ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 1996
Seria: AUTOMATYKA z.119 Nr kol. 1339

Andrzej FORTUNA, Witold SILEIKIS
Politechnika Śląska

ALGORYTM NAWIGACJI ROBOTA MOBILNEGO W PRZESTRZENI 2D

Streszczenie. Tematem referatu jest synteza i analiza algorytmu nawigacji autonomicznego robota mobilnego, poruszającego się w ograniczonej przestrzeni dwuwymiarowej, w której mogą znajdować się proste przeszkody, pomieszczenia, korytarze itp. System sensoryczny robota wykorzystuje dalmierz do pozyskiwania informacji o stanie otoczenia. Przedstawiono algorytm tworzenia lokalnej mapy otoczenia na podstawie informacji sensorycznej pozwalający na rozwiązywanie zadań związanych z nawigacją w tym otoczeniu, wyszukiwaniem bezkolizyjnej trajektorii, oraz sposób syntezy hierarchicznej mapy otoczenia.

ALGORITHM OF MOBILE ROBOT NAVIGATION IN 2D SPACE

Summary. Subject of this paper is synthesis and analysis of navigation algorithm for autonomous mobile robot, which could navigate in two dimensional space, filled with simply obstacles, rooms, corridors etc. Robots sensor system is using range-finder to gain information about environment. There is presented algorithm of building local environment map, which enable solving problems of navigation in this environment, searching for trajectory free from the possibility of collisions and method of synthesis of hierarchical environment map.

1. Wstęp

Autonomiczny Robot Mobilny (ARM) jest robotem potrafiącym samodzielnie poruszać się w otoczeniu o znanym lub nieznanym modelu, umiejącym reagować na dynamiczne zmiany tego otoczenia, posiadającym możliwość omijania przeszkód lokalnych, samodzielnego tworzenia obrazu (modelu wewnętrznego, mapy) otoczenia i rozwiązywania zadań związanych z nawigacją w tym otoczeniu.

Jednym z ważniejszych podsystemów systemu sterowania robotem mobilnym jest podsystem nawigacji. Podsystem nawigacji w celu spełnienia wymagań przedstawionych powyżej powinien umożliwiać :

- wyznaczenie wszystkich podstawowych parametrów określających położenie i orientację ARM w otoczeniu,
- lokalizację obiektów otoczenia oraz ich współrzędnych,

- tworzenie i modyfikowanie obrazu (mapy) otoczenia,
- wyszukiwanie bezkolizyjnej trajektorii w celu realizacji zadań,
- omijanie przeszkód lokalnych.

Schemat blokowy systemu nawigacji oraz opis podstawowych jego elementów przedstawiono w [12].

Realizacja zadania przemieszczenia robota do zadanego położenia końcowego, sformułowana przez system planowania zadań, wymaga wstępnej dekompozycji zadania na zadania cząstkowe i alternatywne. Każde zadanie cząstkowe jest przez system planowania trajektorii lokalnej (na podstawie modelu topologicznego otoczenia) przekazywane do układów wykonawczych robota, które realizują wyznaczoną trajektorię. Tę część systemu można nazwać częścią wykonawczą dla przemieszczeń ARM w otoczeniu.

System sensoryczny robota mobilnego nie jest częścią systemu nawigacji, dostarcza jednak niezmiernie istotnych danych dla tego systemu. Dla systemu nawigacyjnego potrzebne są dwa rodzaje informacji :

1. dane o zbliżaniu się do lokalnych przeszkód, nie będących elementami stałymi otoczenia, w celu zapewnienia możliwości ich ominięcia lub podjęcia przez system sterowania innych decyzji mogących zmienić realizację zadań. W tym celu stosuje się proste czujniki zbliżeniowe,np. ultradźwiękowe [7],[1],
2. dane o położeniu robota w otoczeniu oraz dane o położeniu innych obiektów, niezbędne do lokalizacji robota oraz do tworzenia modelu otoczenia. W projekcie badawczym HILARE[7] wykorzystano do tego celu dalmierz laserowy umieszczony na obrotowej głowicy oraz kamerę wizyjną.

