Software for specification and monitoring of mobile robots missions

Rafał Wiglenda, Wawrzyniec Panfil, Wojciech Moczulski

Katedra Podstaw Konstrukcji Maszyn, Politechnika Śląska w Gliwicach

Abstract: The need of a software used for a mission control and maintenance of a group of robots increases with the expansion of mobile robotics applications. This paper discusses requirements for a software dedicated for semiautonomous mission planning, execution and monitoring. The authors present a working solution proposal which meets the majority of requirements stated before. The designed software contains path planning and trajectory generating algorithms necessary for the navigation of robots. The solution is equipped with an additional tool used to create surroundings for the mission in the simulation environment. The software is an universal tool which allows its application with different types of robots and missions. The applications of the system confirm the demand for mission control software and the results obtained from tests of the project prove its design propriety.

Keywords: mission planning, control and maintenance, group of robots

1. Introduction

In the last twenty years the field of mobile robotics has been experienced a significant growth. The widespread use of the mobile robots and increase of their level of autonomy cause a change in the nature of the high level control systems. Historically, the robots were controlled usually by an operator in a teleoperated mode via direct commands like "*turn left by 30° …*". In addition, there was a lack of missions where robots were cooperating together. Since then, the autonomy which the robots are equipped with has evolved and therefore, the commands issued by the operator are substituted by more general tasks (for example: "*go to location…*", "*follow the target*", etc.). Direct commands for actuators are now usually generated by the control algorithms onboard the robot.

The change in robots autonomy allows the operator to supervise more than one robot at the same time. Therefore, it is possible to maintain a mission where its tasks are shared between the robots. This increases the probability of mission success and enables to perform autonomous or semiautonomous missions of a group of cooperating autonomous vehicles. An example of such a mission is the Grand Challenge UK 2008 [7] where ground and aerial vehicles were used.

The paper is based upon the research made in the Master Thesis "Software for modelling and monitoring mission of a group of robots" [15].

2. Project Objectives

The main aim of the research was to design a software able to define a mission of a group of semiautonomous, cooperating robots. It was expected that the solution would allow the user to define the mission agents' types including their onboard equipment. In addition, there was a requirement to make the software being modular and easily expandable. The main expectations related with the project were:

- Dynamic creation of robots' surroundings – parsing digital maps to the simulation environment.
- Mission creation – definition of types and equiment of inserted robots.
- Two control types: semiautonomous and teleoperation.

3. Existing Solutions

There are various publications concerning different approaches to the problem of user interfaces for mission planning and control.

Chadwick [2] suggests a single-operator interface for many vehicles control. The solution is equipped with an interface for autonomous robot navigation with a possibility of a human influence in case of exceptions. The operator gets information from a camera and sensors. Moreover, messages are sent from the robots to the user in order to pay attention if necessary. The usability of the interface was proven through mission experiments with different operators.

An approach to web user interfaces for autonomous mobile robots is presented in the paper [13]. The authors' interface includes data about states of the robots, their surroundings and progress of the mission. The mission can be performed with a high level or teleoperating control of the robots.

The document [4] presents a framework of the system maintaining a mission in real time and keeping control on multiple robots. The authors discussed problems of the system design that are: localization, realtime communication and resource management. They proposed the architecture which solves these problems.

The authors of [9] designed and developed an interface which may be changed dependingly on specification of the robot and the character of the mission. The aim of this solution was to design a user-friendly interface which allows an interaction between the operator and the group of robots and also between robots within the group.

In the paper of Gaertner and Holzhausen [5] is presented another approach to the user interface adaptivity. They present the performance of their system for two different types of robots and explain the idea of modularity of their design. The application is equipped with two types of mission control: semiautonomous and teleoperation.

A comprehensive approach to the mission design and maintenance was carried out within the MissionLab project [1, 3, 6, 10, 11] of the Mobile Robot Laboratory at Georgia Tech. A complete application is freely available for tests and development. There are also provided an user manual and publications discussing the concepts of the system and its versatility. Software is ready to work in a simulation environment and also with real unmanned vehicles.

The PerceptOR solution presented in paper [14] shows a complete interface of a Control Station which consists of a 2D map of the terrain. The mission can be executed in three modes:

– Monitoring – user supervises the robot movements.
– Manual Override – teleoperated mode.
– Edit Mission – user defines waypoints to reach by the robot and the obstacles to avoid.

Additionally, the interface is equipped with a view from robot's camera and a display with robot's and sensors' states.

