ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 2002
Seria: AUTOMATYKA z. 136 Nr kol. 1556

Krzysztof SKRZYPCZYK, Adam GAŁUSZKA
Politechnika Śląska

KONCEPCJA HYBRYDOWEGO SYSTEMU STEROWANIA ZESPOŁEM AUTONOMICZNYCH POJAZDÓW CZYSZCZĄCYCH

Streszczenie. W prezentowanej pracy przedstawiono koncepcję hybrydowego systemu sterowania grupą robotów mobilnych działających w środowisku pomieszczeń biurowych. Celem działania grupy pojazdów jest oczyszczenie podłogi zadanej strefy biura oraz powrót pojazdów do swoich miejsc parkowania. Do zrealizowania tych zadań zaproponowano dwupoziomową hybrydową strukturę systemu sterowania. Warstwa, zwana nadrzędną, ma za zadanie zaplanowanie efektywnego czyszczenia sektorów, na które podzielone zostały pomieszczenia biurowe, bazując na informacji o rozkładzie pomieszczeń oraz pozycji poszczególnych pojazdów. Druga z warstw systemu jest odpowiedzialna za realizację poszczególnych zadań zlecanych przez warstwę nadrzędną. Ta część systemu steruje w sposób bezkolizyjny ruchem danego pojazdu czyszczącego zgodnie z zaimplementowaną strategią. Do realizacji ruchu robotów użyto koncepcji sterowania typu behawioralnego.

A CONCEPTION OF A HYBRID SYSTEM FOR CONTROLING A TEAM OF AUTONOMOUS FLOOR-SWEEPERS

Summary. A conception of a hybrid system for coordination and controlling a team of autonomous floor sweepers is presented in the paper. The goal of the team of vehicles is to clean up a floor of a given office sector and to return to their initial positions. A two-layer hybrid control architecture was proposed in the paper. The task of the upper layer is to plan in an effective way a process of cleaning sectors by robots. Robots task planning problem has been modeled using STRIPS representation. The lower layer is responsible for executing a robot collision free movement using the given strategy. Behavior-based model is used for synthesis of the lower layer of the control system.

1. Wprowadzenie

Jednym z podstawowych wymagań stawianych systemom sterowania robotów (pojazdów) mobilnych jest ich autonomiczność, czyli zdolność do działania robota bez udziału operatora (człowieka) w środowisku o nieznanym modelu oraz do reagowania na dynamiczne zmiany tego otoczenia [2]. Tak postawione wymagania implikują wiele problemów, z którymi musi poradzić sobie system sterowania. Do najważniejszych z nich należą: planowanie działań, generowanie bezkolizyjnej trajektorii ruchu robota, konieczność działania na podstawie niepewnej i niekompletnej informacji, budowa modelu otoczenia itp. [1,2,4,6,8,9].

Operowanie przez robot w rzeczywistym środowisku o złożonej strukturze, takim jak np. duże biuro, bez jakiejkolwiek informacji o tymże otoczeniu jest zadaniem bardzo skomplikowanym. Dlatego w praktyce często, wprowadza się do systemu wstępną informację o rozkładzie pomieszczeń, co umożliwia łatwe zdekomponowanie globalnego zadania stawianego robotowi [8].

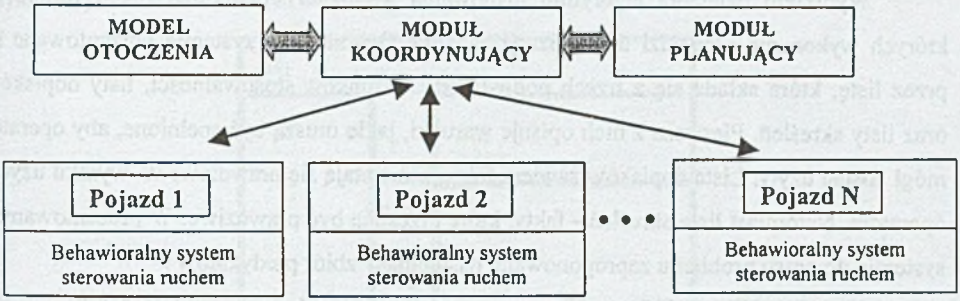
Pole zastosowań robotów (pojazdów) będących autonomicznymi przynajmniej w zakresie zdefiniowanym powyżej jest bardzo szerokie. Człowiek stara się konstruować urządzenia, które mogłyby go zastąpić przy wykonywaniu różnorodnych prac (zadań), począwszy od transportu materiałów produkcyjnych, poprzez rozbrajanie min a na eksploracji innych planet skończywszy. Problemy modelowania i sterowania w środowisku wieloagentowym są ostatnio często poruszane w literaturze [3,4,5,6].