Dane drugiego rodzaju są wykorzystywane przez system tworzenia modelu otoczenia, który na podstawie wstępnie przetworzonych danych systemu sensorycznego buduje i modyfikuje model topologiczny. Model topologiczny jest podstawą syntezy modelu strukturalnego otoczenia. Warunkiem koniecznym poprawności modelu jest wiarygodne określenie współrzędnych robota w otoczeniu, za co odpowiedzialny jest system lokalizacji robota.

W opracowaniu wykorzystywane są pojęcia mapy lokalnej i globalnej (hierarchicznej). Mapa lokalna jest to opis otoczenia ARM w układzie bazowym, uwzględniający elementy topologiczne (ściany, przeszkody, wolne obszary) oraz relacje pomiędzy nimi i ARM. Natomiast mapa globalna stanowi opis otoczenia ARM w układzie bazowym, uwzględniający grupy elementów topologicznych (pomieszczenia, korytarze) oraz relacje pomiędzy nimi.

Celem badań jest sformułowanie algorytmu, który na podstawie informacji z dalmierza oraz systemu nawigacji inercyjnej pozwoli utworzyć mapę lokalną otoczenia ARM.

2. Algorytm budowy mapy lokalnej ARM

Schemat blokowy algorytmu nawigacji i budowy mapy lokalnej przedstawiono na rys.1.

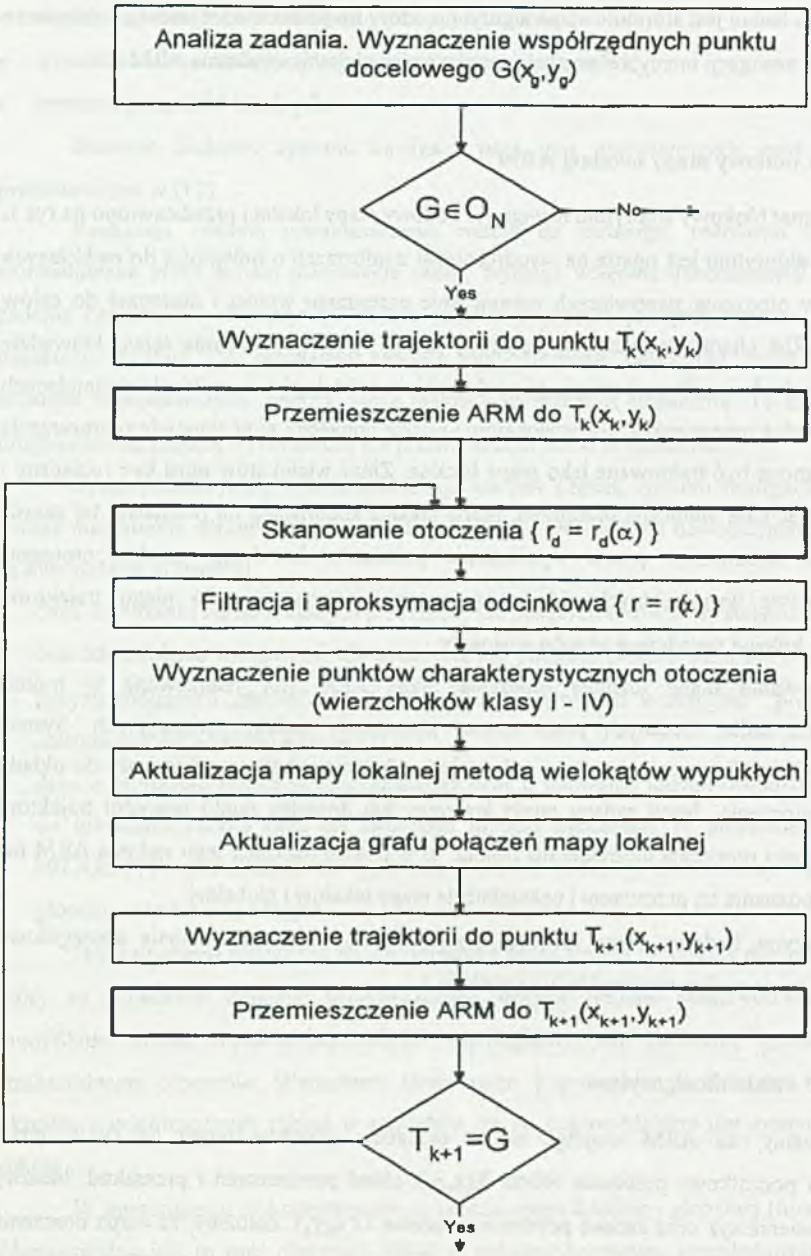
Idea algorytmu jest oparta na wyodrębnieniu z informacji o odległości do najbliższych przedmiotów otoczenia stanowiących ograniczenie przestrzeni wolnej i dostępnej do celów nawigacji ARM charakterystycznych punktów, jak np. punkty załamania ścian, krawędzie przeszkód, i zbudowanie w oparciu o te punkty zbioru wielokątów wypukłych obejmujących widzianą i wolną przestrzeń. Zbiór wielokątów i relacje pomiędzy nimi stanowią reprezentację otoczenia i mogą być traktowane jako mapa lokalna. Zbiór wielokątów musi być rozłączny i musi pokrywać całą widoczną przestrzeń. Mapa lokalna zbudowana na podstawie tej zasady pozwala na stosunkowo proste wyznaczenie położenia dowolnego punktu otoczenia wyrażonego we współrzędnych układu bazowego i wyznaczenie do niego trajektorii, obejmującej kolejne sąsiadujące ze sobą wielokąty.