4. Software Design

The system was designed in C# language with the use of Microsoft Robotics Developer Studio's simulation environment. The application development was preceeded by an optimal conception selection and a design of the system structure.

4.1. Conception

The conceptions to choose were mainly based on the paper [8] where possible human-robot interfaces were discussed. In the selection the human-robot ratio and the level of mission's autonomy were taken into account. The final proposal is a system dedicated for one human operator being able to control multiple robots. The control can be performed in two ways: by high-level commands or remotely. The mission control has a high-level autonomy where the commander can interfere at any time. Such a solution ensures a higher mission safety which is required especially for a growing up system which may be laden with some inadequacies in autonomy. The name of the project is *Mission Manager*.

4.2. Structure

The program consists of three main classes:

– `Mission Manager` – represents the main application window, where the user can define and control the mission.
– `Map Creator` – is an additional tool which allows loading a digital map in DXF format and convert it into three-dimensional objects in Microsoft Robotics Developer Studio.
– `Path Finder` – a class used for finding the optimal path of a robot. Currently, it is equipped only with A* algorithm

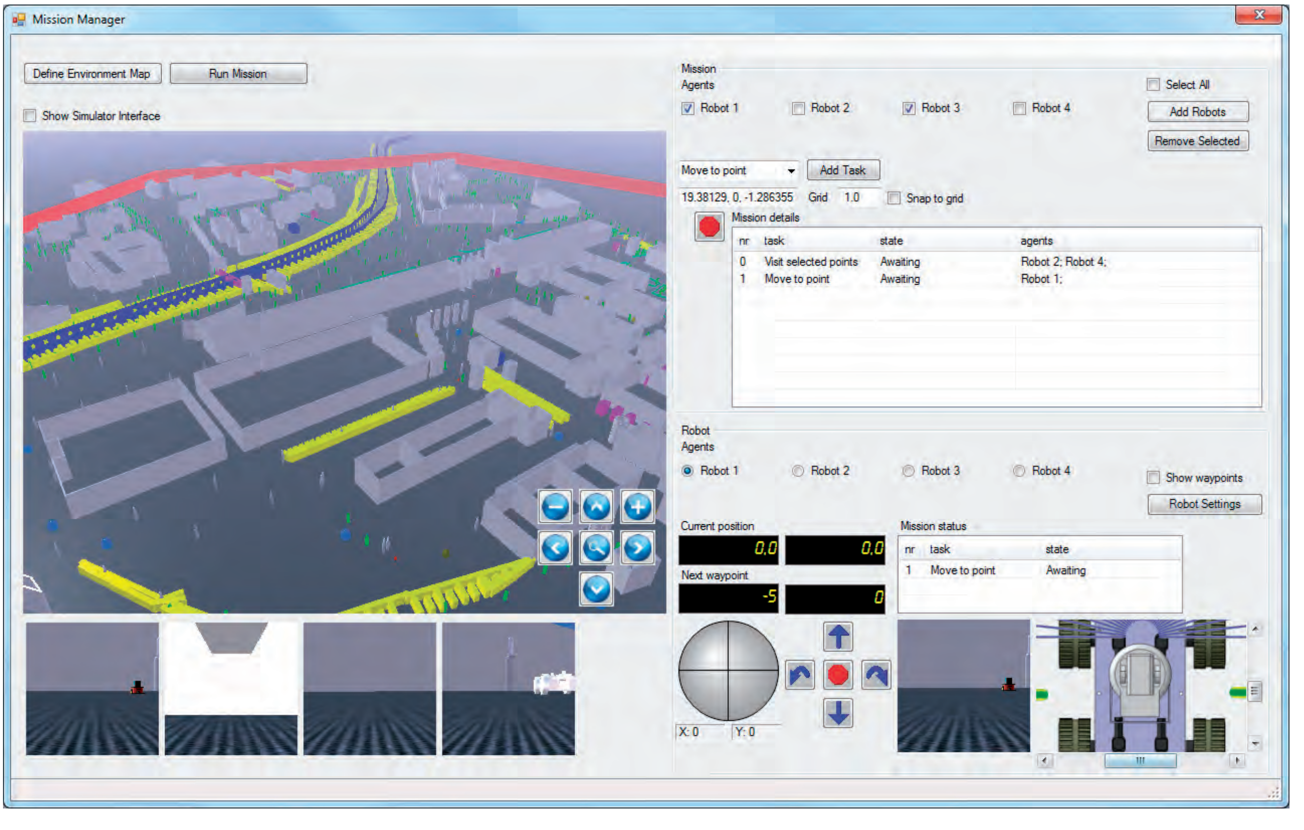


Fig. 1. Mission Manager main window
Rys. 1. Okno główne programu Mission Manager

implementation, but also new algorithms can be added easily.