Jednym z zastosowań pojazdów mobilnych, popularnym w ostatnich czasach, jest wykorzystanie ich do czyszczenia podłóg dużych powierzchni hal fabrycznych, biur itp.

W prezentowanej pracy przedstawiono zarys koncepcji hybrydowego systemu sterowania grupą robotów mobilnych (urządzeń czyszczących) działających w środowisku pomieszczeń biurowych. Celem działania grupy pojazdów jest wyczyszczenie zadanej strefy biura oraz powrót pojazdów do swoich miejsc parkowania. System sterowania powinien zaplanować oraz koordynować działania poszczególnych robotów, jak również zapewnić ich bezkolizyjne poruszanie się po otoczeniu o dynamicznej strukturze.

2. Opis systemu

Do zrealizowania zadań stawianych grupie robotów zaproponowano dwupoziomową hybrydową strukturę systemu sterowania (rys.1). Pierwsza warstwa, zwana nadrzędną, ma za zadanie zaplanowanie efektywnego czyszczenia sektorów, na które podzielono pomieszczenia biurowe, bazując na informacji o rozkładzie pomieszczeń oraz pozycji poszczególnych pojazdów. Model otoczenia zdefiniowany jest na dwóch poziomach: topologicznym i geometrycznym. Pierwszy z nich ma postać listy predykatów opisujących wzajemne położenie pomieszczeń, natomiast drugi dostarcza informacji o ich wymiarach i położeniu względem pewnego stałego układu odniesienia.



Rys. 1. Schemat architektury systemu sterowania
Fig. 1. A diagram of the control system architecture

Moduł planujący generuje listę operatorów stanowiących plan działań, który za pośrednictwem *modułu koordynującego* jest dzielony i przesyłany z użyciem komunikacji radiowej do poszczególnych pojazdów. Każdy z pojazdów dostarcza tą samą drogą informacje o swoim położeniu i stanie realizacji przesłanego zadania: czyszczenie w toku, czyszczenie wykonane, niemożność wykonania zadania. W ostatnim z wymienionych przypadków dokonywane jest przeformułowanie planu i ponowne rozdzielenie go pomiędzy poszczególne jednostki.

Druga z warstw systemu jest odpowiedzialna za realizację poszczególnych zadań zlecanych przez warstwę nadrzędną. Ta część systemu steruje w sposób bezkolizyjny ruchem danego pojazdu czyszczącego zgodnie z zaimplementowaną strategią. Do realizacji ruchu robotów zaproponowano użycie koncepcji sterowania typu behawioralnego.

2.1. Moduł planujący

Do zamodelowania problemu planowania działań poszczególnych robotów wykorzystano reprezentację STRIPS [7]. Zadanie w systemie STRIPS jest reprezentowane przez trójkę (ϑ, Σ, Ω), gdzie:

ϑ - zbiór operatorów (modeli działań robota),

Σ - opis początkowej sytuacji problemowej,

Ω - opis końcowej sytuacji problemowej.

Opis początkowej, końcowej (celu) oraz bieżącej (stanów pośrednich) sytuacji problemowej jest zrealizowany za pomocą listy predykatów, opisujących obiekty występujące w zadaniu planowania oraz związki między nimi.

