacja algorytmu mrówkowego, która była wzorowana na głównych zasadach funkcjonowania zawartych w pracach i publikacjach naukowych Marco Dorigo [3, 4].

Główną ideą algorytmu mrówkowego jest wykorzystanie zasad samoorganizacji występujących w naturalnym ekosystemie kolonii mrówek. Zastosowane zasady mają na celu koordynację ich sztucznych, cyfrowych odpowiedników, które umożliwią rozwiązywanie problemów [4]. Inspiracją były różne zachowania mrówek: żerowanie, podział pracy, sortowanie i transport kooperacyjny. Biologowie prowadzący badania wykazali, że mrówki koordynują swoją pracę za pomocą pośredniego mechanizmu porozumiewania przy wykorzystaniu zmian środowiska zwanym stygmergią. Najlepsze rezultaty osiągnięto przy opracowaniu algorytmu bazującego na mechanizmie występującym podczas żerowania i jest on ukierunkowany

na rozwiązywanie problemu optymalizacji kombinatorycznej [4].

Za główny punkt nowej implementacji przyjęto przystosowanie algorytmu do wykorzystania możliwości nowoczesnych systemów wieloprocesorowych, przetwarzania równoległego i rozproszonego. Dzięki temu rozwiązaniu będzie możliwe znaczne przyspieszenie wykonywanych operacji. Pozwoli to wykorzystać moc obliczeniową procesorów wielordzeniowych, systemów wieloprocesorowych oraz klastrów obliczeniowych.

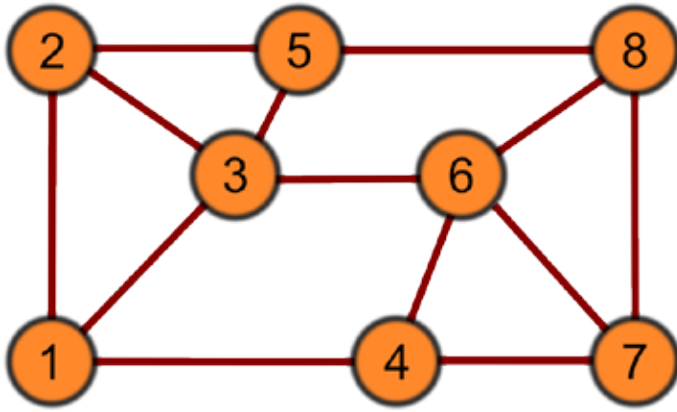
Zaimplementowany algorytm w procesie decyzyjnym uwzględnia również najważniejsze czynniki takie jak: odległość pomiędzy wyznaczonymi węzłami komunikacyjnymi, klasa drogi, maksymalna prędkość pojazdu, występowanie korka drogowego, utrudnienia drogowe. Należy podkreślić, że uwzględniany jest również rodzaj węzła komunikacyjnego takiego, jak skrzyżowanie równorzędne, o ruchu okrężnym, czy kierowane znakami lub sygnalizacją świetlną. Zależnie od jego rodzaju odpowiednio uwzględniono średni czas oczekiwania osoby prowadzącej pojazd na przejazd przez dany węzeł (skrzyżowanie). Należy zauważyć, że wszystkim uwzględnianym parametrom zostały przypisane odpowiednie wagi. Zdefiniowane we wcześniejszym etapie wartości wag poszczególnych parametrów pozwalają obliczyć wagę na odcinku zlokalizowanym pomiędzy dwoma indywidualnymi węzłami. Tym samym staje się możliwe wyznaczenie wag pomiędzy wszystkimi połączonymi parami węzłów zaznaczonych na fragmencie mapy.

Z założenia algorytm nie będzie wyznaczał statycznej trasy pomiędzy punktem startowym a docelowym, lecz będzie w czasie jazdy aktualizował wyznaczoną trasę wykorzystując zmieniające się dane o utrudnieniach występujących w ruchu drogowym. Przy wyborze trasy przejazdu algorytm szacuje przewidywany czas podróży. Jeżeli jego wartość będzie mniejsza to algorytm wyznaczy nową lepszą trasę podróży. Warto zauważyć, że uaktualnienie trasy wiąże się również z nowym oszacowaniem czasu podróży z węzła komunikacyjnego, w którym się w danej chwili pojazd znajduje, a punktem docelowym. Należy zwrócić uwagę, że obliczenia są wykonywane tylko i wyłącznie podczas pokonywania trasy pomiędzy węzłami w ściśle określonym czasie. Algorytm wyznacza nową trasę, kierowca zauważa informację, podejmuje decyzję i reaguje przed dojazdem do węzła komunikacyjnego.