Budowanie mapy lokalnej otoczenia przez ARM jest realizowane w trakcie wykonywania zadań zlecanych przez system planowania zadań nawigacyjnych. System dekompozycji zadań wyznacza zadane położenie i orientację robota w odniesieniu do układu bazowego otoczenia. Jeżeli zadany punkt końcowy lub dowolny punkt pośredni trajektorii należy do części otoczenia nieznanej dla robota, to w trakcie realizacji tego zadania ARM ma możliwość poznania tej przestrzeni i uaktualnienia mapy lokalnej i globalnej.

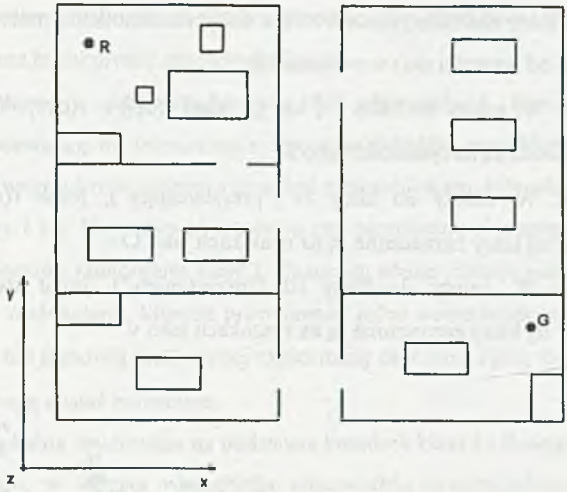
Algorytm budowy mapy lokalnej i wyznaczenia trajektorii zostanie szczegółowo przedstawiony na podstawie przykładu w punkcie 3.

3. Przykład działania algorytmu

Załóżmy, że ARM znajduje się w otoczeniu przedstawionym na rys.2, gdzie zaznaczono początkowe położenie robota $R(x_r,y_r)$, układ pomieszczeń i przeszkód, bazowy układ odniesienia xyz oraz zadane położenie końcowe $G(x_g,y_g)$. Załóżmy, że mapa otoczenia jest pusta, tzn. robot nie zna otoczenia, w którym ma wykonać zadanie. W trakcie jego realizacji robot musi zbudować mapę tej części przestrzeni, przez którą wiodła trajektoria. Współrzędne punktu docelowego zostały wyznaczone przez system planowania i dekompozycji zadań.



Rys.1.Algorytm nawigacji i budowy mapy lokalnej
Fig.1.Navigation algorithm and building local environment map



Rys.2. Schemat przykładowego otoczenia robota
Fig.2. Schema of sample robot environment

Pierwszym krokiem algorytmu jest sprawdzenie, czy punkt docelowy G leży w przestrzeni O_N, czyli w przestrzeni leżącej poza mapą lokalną otoczenia. Zgodnie z powyższymi założeniami warunek ten jest spełniony.