Wynikiem działania algorytmu planowania jest zbiór operatorów (akcji robotów), których wykonanie prowadzi do realizacji zadania. Operatory w systemie prezentowane są przez listę, która składa się z trzech podlist: listy warunków stosowalności, listy dopisków oraz listy skreśleń. Pierwsza z nich opisuje warunki, jakie muszą być spełnione, aby operator mógł zostać użyty. Lista dopisków zawiera fakty, które stają się prawdziwe w wyniku użycia operatora. Natomiast lista skreśleń - fakty, które przestają być prawdziwe. W prezentowanym systemie do opisu problemu zaproponowano następujący zbiór predykatów :

Connected(SectorXSectorY) – oznacza, że istnieje połączenie sektora X z Y;

RobotInSector(RobotNo, SectorNo) – oznacza, że robot o numerze *No* znajduje się w sektorze *SectorNo* ;

SectorFree(SectorNo) – oznacza, że w sektorze *SectorNo* nie znajduje się żaden robot;

SectorClean(SectorNo) – oznacza, że sektor *SectorNo* jest czysty (został wyczyszczony);

Działania robotów zostały zamodelowane za pomocą dwóch operatorów:

MoveToSector(RobotNo, SectorF, SectorT) – oznacza operację przemieszczenia robota *RobotNo* z sektora *SectorF* do sektora *SectorT*;

Warunki stosowalności: Connected(SectorF,SectorT), RobotInSector(RobotNo,SectorF), SectorFree(SectorT)

Lista dopisków: RobotInSector(RobotNo,SectorT), SectorFree(SectorF)

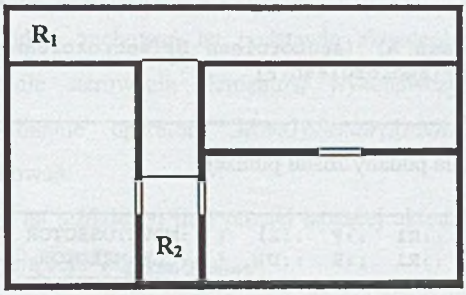
Lista skreśleń: RobotInSector(RobotNo,SectorF), SectorFree(SectorT)

CleanSector(RobotNo, SectorNo) – oznacza operację czyszczenia sektora *SectorNo*, przez robot *RobotNo*;

Warunki stosowalności: RobotInSector(RobotNo,SectorNo), not SectorClean(SectorNo),

Lista dopisków: SectorClean(SectorNo)

W celu zilustrowania procesu planowania wykonywanego przez moduł planujący, rozpatrzmy następujący przykład. Dane jest pomieszczenie biurowe, które zostało podzielone na 7 sektorów w sposób pokazany na rys. 2. W stanie początkowym wszystkie sektory zostały oznaczone jako nie wyczyszczone. Miejsca parkowania robotów R1 i R2 są zlokalizowane odpowiednio w sektorach F i D. Celem robotów jest oczyszczenie wszystkich sektorów oraz powrót do miejsc parkowania w taki sposób, aby roboty nie spotkały się (nie znalazły się w tym samym czasie w jednym sektorze).



Rys. 2. Podział pomieszczeń na sektory i początkowe rozmieszczenie robotów

Fig. 2. Sectors layout and initial robots location

Problem planowania STRIPS został zamodelowany w języku PDDL [11] (Planing Domain Definition Language). Rozwiązanie uzyskano wykorzystując implementację uniwersalnego systemu planującego Graphplan [10]. Wadą takiego podejścia jest mała efektywność (w sensie czasu) poszukiwania rozwiązania problemu, co związane jest z uniezależnieniem algorytmu planującego od dziedziny rozważanego problemu. Dla problemu poruszanego w artykule korzystniej byłoby zaprojektować system planujący jedynie dla rozważanej dziedziny, jednak klasa dziedziny nie została do końca jeszcze sprecyzowana. Sytuacja początkowa została zamodelowana w sposób następujący:

```
(:init
        (connected F E)
        (connected E F)
        (connected E D)
        (connected D E)
        (connected A D)
        (connected D A)
        (connected D C)
        (connected C D)
        (connected C B)
        (connected B C)
        (robotinsector R1 F)
        (robotinsector R2 D)
        (sectorfree A)(sectorfree B)(sectorfree C)(sectorfreeE)
)
```

Natomiast cel:

