*Studia i Materiały Informatyki Stosowanej, Tom 6, Nr 17, 2014*
*str. 16-21*

# PROBLEMY WYZNACZANIA OPTYMALNEJ TRAJEKTORII OBIEKTU NA OBSZARZE OGRANICZONYM

**Mariusz Dramski**

*Akademia Morska w Szczecinie*
*Instytut Technologii Morskich*
*ul. Wały Chrobrego 1-2, 70-500 Szczecin*
*e-mail: m.dramski@am.szczecin.pl*

**Streszczenie:** *Podstawowym celem systemów informatycznych w transporcie jest wspomaganie procesu przemieszczenia ładunku lub ludzi z punktu początkowego do punktu docelowego. Poza typowymi funkcjami związanymi z informatyczną obsługą przedsiębiorstwa, istnieje szereg metod wspomagających np. dobór optymalnej trasy dla środka transportu. Trasa optymalna to taka, której koszt (uwzględniając nie tylko odległość, ale także i inne czynniki) jest możliwie jak najmniejszy. W niniejszym artykule przedstawione są wyniki dotychczasowych prac prowadzonych w tym zakresie na Akademii Morskiej w Szczecinie.*

**Słowa kluczowe:** *Trasa, optymalna trajektoria, obszar ograniczony, poszukiwanie drogi*

## Problems of the shortest path selection in restricted area

**Abstract:** *The main task of the information systems in transport is supporting the process of the movement of cargo and people from the departure point to the destination. Besides the typical ERP functions, these systems have also some tools like searching for the optimal route etc. The optimal route is when it's total cost (considering different factors) is minimal. In this paper the results of experiments carried out at The Maritime University of Szczecin are described.*
**Keywords:** *Route, path, optimal path, resticted area, path search*

## 1. WSTĘP

Poszukiwanie optymalnej trasy dla obiektu poruszającego się w obszarze ograniczonym jest jednym z ciekawszych problemów optymalizacyjnych z którymi boryka się transport. Na Akademii Morskiej w Szczecinie przeprowadza się szereg badań związanych głównie z transportem morskim, co ma oczywiście związek za charakterem tej instytucji.

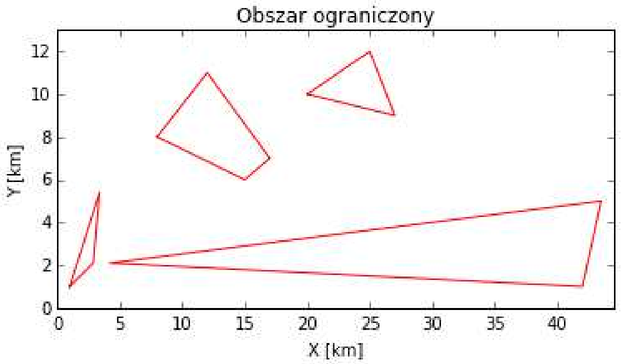
W przypadku obszarów otwartych, na których nie ma żadnych znaczących przeszkód optymalną trajektorią dla pojedynczego obiektu będzie oczywiście linia prosta. Sytuacja komplikuje się kiedy w grę wchodzą inne poruszające się lub nie obiekty. Wówczas problem znalezienia optymalnej trasy sprowadza się do zastosowania odpowiednich reguł. W przypadku transportu morskiego reguły te ujęte są w Międzynarodowym Prawie Drogi Morskiej, które jednoznacznie definiują zasady poruszania się na drogach morskich. Właśnie w tej dziedzinie transportu ma zastosowanie pojęcie obszaru otwartego. Tutja ośrodkiem ruchu jest wielki obszar geograficzny na którym nie ma typowych dla transportu drogowego przeszkód.

Zupełnie inaczej ma się sprawa w przypadku tzw. obszarów ograniczonych (akwenów ograniczonych w transporcie morskim) w których oprócz zwykłych reguł zarządzania ruchem dochodzą takie problemy jak:

- linia brzegowa,
- inne obiekty niebędące statkami,
- głębokość akwenu,
- szerokość toru wodnego,
- i wiele innych…

W [1] autorzy zaproponowali rozwiązanie tego problemu w kontekście tzw. domeny statku [2].