Kolejny punkt algorytmu to wyznaczenie trajektorii do punktu $T_k(x_k,y_k)$, który leży w przestrzeni znanej dla robota i leżącego najbliżej punktu docelowego. Na tym etapie rozwiązania zadania taki punkt nie istnieje i robot w celu rozpoczęcia budowy mapy lokalnej musi wykonać skanowanie otoczenia.

Proces skanowania otoczenia dalmierzem polega na wyznaczeniu odległości do najbliższej przeszkody w pełnym zakresie obrotu głowicy dalmierza od 0 do 360 stopni z zadanym skokiem. Efektem skanowania jest wyznaczenie dyskretnej funkcji $r_d = r_d(\alpha)$, gdzie α przyjmuje dyskretne wartości od 0 do 360, natomiast $r_d(\alpha)$ jest odległością do przeszkody dla określonego położenia głowicy dalmierza α.

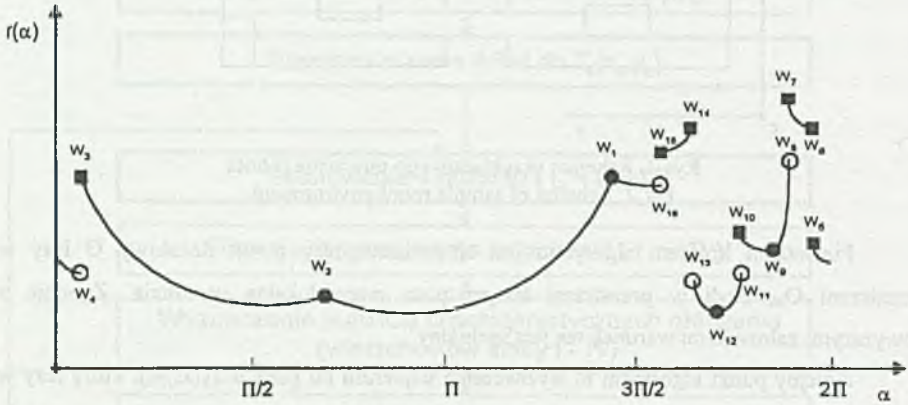
Po odfiltrowaniu szumów i aproksymacji odcinkowej funkcji $r_d = r_d(\alpha)$ metodą najmniejszych kwadratów uzyskuje się funkcję $r = r(\alpha)$ dla ciągłej wartości argumentu α. Wykres tej funkcji jest przedstawiony na rys.3.

Analiza punktów nieciągłości tej funkcji oraz jej pierwszej i drugiej pochodnej pozwala na wyodrębnienie charakterystycznych punktów otoczenia, pokazanych na rys. 4.

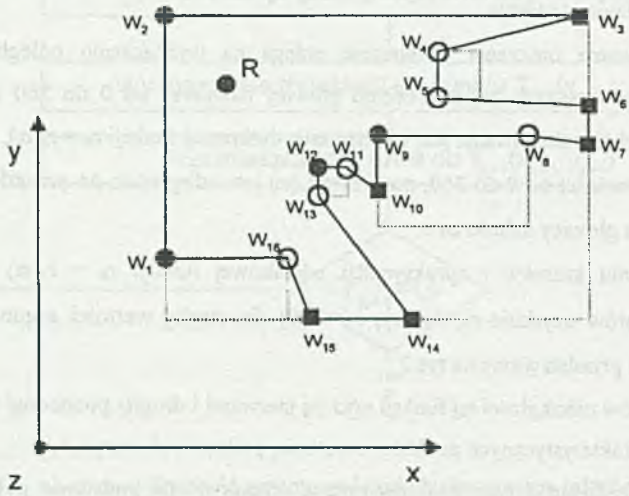
Współrzędne każdego z tych punktów można wyznaczyć na podstawie położenia i orientacji robota w układzie bazowym oraz wartości α i $r(\alpha)$ [12], które są znane.

Wyznaczone punkty charakterystyczne zwane dalej wierzchołkami można podzielić na trzy klasy w zależności od wartości $r(\alpha)$ w otoczeniu α.