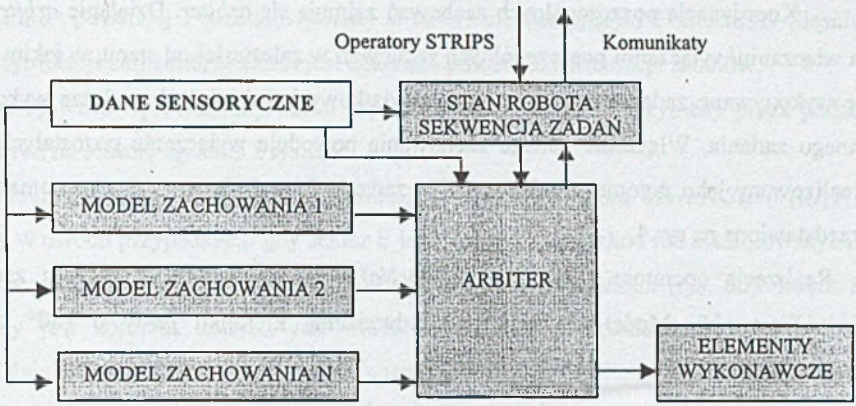
```
(:goal
        (and (sectorclean A) (sectorclean B)(sectorclean C)(sectorclean D)
        (sectorclean E)(sectorclean F)
        )
```

Wynik operacji planowania podany został poniżej:

```
((( ::MOVETOSECTOR  ::R1  ::F  ::E)  ( ::MOVETOSECTOR  ::R2  ::D  ::C))
 (( ::MOVETOSECTOR  ::R1  ::E  ::D)  ( ::CLEANSECTOR  ::R2  ::C))
 (( ::MOVETOSECTOR  ::R2  ::C  ::B)  ( ::MOVETOSECTOR  ::R1  ::D  ::A))
 (( ::CLEANSECTOR  ::R1  ::A) ( ::CLEANSECTOR  ::R2  ::B)))
```

3. Moduł nawigacji i wykonania zadań

Ta część systemu odpowiedzialna jest za realizacje zadań (operatorów) zlecanych poszczególnym jednostkom (robotom) przez moduł planujący. Cechą charakterystyczną tego modułu jest to, iż implementowany on jest na poszczególnych jednostkach i komunikuje się z nadrzędnym modułem planującym drogą radiową. Do jego realizacji zastosowano behawioralną metodę sterowania [6]. Ogólna struktura tego modułu została przedstawiona na rys. 3.



Rys. 3. Ogólny schemat blokowy modułu wykonywania zadań

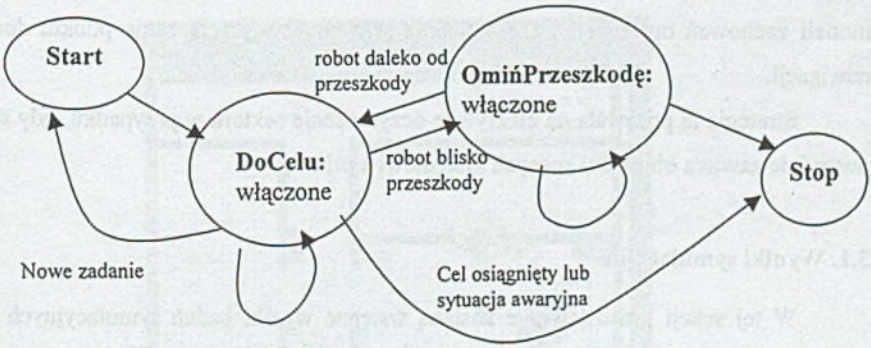
Fig. 3. General structure of the task execution module

Poszczególne modele zachowań na podstawie aktualnej informacji sensorycznej wypracowują odpowiednie sterowania elementów wykonawczych (silników robota). Przykładowo, na wykonanie operatora *MoveToSector(RobotNo,SecNo)* składają się następujące modele zachowań:

DoCelu – polega na jeździe po linii prostej łączącej aktualną pozycję środka robota $(x_{s,robotno}, y_{s,robotno})$ z pozycją celu $(x_{g,secno}, y_{g,secno})$.

OmińPrzeszkodę – powoduje zmianę kierunku jazdy na lewo (prawo) od wykrytej przeszkody.