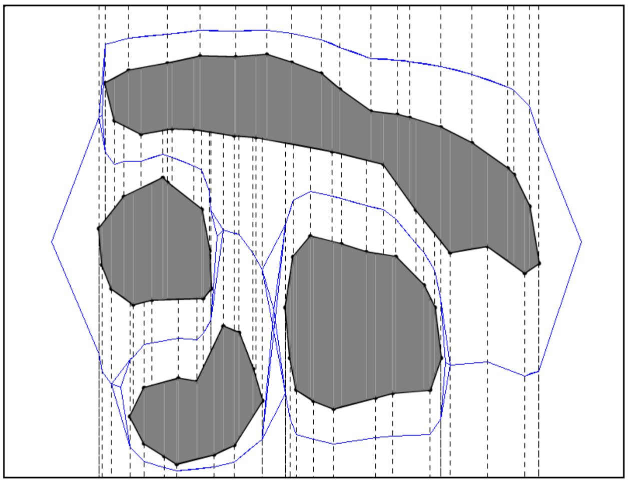
Przykładowy, hipotetyczny obszar ograniczony widoczny jest na Rysunku 1. Wielokąty reprezentują obszary zabronione w obrębie których nie ma możliwości poruszania się.




**Rysunek 1** Przykład obszaru ograniczonego, źródło: opracowanie własne.

## 2. POSZUKIWANIE OPTYMALNEJ TRAJEKTORII

Mając daną cyfrową mapę obszaru należy określić punkty początkowy oraz końcowy. Następnym krokiem jest dyskretyzacja danego obszaru w celu zbudowania grafu wszystkich możliwych przejść w danym obszarze. Wagi krawędzi grafu będą reprezentować odległości pomiędzy dwoma sąsiednimi jego elementami w rozumieniu euklidesowym. Można je również modyfikować uwzględniając inne kryteria jak np. głębokość akwenu, oznakowanie nawigacyjne itp.



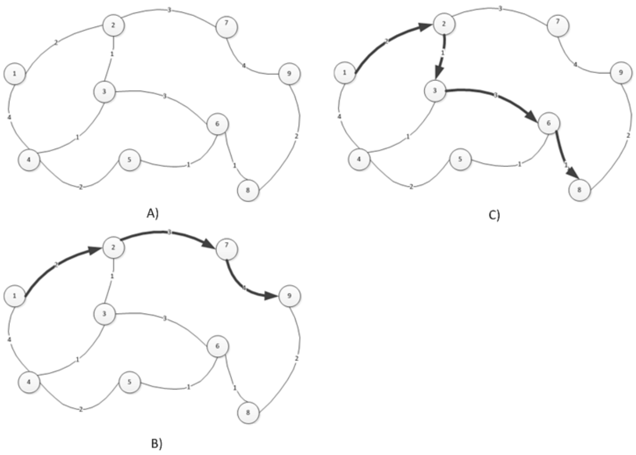
**Rysunek 2** Przykład dyskretyzacji obszaru, źródło: [3].

Rysunek 2 ilustruje przykład dyskretyzacji obszaru ograniczonego z zastosowaniem siatki trapezowej. Ciemne wielokąty reprezentują obszary zabronione. Kolorem niebieskim przedstawiono przykładowe możliwe przejścia pomiędzy poszczególnymi punktami pośrednimi (krawędzie grafu).

Dyskretyzacja obszaru może być realizowana również innymi metodami jak np.:

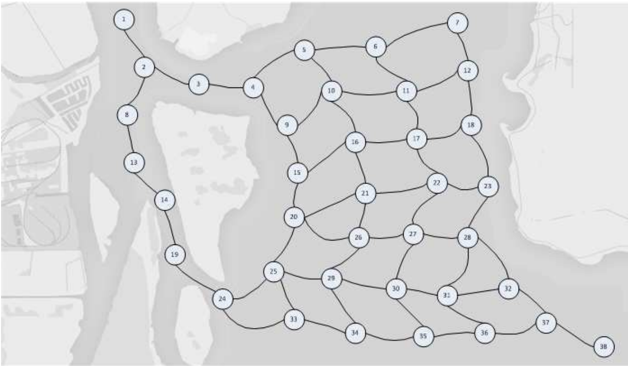
- drzewa czwórkowe,
- triangulacja Delaunaya,
- diagramy Voronoia (przekształcenie dualne do triangulacji Delaunaya),
- i wiele innych.

Po dokonaniu dyskretyzacji mamy wyznaczony graf możliwych przejść. Jego węzły określają punkty przez które powinien poruszać się obiekt. Krawędzie oznaczają trasy pomiędzy nimi.