- Wierzchołek W_i należy do klasy I ("kąt"), jeżeli $r(\alpha_i-) = r(\alpha_i+)$. Wierzchołki tej klasy zaznaczone są na rysunkach jako λ.
- Wierzchołek W_i należy do klasy II ("przysłaniający"), jeżeli $r(\alpha_i-) < r(\alpha_i+)$. Wierzchołki tej klasy zaznaczone są na rysunkach jako O.
- Wierzchołek W_i należy do klasy III ("przysłonięty"), jeżeli $r(\alpha_i-) > r(\alpha_i+)$. Wierzchołki tej klasy zaznaczone są na rysunkach jako v.



Rys.3. Wykres funkcji $r = r(\alpha)$ dla punktu R z rys.2
Fig.3. Graph of the function $r = r(\alpha)$ for point R from fig.2



Rys.4. Analiza punktów charakterystycznych skanowania
Fig.4. Analysis of scaning characteristic points

Połączenie kolejnych punktów pozwala na określenie obszaru widocznego i niezajętego przez jakiekolwiek przeszkody (rys.4).

Wyznaczone i przyporządkowane do odpowiednich klas wierzchołki obszaru widocznego pozwalają na wyznaczenie zbioru wielokątów wypukłych, pokrywających ten obszar. W pierwszym kroku należy wyznaczyć takie wielokąty, których wszystkie wierzchołki należą do klasy I lub II, ponieważ stanowią one niezmienne elementy mapy lokalnej. Takie wielokąty nazywamy komórkami klasy I. Pozostały obszar zostaje pokryty komórkami klasy II, tzn. takimi wielokątami, których przynajmniej jeden wierzchołek należy do klasy III. Tej klasy komórki nie stanowią niezmiennej części mapy otoczenia i przy innym ustawieniu robota w otoczeniu mogą zostać zmienione.

Mapę lokalną zbudowaną na podstawie komórek klasy I i II wygodnie jest przedstawić w postaci grafu, w którym wierzchołki odpowiadają poszczególnym komórkom, a łuki odpowiadają możliwości przemieszczenia się robota pomiędzy tymi komórkami. W każdej komórce można wyróżnić dowolnie wybrany punkt (np. środek ciężkości) w celu wyznaczania odległości pomiędzy nimi.

Mapę lokalną w postaci zbioru komórek klasy I i II oraz odpowiadającego im grafu połączeń dla skanowania z punktu R przedstawia rys.5.

Węzły grafu oznaczone jako λ odpowiadają komórkom klasy I, natomiast węzły oznaczone jako μ odpowiadają komórkom klasy II.

Po zakończeniu tego kroku algorytmu możliwe jest wyznaczenie trajektorii zbliżającej robota do punktu docelowego. Realizowane jest to za pomocą algorytmu A*, pozwalającego znaleźć ścieżkę w grafie minimalizującą funkcję kosztów

$$F = F_Z + F_N$$

gdzie:

$$F_Z = \sum_{i=1}^{i=k-1} d_{i,i+1}$$ - długość trajektorii, obejmującej komórki kl. I i II

$$F_N = C \cdot \sqrt{(x_k - x_g)^2 + (y_k - y_g)^2}$$ - najkrótsza odległość od ostatniego znanego punktu trajektorii do punktu końcowego (wg miary Euklidesowej)

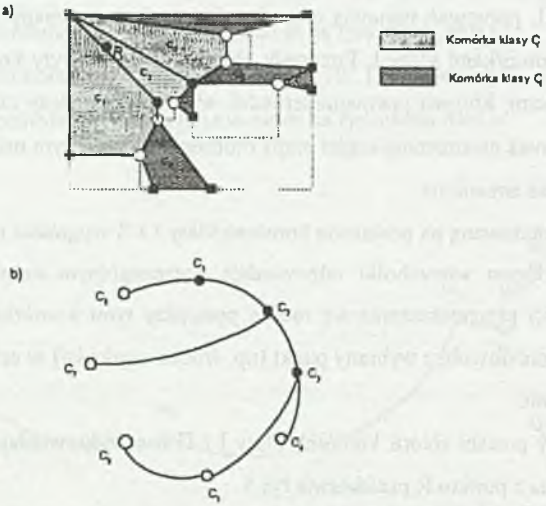
lub

$$F_N = C \cdot (|y_k - y_g| + |x_k - x_g|)$$ - jw. (wg miary nowojorskiej).