Stop – powoduje zatrzymanie robota.



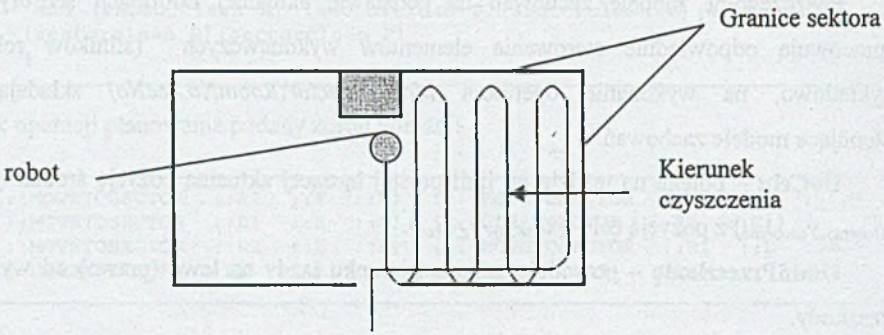
Rys. 4. Poszczególne stany modułu arbitrującego

Fig. 4. States of the arbiter

Koordynacją poszczególnych zachowań zajmuje się *arbiter*. Działanie *arbitra* polega na włączaniu/wyłączaniu poszczególnych zachowań w zależności od stanu, w jakim znajduje się wykonywane zadanie oraz od sytuacji wyjątkowych zaistniałych podczas wykonywania danego zadania. Włączenie danego zachowania powoduje wyłączenie pozostałych. *Arbiter* zrealizowany jako automat skończony. Dla zadania *MoveToSector* postać automatu została przedstawiona na rys.4.

Realizacja operatora *CleanSector(SectorNo)* wymaga dołączenia modelu zachowania *ZmieńKierunek()*. Model ten powoduje odwrócenie kierunku jazdy o 180° w dwóch przypadkach:

- gdy przeszkoda zostanie wykryta z przodu robota,
- gdy osiągnięty zostanie kraniec sektora równoległy do wektora kierunku czyszczenia (rys.5).



Rys. 5. Strategia czyszczenia sektora

Fig. 5. A sector cleaning strategy

Efekt czyszczenia sektora przez robot otrzymuje się przez połączenie wszystkich modeli zachowań opisanych wcześniej oraz sekwencyjne przełączanie punktu docelowego nawigacji.

Strategia ta pozawala na efektywne oczyszczenie sektora w przypadku, gdy sektor jest pusty (nie zawiera obiektów znanych systemowi a priori).

3.1. Wyniki symulacyjne

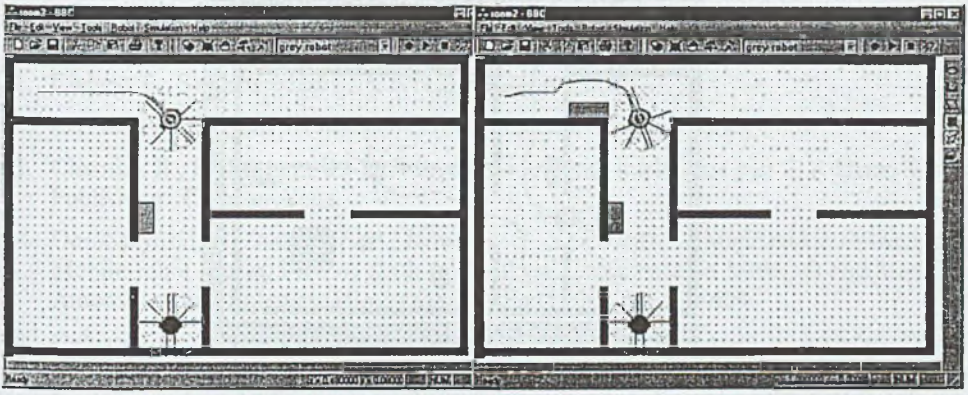
W tej sekcji przedstawione zostaną wstępne wyniki badań symulacyjnych działania części *modułu nawigacji i wykonywania zadań* odpowiedzialnej za realizację zadania (operatora) *MoveToSector*. Prezentowane wyniki otrzymano za pomocą aplikacji

BBCSimulator powstałej i rozbudowywanej w Instytucie Automatyki Politechniki Śląskiej. Podczas symulacji założono, iż znane jest dokładne położenie i orientacja robotów.