**Rysunek 3** Przykładowe grafy możliwych tras, źródło: [4].

Powyższy rysunek ilustruje możliwe grafy skierowane (w przypadku grafów nieskierowanych można przyjąć, że ruch obiektu może odbywać się w obu kierunkach) przedstawiające możliwe przemieszczenia obiektu na trasie. Dotarcie z punktu początkowego do punktu końcowego sprowadza się zatem do znalezienia najkrótszej drogi w grafie. Można tu wykorzystać szereg powszechnie dostępnych metod czy zaproponować inne rozwiązania. Tym niemniej najpopularniejszym z nich będzie algorytm Dijkstry lub A*.



**Rysunek 4** Przykładowy graf przejść naniesiony na rzeczywisty obszar ograniczony, źródło: [4]

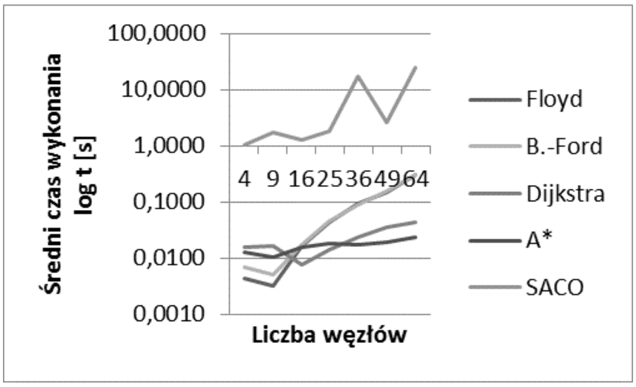
Powyższy rysunek ilustruje przykładowy sposób naniesienia grafu możliwych przejść na konkretny akwen ograniczony. W tym przypadku widzimy wejście do portu w Szczecinie wraz przylegającym jeziorem Dąbie.

## 3. KRYTERIA WYBORU ALGORYTMU POSZUKIWANIA NAJKRÓTSZEJ DROGI

W [5] opisano podstawowe metody poszukiwania najkrótszej drogi w grafie. Ponadto dodano rozwiązanie alternatywne w postaci uproszczonego algorytmu mrówkowego. Testowanie wszystkich metod rozpoczęto od badania złożoności czasowej każdego algorytmu na tzw. grafach nieskończonych. I tak biorąc pod uwagę wycinki grafu nieskończonego, otrzymano następujące wyniki:

**Tabela 1** Przykładowe wyniki w [ms] dla poszczególnych algorytmów poszukiwania najkrótszej drogi, źródło: [5].

| Liczba węzłów | Floyd | B.-Ford | Dijkstra | A* | SACO |
|---|---|---|---|---|---|
| 4 | 0.0043 | 0.0068 | 0.0157 | 0.0130 | 1.0595 |
| 9 | 0.0033 | 0.0050 | 0.0166 | 0.0105 | 1.7452 |
| 16 | 0.0163 | 0.0170 | 0.0076 | 0.0159 | 1.2820 |
| 25 | 0.0444 | 0.0466 | 0.0144 | 0.0179 | 1.8591 |
| 36 | 0.0931 | 0.0872 | 0.0240 | 0.0169 | 16.8955 |
| 49 | 0.1519 | 0.1551 | 0.0349 | 0.0189 | 2.6338 |
| 64 | 0.3013 | 0.3030 | 0.0435 | 0.0232 | 24.5784 |




**Rysunek 5** Porównanie czasów wykonania dla poszczególnych algorytmów, źródło: [5].

Na podstawie porównania powyższych algorytmów można łatwo wywnioskować, że najkrótsze czasy otrzymuje się w przypadku algorytmu A*. Jest to stosunkowo proste rozwiązanie polegające na tym, że poszukuje się minimum funkcji:

$$f(x) = g(x) + h(x)$$

gdzie:

$g(x)$ – koszt drogi od punktu początkowego do punktu pośredniego w którym aktualnie się znajdujemy,

$h(x)$ – koszt drogi od punktu pośredniego do punktu docelowego.

Algorytm ten daje gwarancję optymalności i charakteryzuje się najniższą złożonością obliczeniową, przez co znajduje powszechne zastosowanie w problemach informatyki, transportu i wielu innych. Tym niemniej w tym wypadku konieczna jest znajomość dokładnych współrzędnych geograficznych dla każdego punktu w grafie.