4. Ogólna struktura algorytmu

Zanim zostanie przedstawiona ogólna struktura algorytmu należy zdefiniować pojęcie grafu. Graf jest to para złożona z dwu zbiorów: wierzchołków $V(G)$ i krawędzi $E(G)$. W grafie prostym krawędzie te są nieuporządkowanymi parami wierzchołków, natomiast w grafie skierowanym są uporządkowanymi parami wierzchołków [6]. W przypadku opisywanego problemu V zawiera zbiór węzłów komunikacyjnych, natomiast E zawiera zbiór wag obliczonych na podstawie charakterystycznych wartości parametrów

dla ruchu drogowego. Na potrzeby projektu wykorzystano strukturę grafu nieskierowanego (rys. 1.1.),



Rys. 1.1. Przykładowa graf dla algorytmu mrówkowego
Fig. 1.1. Sample the graph for ant algorithm

- (1) W grafie G występuje unikalny węzeł, który jest oznaczony jako startowy.
- (2) W grafie G występuje unikalny węzeł, który jest oznaczony jako końcowy.
- (3) Niech T będzie zbiorem węzłów ścieżki t pomiędzy dwoma wyznaczonymi węzłami (1), (2) w grafie G spełniając następujące założenia:
 1. Pierwszy element ścieżki t jest węzłem oznaczonym jako startowy w grafie G (1).
 2. Ostatni element ścieżki t jest węzłem oznaczonym jako końcowy w grafie G (2).
 3. W ścieżce t każdy z węzłów grafu G może wystąpić tylko raz.
 4. Ścieżka t niekoniecznie zawiera wszystkie węzły znajdujące się w grafie G .
 5. Elementy ścieżki t należą do liczb naturalnych z zerem włącznie i tym samym jest spełnione założenie:

$$T = \{t : t \in \mathbb{N}_0\}$$

ze względu na możliwość optymalizacji złożoności wykonywanych obliczeń. Należy pamiętać, że muszą zostać spełnione ściśle określone założenia:

List. 1.1. Uproszczony pseudokod algorytmu w języku programowania Pidgin ALGOL
List. 1.1. The simplified pseudocode of the algorithm in Pidgin ALGOL programming language

begin

```

comment Wczytywanie danych i parametrów działania programu:
comment I - liczba iteracji, M - wagi pomiędzy węzłami komunikacyjnymi
comment S - obecny węzeł komunikacyjny, K - końcowy węzeł komunikacyjny
comment P - liczba procesów
read M, I, S, K, P

comment Zmienne, do której mają dostęp wszystkie procesy programu
koszt ← ∅
trasa ← ∅

for i ← 0 step 1 until (P - 1) do
  beginPARA
    comment Funkcja generuje macierz
    feromon ← GenerujMacierz(M)

    for m ← I step -1 until 0 do
      comment Funkcje generujące trasę podróży
      temp ← GeneratorTrasy(M, S, K)
      route ← Trasa(M, temp)
      comment Funkcja oblicza koszt trasy podróży
      nowaTrasaKoszt ← TrasaKoszt(M, temp)

      if nowaTrasaKoszt < trasa and route ≠ 1 then
        begin
          comment Zablokuj dostęp procesom do zmiennych
          koszt ← nowaTrasaKoszt
          trasa ← temp
          comment Odblokuj dostęp procesom do zmiennych
          comment Funkcja aktualizuje wartość feromonu
          RozłóżFeromon(feromon, trasa)
        end

      comment Aktualizacja feromonu zmieniającego się w czasie
      ParowanieFeromonu(feromon)
    endPARA
  write trasa
end

```



Rys. 1.2. Przypadek w którym mrówka nie może kontynuować podróży (mapa: <http://osmapa.pl/>)

Fig. 1.2. The case in which an ant cannot continue the journey (map: <http://osmapa.pl/>)

Należy zauważyć, że może pojawić się wyjątek w działaniu algorytmu. Istnieje prawdopodobieństwo wystąpienia sytuacji, w której cyfrowa mrówka z obecnego węzła nie może udać się do kolejnego, gdyż już był wcześniej odwiedzony. W tej sytuacji, aby zapobiec zapętleniu algorytmu podróż mrówki zostanie przerwana, a trasa zostanie oznaczona jako błędna. Nie ma to żadnego wpływu na działanie pozostałych osobników w kolonii. Opisywany przypadek ilustruje rysunek 1.2.

Ogólną strukturę algorytmu opisywanego w tym artykule przedstawia listing 1.1. Prezentowany pseudokod został napisany w języku programowania wysokiego poziomu Pidgin ALGOL.

5. Eksperyment w warunkach rzeczywistych

Na potrzeby niniejszego artykułu z aglomeracji miejskiej wybrano fragment miasta Wrocław, w którym wytypowano istotne węzły komunikacyjne (rys. 1.3). Dzięki temu prostemu zbiegowi czytelnik uzyskał wyrazisty przekaz dotyczący metodologii przeprowadzanego badania.