Other classes represent additional objects required by the application tools or are used for communication with the Microsoft Robotics Developer Studio's services.

4.3. Development

The main window of the designed application is shown in the Figure 1. The greatest part of the window belongs to a display which shows a view from the Microsoft Robotics simulation environment. The user can change the view of the display using combinations of keyboard keys and mouse manipulations or clicking one of the buttons on the right bottom corner of the display. Below the display is placed a panel with views coming from robots' cameras. Above the main display a *Define Environment Map* button is present. It opens an additional application used for creating surroundings of the robots.

On the right top part of the application is placed a panel for semiautonomous mission specification and maintenance. It includes the list of available robots and a list of tasks to perform. In this part the robots can be added to the simulation. Each robot can be equipped with a different set of sensors which services start immediately and the devices are ready to use. The robots may be removed from the simulation using the *Remove Selected* button. In this panel the tasks can be created, changed or removed if desired. The first version of the software enables the operator to define two tasks:

- *Move to point* – a task for one robot. The user chooses a robot and specifies the point it moves to. The optimal path of the robot's movement is calculated with the help of A* algorithm incorporating procedures for obstacle avoidance. The resulting waypoints are sent to the robot. Then, after running the mission the robot starts executing the trajectory using path generation algorithms.
- *Visit selected points* – a task for multiple robots. The operator specifies which robots from the group are assigned to the task and selects the points to visit by the robots. The tasks' allocation algorithms obtain the optimal points distribution between the robots. Then, the path for each robot is calculated and executed in the same way as in previous task.

A Robot panel is placed below the panel for semiautonomous actions. It provides data collected from the robot which are:

- Robot's position.
- Position of the next waypoint to reach.
- List of tasks to execute.
- View from robot's camera.
- Sensor states of the robot.

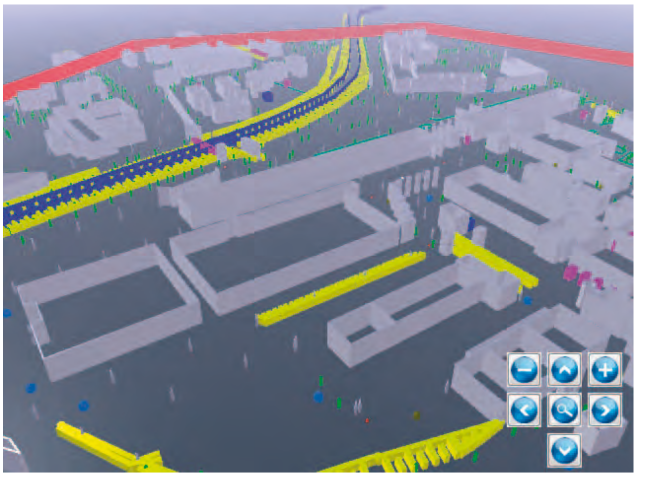
Additonaly, the Robot panel includes buttons for teleoperating mode which can be switched on during the mission execution at any time.

5. Results

In the Figure 1 on the main display is shown an exemplary view of the simulation environment after loading and converting a Dxf map. All objects represented in the Dxf file as polylines are changed into sets of cuboids. The objects shown on Dxf map as block insertion points are displayed as capsules. Each object type has an user defined dimension and color. In Figure 2 are presented pictures showing the same area (Faculty of Mechanical Engineering of the Silesian University of Technology, Gliwice, Poland and it's surroundings) in the simulator 2(a) and in the Google Earth 2(b).

The pictures in Figure 3 show the results of using the tools for mission specification. Figure 3(a) presents the robots added to the simulation. In the picture 3(b) a visualization of a robot's path to a selected by the user point is visible. This path was generated using the generalized A* algorithm.