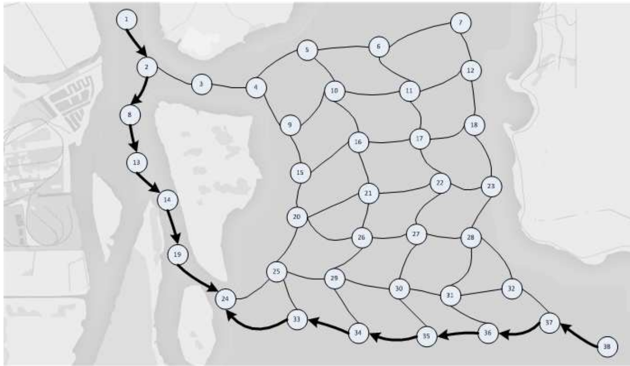
Algorytm Dijkstry [6] jest natomiast najpopularniejszym rozwiązaniem w sytuacji kiedy współrzędne geograficzne nie są znane bądź nie można ich dobrze oszacować. Polega on na kilku krokach:

- określenie punktu początkowego,
- utworzenie macierzy odległości (adjacencji) dla każdego węzła w grafie,
- utworzenie kolejki priorytetowej z uwzględnieniem kryterium odległości,
- dopóki kolejka nie jest pusta:
    - usunięcie węzła o najniższym priorytecie,
    - znalezienie najbliższego sąsiada dla obecnego węzła.
- ostatni rząd w macierzy odległości stanowi wektor zawierający najmniejszą odległość od punktu początkowego do pozostałych punktów w grafie.

Ponadto w przypadku algorytmu Dijkstry możliwe jest poszukiwanie najkrótszej drogi od punktu początkowego do każdego innego punktu w grafie. Zatem w przypadku zmiany punktu docelowego, nie ma potrzeby ponownego uruchamiania całości algorytmu od początku i ten właśnie fakt jest przewagą w odniesieniu do algorytmu A*.

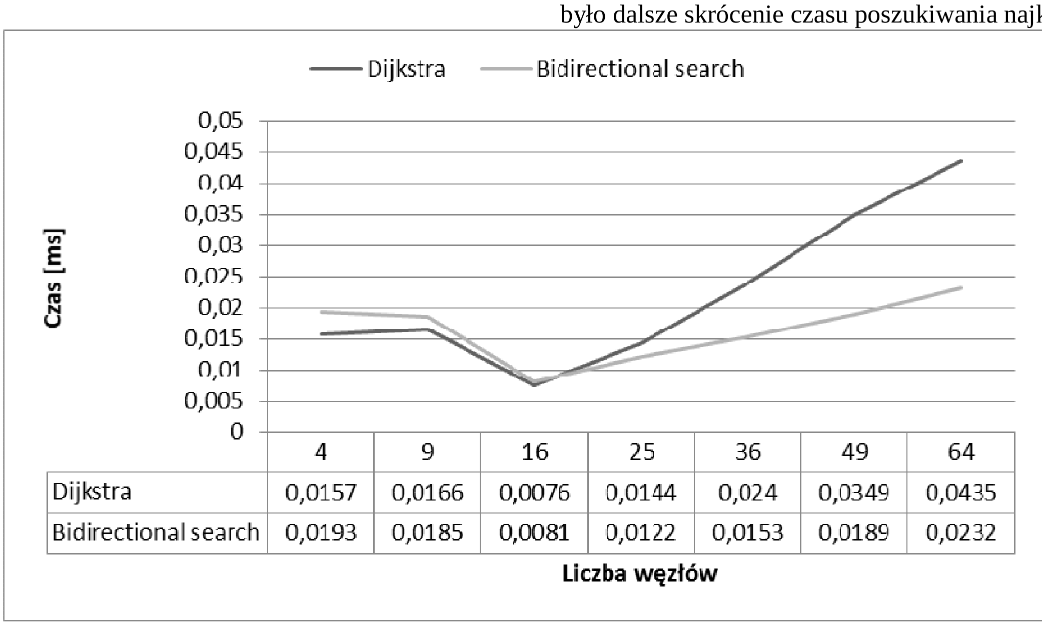
Można jednak poszukiwać dalszych rozwiązań. W [4] zaproponowano podejście polegające na tym, że po zastosowaniu algorytmu Dijkstry, końcowe budowanie optymalnej trasy może się odbywać wielowątkowo.