Pomiędzy wszystkimi wyznaczonymi węzłami przeprowadzono analizę tras, umożliwiającą wyznaczenie wskaźników występowania korków i utrudnień drogowych. Eksperyment przeprowadzono wykorzystując dwa pojazdy, z czego jeden został wyposażony w tradycyjny system nawigacji, a drugi wykorzystywał system nawigacji² oparty na opisanym wcześniej algorytmie mrówkowym. Próbę wykonano w godzinach popołudniowego szczytu, gdyż w tym czasie poruszanie się po mieście sprawia najwięcej problemów ze względu na występujące utrudnienia w ruchu drogowym. Po wyznaczeniu trasy pojazdy wyruszyły jednocześnie z punktu oznaczonego symbolem „S” na fragmencie planu miasta i podążały do punktu docelowego oznaczonego symbolem „K”.

Tab. 1.1. Wyznaczona trasa przejazdu przez tradycyjny system nawigacji

Tab. 1.1. The mapped travel route by the traditional navigation system

Trasa przejazdu	Szacowany czas
S, 48, 5, 50, 8, 13, 16, 17, 20, 31, 37, 41, 42, K	600 sekund

Tradycyjny system nawigacji wyznaczył statyczną trasę przez węzły (tab. 1.1) nie uwzględniając jakichkolwiek



Rys. 1.3. Fragment planu miasta z naniesionymi węzłami komunikacyjnymi (mapa: <http://osmapa.pl/>)

Fig. 1.3. Part of the city map with plotted traffic junctions (map: <http://osmapa.pl/>)

² Autor tekstu używając sformułowania „tradycyjny system nawigacji” ma na myśli rozwiązanie nawigacyjne niewykorzystujące potencjału sztucznej inteligencji i informacji o panujących warunkach drogowych w czasie rzeczywistym, generujące statyczną trasę przejazdu.

warunków drogowych. Szacowany czas przejazdu całej trasy został określony na 600 sekund. Jednakże, rzeczywisty czas przejazdu wyniósł 1020 sekund, co wydłużyło przejazd o 70%. Na trasie przejazdu pojazd natrafił na utrudnienia drogowe pomiędzy węzłami: 50 – 8, 8 – 13, 13 – 16, 20 – 31, 31 – 37. Napotkane utrudnienia miały znaczący wpływ na rzeczywisty czas przejazdu, który znacznie odbiegał od szacowanego czasu. Pokonana trasa przez pojazd miała długość 4,3 km.

Tab. 1.2. Wyznaczone kolejne trasy przejazdu przez nowy system nawigacji

Tab. 1.2. The next mapped travel routes by the new navigation system

Trasa przejazdu	Szacowany czas
S, 48, 4, 7, 10, 18, 22, 31, 37, 41, 42, K	729.72 sekund
48, 5, 4, 7, 10, 9, 14, 15, 20, 33, 34, 36, 38, 42, K	664.64 sekund
5, 4, 7, 10, 18, 22, 31, 37, 41, 42, K	626.72 sekund
4, 7, 10, 18, 22, 31, 37, 38, 42, K	557.16 sekund
7, 10, 18, 22, 31, 37, 41, 42, K	503.52 sekund
10, 18, 22, 31, 35, 36, 38, 42, K	419.40 sekund
18, 22, 31, 35, 36, 38, 42, K	337.40 sekund
22, 31, 35, 36, 38, 42, K	255.00 sekund
31, 35, 36, 38, 42, K	179.08 sekund
35, 36, 38, 42, K	130.04 sekund
36, 38, 42, K	96.00 sekund
38, 42, K	63.76 sekund
42, K	30.08 sekund

Nowy system nawigacji uaktualniał trasę przejazdu w czasie rzeczywistym uwzględniając wszystkie czynniki utrudniające jazdę. Wyznaczone kolejne trasy przejazdu prezentuje tabela 1.2. Wyznaczanie nowej trasy przejazdu następowało pomiędzy węzłami wraz z nowo oszacowanym czasem przejazdu. Rzeczywisty czas przebycia drogi z punktu początkowego do docelowego wyniósł 780 sekund, co wydłużyło przejazd zaledwie o 6,89%, w stosunku do początkowo prognozowanej wartości. Trasa pokonana przez pojazd miała długość 4,2 km.