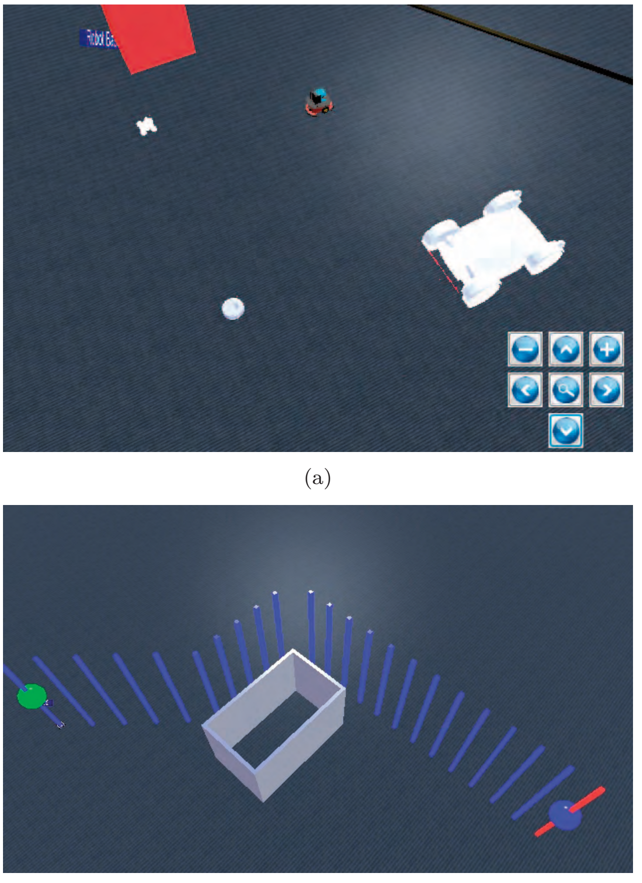
The tests on the designed software and the results analysis show that solution is working properly. The most problematic is the performance of the A* algorithm which requires a big computational power for long paths. That makes it difficult to use in a real-time mission. However, it is working satisfactorily when the number of found waypoints is smaller than 30.



(a)

(b)

Fig. 2. The area of the Faculty of Mechanical Engineering in the simulator 2(a) and in the Google Earth software 2(b)
Rys. 2. Okolice Wydziału Mechanicznego Technologicznego w symulatorze 2(a) i w programie Google Earth 2(b)



(a)

(b)

Fig. 3. Robots in simulation (a) and exemplary path for one of robots (b)
Rys. 3. Roboty w symulacji (a) i przykładowa trajektoria jednego z robotów (b)

6. Conclusion

The aim of the project was achieved successfully. The requirements were met and the available functions of the software are working properly. The software needs to be evaluated in a real environment and then modified if necessary. These activities will enable to use the software in various robotics applications.

The development has a lot of functions and is easy to build up in the future with new capabilities. That makes the design a good framework for future research.

A modification of this solution is being implemented in the framework of a project „Multitask mobile robots using advanced technologies" made by the Department of Fundamentals of Machinery Design together with Institute of Technology and Maintenance in Radom, Poland. This project and potential future applications of the created interface confirm the relevance of creation the system and its usability.

6.1. Further Work Suggestions

In order to improve the existing solution a set of modifications can be implemented:

- Improvements of the searching algorithm – automatic optimization of nodes density for A* algorithm, implementation of A* modifications and implementations of other searching algorithms.
- Verification of the system in a mission with real robots.
- Refining the user interface to make it more intuitive.
- Adding compatibility with other digital map formats.
- Implementation of other available robot and sensor types.

The suggestions written above will increase the versatility of the software which may be an interesting option for future researchers and commercial users.

Bibliography

1. Cameron J.M., Mackenzie D.C., Arkin R.C. (1995): *Multiagent mission specification and execution*, [in:] Intelligent Robots and Systems 95. *Human Robot Interaction and Cooperative Robots*.
2. Chadwick R.A. (2006): *Operating multiple semi-autonomous robots: monitoring, responding, detecting*. [in:] Human factors and ergonomics society annual meeting proceedings. *Cognitive engineering & decision making*.
3. Endo Y., Mackenzie D.C., Arkin R.C. (2004): *Usability evaluation of high-level user assistance for robot mission specification*, [in:] IEEE transactions on systems, man, and cybernetics.
4. Franchino G., Butazzo G., Facchinetti T. (2005): *A distributed architecture for mobile robots coordination*, [in:] IEEE international conference on emerging technologies and factory automation, 10th edn.
5. Gaertner K.-P., Holzhausen K.-P. (1994): *A modular control station for semiautonomous mobile systems*, [in:] Intelligent vehicles '94 symposium.
6. Georgia Tech Mobile Robot Laboratory (2006): *MissionLab. User manual for MissionLab version 7.0.*
7. *Grand Challenge UK 2008 – characterizing the mission*, [in:] Defense Update. International Online Defense Magazine, `http://defense-update.com/features/2008/august08/challenge158083.html`.
8. Johnson C.A. (2002): *Enhancing a human-robot interface using sensory egosphere.*
9. Kawamura K., Nilas P., Muguruma K., Adams J.A., Zhou C. (2003): *An agent-based architecture for an adaptive human-robot interface*, [in:] Hawaii international conference on system sciences, 36th edn.
10. Mackenzie D.C., Arkin R.C. (1997): *Evaluating the usability of robot programming toolsets*, [in:] The international journal of robotics research.
11. *MissionLab v7.0*, [in:] Mobile Robot Laboratory at Georgia Tech, `http://www.cc.gatech.edu/ai/robot-lab/research/MissionLab/`.
12. Moczulski W., Adamczyk M., Januszka M., Panfil W., Przystalka P., Wylezol M. (2009): *Team of specialized mobile robots for group inspection of large-area technical objects*, [in:] *Robot Motion and Control*, Kozlowski K. (ed.) Tom: 396, Springer, 411–420.
13. Moreau B., Tomatis N., Arras K.O., Jensen B., Siegwart R. (2000): *A multi modal web interface for task supervision and specification*, [in:] Mobile robots XV and telepresence technologies VII.