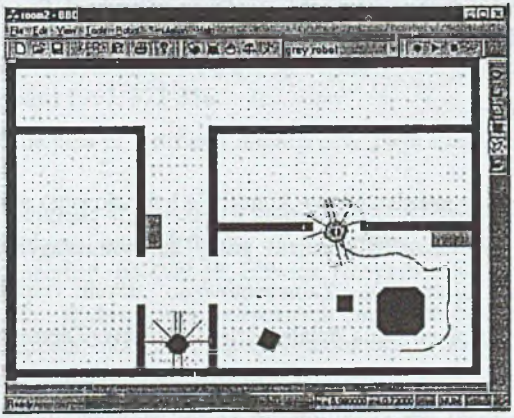
Do systemu wprowadzony został a priori model otoczenia otrzymany przez podział pomieszczeń na sektory zgodnie z rys.2.

Rysunek 6 przedstawia wynik symulacji działania operatora *MoveToSector(R1,F,E)* uzyskany w dwóch przypadkach: gdy sektor E jest wolny od przeszkód nie zdefiniowanych w systemie (rys. 6a) oraz gdy w tym sektorze pojawia się przeszkoda (rys. 6b). Robot *R1* oznaczony jest kolorem szarym. Jak widać, w obydwu przypadkach moduł nawigacji wykonał działanie operatora, czego skutkiem jest bezkolizyjny przejazd robota do sektora E.

Podobnie na rys.7 pokazano symulację działania operatora *MoveToSector(R1,C,B)*. W tym przypadku również moduł nawigacji zrealizował działanie operatora, pomimo że na drodze robota umieszczono kilka przeszkód.



Rys. 6. Wynik symulacji działania operatora *MoveToSector(R1,F,E)*
Fig. 6. Result of simulation of the operator *MoveToSector(R1,F,E)*

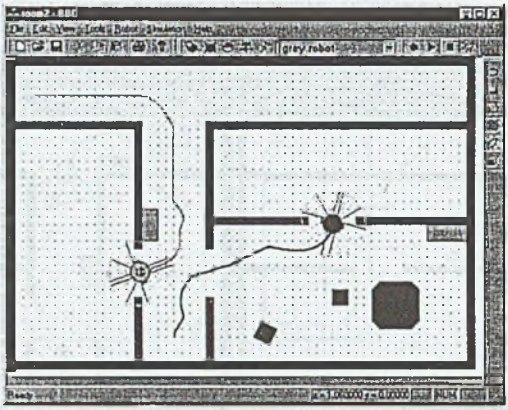


Rys. 7. Wynik symulacji działania operatora *MoveToSector(R1,C,B)*
Fig. 7. Result of the simulation of the operator *MoveToSector(R1,C,B)*

Kolejnym krokiem było sprawdzenie działania systemu w środowisku wielu robotów. W tym celu zasymulowano zadanie opisane następującą listą operatorów:

```
(((  ::MOVETOSECTOR  ::R1  ::F  ::E)  (  ::MOVETOSECTOR  ::R2  ::D  ::C))
 (  ::MOVETOSECTOR  ::R1  ::E  ::D)
 (  ::MOVETOSECTOR  ::R1  ::D  ::A)  (  ::MOVETOSECTOR  ::R2  ::C  ::B))
```

Rezultat eksperymentu przedstawiono na rys.8. Robot R1 oznaczony został kolorem, szarym natomiast R2 kolorem czarnym. W wyniku wykonania sekwencji operatorów przez moduły nawigacji poszczególnych robotów zrealizowany został bezkolizyjny przejazd robota R1 z sektora F do A oraz przejazd robota R2 z sektora D do B. Tym samym otrzymano rozwiązanie postawionego zadania.