Pomiar również powtórzono w warunkach porannego szczytu oraz normalnego natężenia ruchu pomiędzy dwoma punktami „S” i „K” uzyskując następujący średni wynik. Dla wykorzystywanego obecnie systemu nawigacji średni realny czas przejazdu trwał 870 sekund, okres podróży zaś wydłużył się średnio o 47% w stosunku do szacowanego czasu. Wszak stosując nowe rozwiązanie uzyskano średnią wartość realnego czasu przejazdu rów-

ną 750 sekund i względem szacowanej wartości czasu, przejazd wydłużył się zaledwie o 2,78%.

Przeprowadzone eksperymenty uwzględniające topologię miasta i jego okolic na trasach przejazdu o długości powyżej 30 km wykazały, że dzięki nowej implementacji czas przejazdu uległ skróceniu w stosunku do ogólnodostępnego systemu nawigacji. Faktem jest, że każda z nawigacji wyznaczała indywidualną trasę przejazdu.

Dzięki temu, że były uwzględniane przez algorytm rzeczywiste warunki drogowe panujące w danym czasie z powodzeniem udało się ominąć występujące utrudnienia. Zmniejszeniu uległa również liczba negatywnych bodźców oddziałujących na stan psychofizyczny osoby kierującej pojazdem. Analizując wyniki badań przeprowadzonych eksperymentów można stwierdzić wyższość nowej implementacji algorytmu mrówkowego nad tradycyjnym rozwiązaniem.

Analizując wyniki badań przeprowadzonych eksperymentów można stwierdzić, że nowa implementacja algorytmu doskonale nadaje się do celów nawigacji w aglomeracjach miejskich.

6. Podsumowanie

Przeprowadzone eksperymenty mające na celu sprawdzenie funkcjonowania algorytmu w warunkach rzeczywistych zakończyły się pełnym sukcesem. Opracowany na potrzeby projektu system nawigacji wyznaczył trasę o krótszym czasie przejazdu. Tym samym przyczynił się na poprawę komfortu jazdy i wpłynął znacząco na samopoczucie kierowcy, co przekłada się również na większe bezpieczeństwo jazdy. Ponadto, z punktu widzenia ekonomicznego czas przejazdu wpływa również na ilość zużytego paliwa. Natomiast krótszy czas przejazdu to nie tylko oszczędności, ale również zmniejszona emisja zanieczyszczeń do atmosfery.

Poszukiwanie najlepszej trasy przejazdu będzie w przyszłości najważniejszym elementem nowoczesnych systemów nawigacji. Algorytmy poszukujące optymalnego rozwiązania z uwzględnieniem wielu parametrów mogą wpłynąć znacznie na zmniejszenie zatorów na drogach w dużych aglomeracjach miejskich poprzez kierowanie ruchu alternatywnymi trasami. Również urządzenia nawigacyjne posiadające wbudowane procesory wielordzeniowe pozwolą na stosowanie rozwiązań przetwarzania równoległego, co znacznie skróci czas generowania rozwiązań w czasie rzeczywistym. Z biegiem czasu stanie się również możliwe pobieranie danych o przepustowości ruchu drogowego dzięki montażowi odpowiednich urządzeń pomiarowych na skrzyżowaniach.

Możliwe, że w bliskiej przyszłości do wyznaczania trasy pojazdów będzie się używać algorytmów niedeterministycznych, które umożliwią wyznaczenie optymalnego w danych warunkach rozwiązania.

Warto dodatkowo zauważyć, że w obecnych czasach do rozwiązania przedstawionego problemu niewystarczająca okazuje się wszechstronna wiedza z jednej, konkretnej

dziedziny nauk. Powstaje więc wrażenie, że zagadnienie jest w swojej istocie bardzo skomplikowane i rzeczywiście w praktyce wymaga zastosowania wiedzy z dziedzin nauk behawioralnych, ścisłych, i współczesnej techniki. Powoduje to, że osoba pracująca nad takim zagadnieniem musi się charakteryzować nieprzeciętną erudycją lub wymagane jest powstanie zespołów badawczych.

Literatura (References)

- [1] E. Bonabeau, M. Dorigo, G. Theraulaz, *Swarm Intelligence From Natural to Artificial Systems*, Oxford University Press 1999.
- [2] J. Bąk, *Psychologiczne badania kierowców*. Bezpieczeństwo pracy nr 6, pp.12-15, 2004.
- [3] M. Dorigo, L. Maria Gambardella, *Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem*, IEEE Transactions on Evolutionary Computation, Vol.1, No.1, pp.53-66, 1997.
- [4] M. Dorigo, T. Stützle, *Ant Colony Optimization*, MIT Press, Cambridge 2004.
- [5] M. T. Jones, *Artificial Intelligence: A Systems Approach*, Infinity Science Press LLC 2008.
- [6] W. Mizerski, W. Sadowski, A. Garbarczyk, B. Tokarska, K. Mazur, *Tablice matematyczne*, Adamantan, Warszawa 2008.