14. Platz C., Kelleher M., Leonard E., Blake C., Remala S. (2002): *PerceptOR robotics project. Operator Control Station (OCS). Interface design: proposal and process.*
15. Wiglenda R. (2010): *Software for modelling and monitoring mission of a group of robots*, Master Thesis, Silesian University of Technology, Gliwice.

Oprogramowanie do tworzenia i prowadzenia misji grupy robotów

Streszczenie: Wraz z rozwojem zastosowań robotyki mobilnej rośnie zapotrzebowanie na oprogramowania stosowane do prowadzenia misji grupy robotów. Artykuł przedstawia główne wymagania stawiane oprogramowaniu dedykowanemu półautonomicznemu planowaniu, przeprowadzaniu i nadzorowaniu misji. Autorzy prezentują propozycję aplikacji spełniającej większość wymagań. Utworzone oprogramowanie zawiera algorytmy planowania ścieżki i generowania trajektorii niezbędne do nawigowania robotami. Rozwiązanie wyposażono w dodatkowe narzędzie tworzenia otoczenia misji w środowisku symulacyjnym. Oprogramowanie jest uniwersalnym narzędziem do sterowania różnymi typami robotów mobilnych w misjach o różnym charakterze. Obecne zastosowania i dalsze prace rozwojowe potwierdzają popyt na aplikacje dla sterowania misją, zaś wyniki otrzymane w ramach testów programu potwierdzają poprawność jego działania.

Słowa kluczowe: planowanie, kontrola i prowadzenie misji, grupa robotów

mgr inż. Rafał Wiglenda

Absolwent Wydziału Mechanicznego Technologicznego Politechniki Śląskiej w Gliwicach na kierunku Automatyka i Robotyka, specjalność Projektowanie i Eksploatacja Maszyn, a także kierunku Autonomous Vehicles Dynamics and Control w School of Engineering, Cranfield University (Wielka Brytania).

e-mail: wiglenda@wp.pl

dr inż. Wawrzyniec Panfil

Adiunkt w Katedrze Podstaw Konstrukcji Maszyn Politechniki Śląskiej. Jego zainteresowania naukowe skupiają się wokół projektowania, konstruowania, a w szczególności sterowania robotów mobilnych. W swych pracach koncentruje się nad zastosowaniami metod i środków sztucznej inteligencji w sterowaniu grupami robotów.

e-mail: wawrzyniec.panfil@polsl.pl

prof. dr hab. Wojciech Moczulski

Profesor zwyczajny w Katedrze Podstaw Konstrukcji Maszyn Politechniki Śląskiej. Prowadzi badania m.in. w zakresie diagnostyki maszyn i diagnostyki procesów ze szczególnym uwzględnieniem zastosowań metod i środków sztucznej inteligencji. Jest autorem ok. 200 publikacji – artykułów, referatów konferencyjnych, autorem i współautorem kilku monografii, w tym: „Diagnostyka procesów przemysłowych. Modele, metody sztucznej inteligencji, zastosowania" (WNT, Warszawa 2002) oraz „Modelowanie, diagnostyka i sterowanie nadrzędne procesami. Implementacja w systemie DiaSter" (WNT, Warszawa 2009). Jest przewodniczącym Sekcji Diagnostyki Komitetu Budowy Maszyn PAN oraz redaktorem Działu Diagnostyka czasopisma Zagadnienia Eksploatacji Maszyn.

e-mail: wojciech.moczulski@polsl.pl