Rys. 8. Wynik symulacji działania operatora sekwencji operatorów

Fig. 8. Result of simulation of operators sequence

4. Podsumowanie

W niniejszej pracy przedstawiono koncepcję hybrydowego systemu sterowania, która może posłużyć np. do sterowania ruchem grupy autonomicznych robotów czyszczących podłogi powierzchni biurowych. Zaprezentowano użycie reprezentacji STRIPS do modelowania problemu planowania działań grupy robotów. Zaproponowano wykorzystanie behawioralnych metod sterowania do realizacji bezkolizyjnego ruchu poszczególnych robotów wchodzących w skład zespołu. Pokazano wyniki symulacji działania modułu

planującego. Ponadto zamieszczono symulacje działania przykładowych operatorów, do których realizacji użyto koncepcji sterowania behawioralnego.

5. Kierunki dalszych prac

W dalszej kolejności planuje się rozszerzenie modułu planującego o możliwość działania na bazie niepewnej informacji. Ponadto przewidywane jest dalsze, symulacyjne przebadanie pozostałych operatorów.

Niniejsza praca finansowana jest z funduszu badań własnych BW 2002.

LITERATURA

1. Cichosz P.: Systemy uczące się, WNT, Warszawa 2000.
2. Cox I.J., Wilfong G.T: Autonomous robot vehicles, Springer-Verlag 1990.
3. Charniak E., McDermott D.: Introduction to artificial inteligence part 2, Addison Wesley Publishing CompanyToga.
4. Dorobczyński D., Dutkiweicz P., Jedwabny T., Kozłowski K., Majchrzak J., Niwczyk G.: Mobile robot soccer, Proc of Sixth International Conference on Mtethods and Models in Automation and Robotics, Międzyzdroje, Poland 2000, pp. 599-604.
5. Gałuszka A., Świerniak A.: Planning in multi-agent environment as inverted STRIPS planning in the presence of uncertainty., Proc. of 2nd WSEAS Int. Conf. on Instrumentation, Measurement, Control, Circuits and Systems (CD-ROM), Cancun, Mexico 2002.
6. Maja J Mataric: Behavior-Based Control:Examples from Navigation, Learning, and Group Behavior, Journal of Experimental and Theoretical Artificial Intelligence, special issue on Software Architectures for Physical Agents, 9(2-3), H.Hexmoor, I. Horswill, and D. Kortenkamp, eds., 1997, 323-336.
7. Nilson, N.J.: Principles of Artificial Intelligence. Toga Publishing Company, Palo Alto, C.A. 1980.
8. Skrzypczyk K.: Heuristic approach to a mobile robot collision free path planning problem, Proc. The Symposium on Methods of Artificial Intelligence in Mechanics and Mechanical Engineering, pp. 249-55, Gliwice, November 2001.
9. Skrzypczyński P.: Budowa geometrycznych map otoczenia do celów nawigacji autonomicznego robota mobilnego, Rozprawa doktorska, Poznań 1997.
10. Weld, D.S., C.R. Anderson i D.E. Smith. 1998. „Extending Graphplan to Handle Uncertainty & Sensing Actions". *Proc. 15th National Conf. on AI, 897-904.*

11. Yale Center for Computational Vision and Control. 1998, *PDDL – The Planning Domain Definition Language*, Tech Report CVC TR-98-003/DCS TR-1165.

Recenzent: Prof. dr hab. inż. Krzysztof Kozłowski

Abstract

One of the fundamental requirements for mobile vehicle control system is the robot autonomy The robot autonomy is defined as an ability of operation without control of a human operator in an environment model of which is unknown and react a dynamical changes of the environment. Meeting the requirements implies the system has to cope and solve many complex problems like task planning, collision free trajectory generation, operating on the basis of imprecise and uncertain information, environmental model building.

Operating by the robot in a real-world environment of a complex structure such as an office without any a'priori information about the environment is the very complex problem. Therefore in practise a floor plan is often given to the control system. Such an approach simplify a primary task decomposition process. In the paper a concept of a hybrid system for controlling a team of autonomous floor sweepers is presented. The goal of the team is to clean up the given office area and return to their parking places. The role of the control system is to plan and coordinate an activity of each team-agent. The system architecture consist of two layers. The upper layer – planner is modelled using STRIPS representation. The lower one that is responsible for the robots collision free movement execution uses the behaviour based control conception.