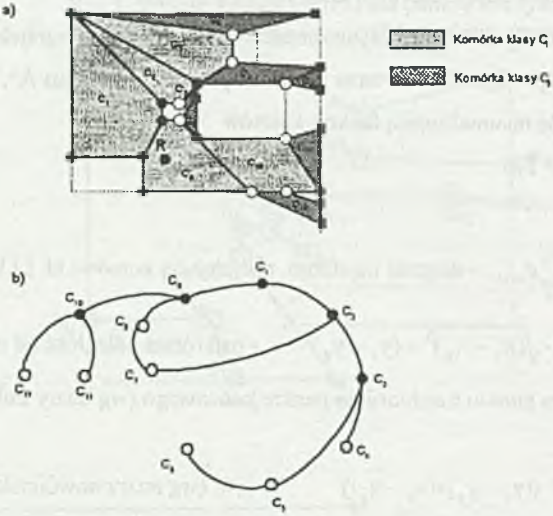
W wyniku analizy mapy lokalnej i zdefiniowanej funkcji kosztów kolejnym krokiem algorytmu w tym przykładzie jest przemieszczenie robora do komórki C_8. Ponieważ cel nie został osiągnięty, a mapa lokalna nie zawiera informacji o otoczeniu z komórki C_8 w kierunku

punktu docelowego proces poznawczy jest realizowany ponownie począwszy od skanowania otoczenia z nowego położenia.

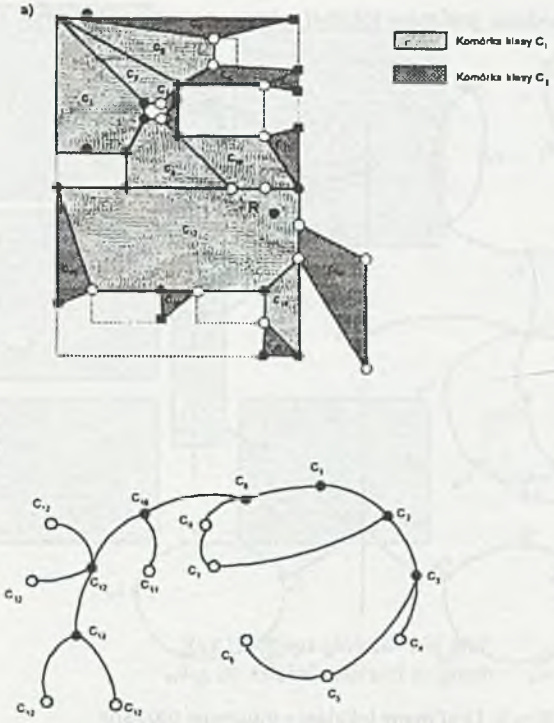
Kolejne rysunki 6 i 7 przedstawiają kolejne fazy budowy mapy lokalnej i przemieszczenia robota do punktu docelowego, natomiast rys.8 przedstawia pełny obraz otoczenia robota, który został zsyntezowany w wyniku realizacji wielu zadań nawigacyjnych.



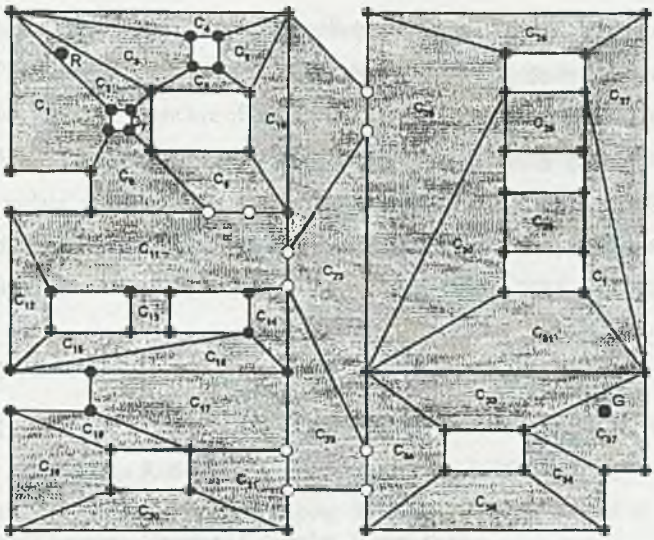
Rys.5. Mapa lokalna otoczenia -faza 1
Fig.5. Local environment map - phase 1