Rysunek 7 ilustruje porównanie czasów wykonania operacji poszukiwania optymalnej trasy przy dwóch sposobach końcowego jej konstruowania. Jak widać w przypadku przeszukiwania dwukierunkowego doszło do ponownego obniżenia czasu wykonania algorytmu.



**Rysunek 6** Dwukierunkowe poszukiwanie optymalnej trasy, źródło: [4].

Na Rysunku 6 widać ten sam obszar co na Rysunku 4. Strzałkami zaznaczono kierunek budowy trasy z uwzględnieniem wektora odległości otrzymanego po zastosowaniu algorytmu Dijkstry. Efektem tego zabiegu było dalsze skrócenie czasu poszukiwania najkrótszej trasy.




| | 4 | 9 | 16 | 25 | 36 | 49 | 64 |
|---|---|---|---|---|---|---|---|
| Dijkstra | 0,0157 | 0,0166 | 0,0076 | 0,0144 | 0,024 | 0,0349 | 0,0435 |
| Bidirectional search | 0,0193 | 0,0185 | 0,0081 | 0,0122 | 0,0153 | 0,0189 | 0,0232 |

**Rysunek 7** Porównanie przeszukiwania “klasycznego” i dwukierunkowego dla algorytmu Dijkstry.

## 4. PODSUMOWANIE

Niniejszy artykuł prezentuje krótki przegląd sposobów wyznaczania najkrótszej trasy dla obiektów poruszających się po obszarze ograniczonym.
Chociaż w tym przypadku rozważano głównie problemy transportu morskiego, to proponowane metody mogą być z powodzeniem stosowane również w innych dziedzinach jak transport lądowy, systemy wspomagania dla osób głuchoniemych itp.
W przypadku transportu morskiego proponowane podejście może być również narzędziem wspomagającym dla VTS (Vessel Traffic Service). VTS z kolei systemem zarządzania ruchem polegającym na ciągłej obserwacji istniejącej sytuacji nawigacyjnej. Obserwacje te są prowadzone w sposób ciągły zarówno w ośrodkach lądowych jak i morskich.
Najpowszechniej stosowanym algorytmem jest rozwiązanie zaproponowane już ponad 50 lat temu przez holenderskiego matematyka E.W. Dijkstrę. W zasadzie do dziś nie opracowano bardziej wydajnego narzędzia jeśli idzie o poszukiwanie optymalnej drogi w grafie o krawędziach z nieujemnymi wagami.
W przypadku kiedy znamy dokładne współrzędne geograficzne dla każdego punktu docelowego, możemy zastosować algorytm A*, który charakteryzuje się jeszcze mniejszą złożonością obliczeniową.
Długość czasu wykonania procedury wyznaczania najkrótszej drogi (niekoniecznie w sensie geometrycznym) jest bardzo istotna z uwagi na to, że obliczenia te mogą mieć zastosowanie np. w informatycznych systemach wspomagania decyzji dla transportu.

Ciekawym podejściem, w przypadku kiedy mamy do czynienia z sytuacją czysto dynamiczną, może być również zastosowanie algorytmu D*, który wykorzystywany jest np. w zastosowaniach militarnych i kosmicznych (łazik na Marsie). Zakłada się w nim, że środowisko po którym porusza się obiekt jest nieznane, w związku z tym informacja o nim jest niepełna. Warto zwrócić uwagę, że warto wykonać pewne badania w tym kierunku i z pewnością będzie to przedmiotem dalszych publikacji.

Literatura

1. Pietrzykowski Z., Magaj J., „The safe ships trajectory in a restricted area”, Zeszyty Naukowe Akademii Morskiej w Szczecinie nr 39(111) 2014.
2. Goodwyn E.M., „A statistical study of ship domain”, Journal of Navigation 28, 1975, 328-341.
3. Dramski M., Mąka M., “An algorithm of solving collision problem of two objects in restricted area”, Activities of Transport Telematics, Communications in Computer and Information Science vol. 395, 2013, pp 251-257.
4. Dramski M., “Bidirectional search in route planning in naviagtion”, Zeszyty Naukowe Akademii Morskiej w Szczecinie nr 39(111) 2014.
5. Dramski M. ,”Shortest path problem in static navigation situations”, Metody Informatyki Stosowanej nr 5/ 2011
6. Dijkstra, E. W. (1959). "A note on two problems in connexion with graphs". Numerische Mathematik 1: 269–271