Rys.6. Budowa mapy lokalnej - faza 2
Fig.6. Local map building - phase 2

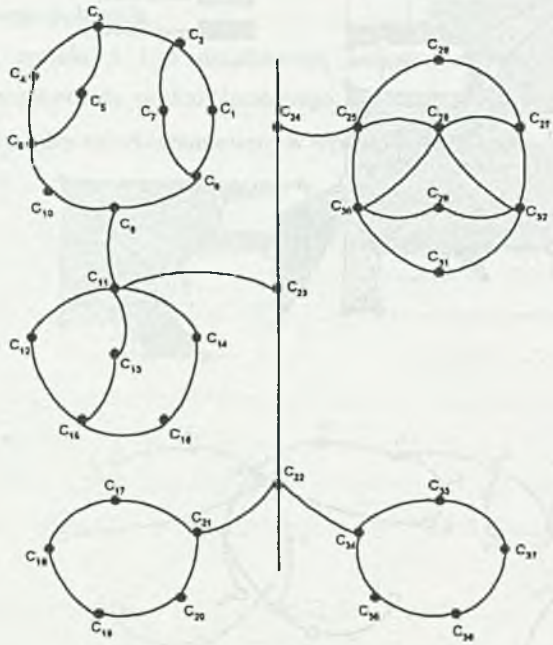


Rys.7. Budowa mapy lokalnej - faza 3
Fig.7. Local map building - phase 3



Rys.8. Budowa mapy lokalnej - otoczenie poznane
Fig.8. Local map building - known environment

Na rys.9 przedstawiono graf mapy lokalnej całego poznanego otoczenia.



Rys.9. Graf mapy lokalnej - otoczenie poznane
Fig.9. Graph of local map - known environment

Tak wyznaczony graf mapy lokalnej dla otoczenia całkowicie poznanego przez robota pozwala na syntezę mapy globalnej za pomocą agregacji części grafu, w których występuje zamknięta pętla. Ideę takiej agregacji przedstawia rys.10.

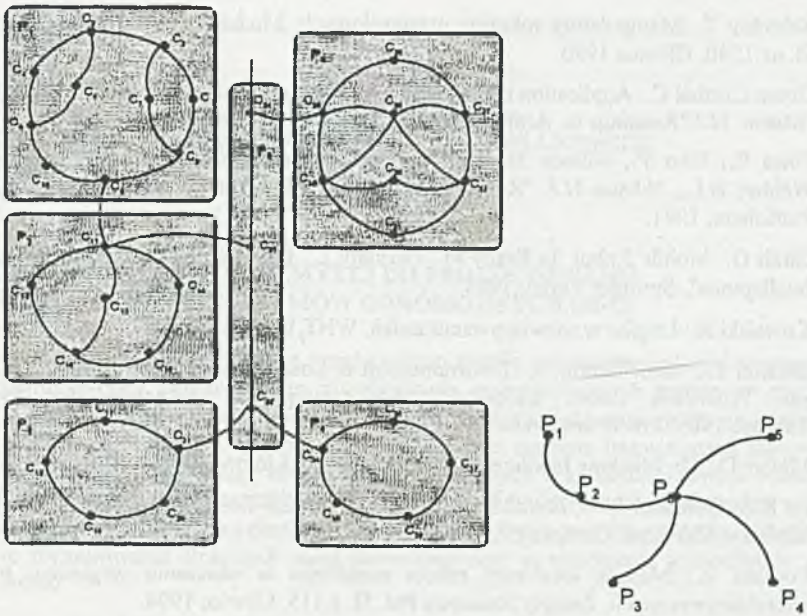
Synteza mapy globalnej pozwala na znaczne uproszczenie wyznaczania trajektorii realizującej określone zadanie nawigacyjne.

Dla przykładu prezentowanego powyżej analiza mapy globalnej pozwala stwierdzić, że

$R(x_r,y_r) \in P_1$

$G(x_g,y_g) \in P_4$

Wynika z tego na podstawie grafu mapy globalnej trajektoria P1 $\Rightarrow$ P2 $\Rightarrow$ P6 $\Rightarrow$ P4, która da się prosto zdekomponować na zadania proste, realizowane na podstawie mapy lokalnej.



Rys.10. Mapa globalna i jej graf
Fig.10. Global map and its graph

4. Wnioski

Przedstawiony w pracy algorytm pozwala na zbudowanie mapy lokalnej otoczenia przez robota mobilnego, wyposażonego w dalmierz i wykonującego zadania nawigacyjne w tym otoczeniu. Otoczenie może być całkowicie nieznane dla robota. Na podstawie informacji z dalmierza i systemu nawigacji inercyjnej względem układu bazowego robot może utworzyć własny obraz świata, w którym pracuje.

Synteza mapy globalnej otoczenia pozwala na znaczne uproszczenie rozwiązywania zadań nawigacyjnych, ale nastręcza duże kłopoty w określeniu reguł agregacji grafu mapy lokalnej.

LITERATURA

1. Cox I.J.: Autonomous Robot Vehicles. Springer-Verlag, 1990.
2. Borkowski A. et in.: Grid-based mapping for autonomous mobile robot. Robotics and Autonomous Systems, Volume 11, Number 1, May 1993.
3. Craig I.I.: Introduction to Robotics. Mechanics and Control. Addison-Wessley Publ., 1986.

4. Szkodny T.: Manipulatory robotów przemysłowych. Modele matematyczne. Skrypt Pol. Śl. nr 1530, Gliwice 1990.

5. Green Cordell C.: Application of Theorem Proving to Problem Solving. In Webber B.L., Nilsson N.J."Readings in Artificial Intelligence", Morgan Kaufmann Publishers, 1981.

6. Fikes R., Hart P., Nilsson N.: Learning and Executing Generalized Robot Plans. In Webber B.L., Nilsson N.J. "Readings in Artificial Intelligence", Morgan Kaufmann Publishers, 1981.

7. Giralt G.: Mobile Robot. In Brady M., Gerhardt L., Davidson H. "Robotics and Artificial Intelligence", Springer Verlag,1984.

8. Kowalski R.: Logika w rozwiązywaniu zadań. WNT, Warszawa 1989.

9. Boissier L.: Modelisation de l'Environnement et Localisation du Robot Mobile HILARE par Telemetrie Laser. Laboratoire d'Automatique et d'Analyse des Systemes, Toulose 1985, Praca doktorska.

10. Michie D.: On Machine Intelligence. Ellis Horwood Limited, 1986.

11. Fu K.S, Gonzales R.C., Lee C.S.G.: Robotics: Control, Sensing, Vision and Intelligence. McGraw-Hill Book Company, 1989.

12. Fortuna A.: Metoda lokalizacji robota mobilnego w otoczeniu względem punktów charakterystycznych. Zeszyty Naukowe Pol. Śl. z.115, Gliwice 1994.

13. Fortuna A.: Algorytm nawigacji robota mobilnego w przestrzeni 2D. Praca nie publikowana, zrealizowana w ramach badań własnych, Gliwice 1995.

Recenzent: Prof.dr hab.inż. Antoni Woźniak

Wpłynęło do Redakcji do 30.06.1996 r.

Abstract

Subject of this paper is synthesis and analysis of navigation algorithm for autonomous mobile robot, which could navigate in two dimensional space, filled with simply obstacles, rooms, corridors etc. Robots sensor system is using range-finder to gain information about environment. There is presented algorithm of building local environment map, which enable solving problems of navigation in this environment, searching for trajectory free from the possibility of collisions and method of synthesis of hierarchical environment map.

The idea of presented algorithm is based on separating characteristic points, like walls, edges of obstacles which enclose free space to navigation, from information about distance to nearest elements of environment and defining set of convex polygons, which cover this free space. This set of convex polygons and relations between them determine representation of the environment and is treated as local map. The set of polygons must be disjoint and must cover all visible space. The local map, constructed this way, allow to calculate coordinates of any point in base coordinate system and calculate trajectory composed of consecutive and adjacent polygons.

The local map is constructed during execution of tasks determined by planning system. The task decomposition system determine position and orientation of robot in base coordinate system. If goal or any point of trajectory belongs to unknown part of environment, robot during execution this trajectory is able to recognize this space and modernize the local map.

The algorithm of local map constructing and determining trajectory to goal is presented in details in point 3 of this paper.