Holonic Communication Structure with Formal Models of Cooperative Autonomous Mobile Robots

Grigor Stambolov¹, Tadas Sperauskas², Rimvydas Simutis², Arunas Lipnickas²
¹Technical University of Sofia, Kliment Ohridski St. 8, Sofia-1000, Bulgaria, e-mail: gstamb@gmail.com
²Kaunas University of Technology, Kęstučio g 27-204, LT-44312 Kaunas, Lithuania, e-mail: tadas.sperauskas@stud.ktu.lt, rimvydas.simutis@ktu.lt, arunas.lipnickas@ktu.lt

Abstract - Recently many authors have been concerned with the problem of cooperation between autonomous mobile robots. In this paper, we present a communication structure which is a part of holonic communication structure in the team of cooperative autonomous mobile robots. To implement the proposed cooperation between autonomous mobile robots the developed UML sequence, collaboration diagrams, Petri Nets models for simulation of communication and cooperation structure are used. Results are implemented in real experimental environment, including two autonomous mobile robots – Koala and Khepera.

Keywords - Cooperative Autonomous Mobile Robots system (CAMRs), Holonic System, Petri Nets, Control strategy

I. INTRODUCTION

In recent years researches use methodology based on Social Theory for describing the behaviour of robot groups [1]. The development of such “individual social” robots requires the use of models and techniques different from “group social” robots [2]. The new generation of Cooperative Autonomous Mobile Robots system (CAMRs) stands ahead of the challenge to solve two basic, diametric opposing tasks of cooperative robotic:

• Integration of information and control systems with the goal to automation of proceeding process;
• Support relative autonomy to the elements of cooperative robotic system through transition from centralized to decentralized embedded control structures.

Existing approaches ensures only simultaneous execution of these two tasks, for instance multi-agent (fractal, bionic and holonic) system [3]. Their applications and realizations have been enabled by fast response and close cooperation of autonomous mobile robotic systems.

This paper has a goal to present results of investigations in the field of development of methodology for cooperation of autonomous mobile robots by allowing the searching possibility of preliminary definitely decisions of possible tasks. The presented methodology consists of four basic components:

1. Definition of the field when the methodology can be implemented;
2. Models, that present different aspects of the examined field and decision taking processes at various levels;
3. Multitude of methods for transforming of representatives from one model to another one at different levels of abstraction;
4. Multitude of procedures and instructions, with defined systematic application of methodological steps.

The methodology is described in two main parts. The first one is devoted to methodological aspects of development of CAMRs system level and to present basic stages of the development – formulation of basic requirements to CAMRs and fundamental approaches in the architecture. The relational links of the reconfigurable manufacturing systems [4] with the holonic reference architecture CAMRs are developed in [5]. The second part of the methodology is devoted for the development of the communication structure and control of CAMRs by using object-oriented (UML) and formal (Petri nets) methods for modelling and verification.

II. BASIC APPROACHES FOR DEVELOPMENT OF CAMRS

The proposed methodology for development of CAMRs is based on applying two fundamental approaches: references architecture with different abstraction levels and on the use of the formal methods for modelling and verification of elements (see Fig.1).

The reference architecture represents structural half-formal description of the system. This architecture is the fundament of the methodology for development of CAMRs system level and is expressed in the defined structure of the system, components and their functionality. The reference architecture is used for the development of different generalized conceptual models that fit the recommendation “the best” practice in the examining. The basic advantage of this approach is based on reduction of the models development phase by using generalized models. It allows to minimize the risk from admission of mistakes, through applying the principle of over re-using of models. The application of reference architecture gives the possibilities for creating the library with models of important processes, structures and architectures.

Important stage in realization of CAMRs is based on a reference architecture, i.e. the choice of suitable environment and modelling language. The predominant part of these models describes processes in a real time.
The object-oriented languages and particularly the Unified Modelling Language (UML) is very powerful tool and most suitable environment for modelling of concurrent and fast changing processes. The basic characteristics of UML satisfy the requirements of CAMRs and allow achieving high quality models and systems.

Another important approach, fundamental for proposed methodology is connected with the use of formal methods for modelling and analysis (verification) of control models in real time. Advantages of formal methods are: formalism for modelling systems, language specification, formal semantics for interpretation of the system characteristics, as well as methods for verification.

Fig. 1. Methodology for development of CAMRS.

III. CASE STUDY COOPERATION

To implement the communication between robots in the team, the following experimental environment is proposed - a work space in the room (approximately 2x2 meters), the objects (in this case three colour balls), three nests, where the objects must be provided, personal computer (PC), WEB-camera and two autonomous mobile robots – Koala and Khepera (made by K-Team S.A.).

The scenario that must be realized by these two robots is as follows:

Three objects are arranged on the floor in the room. The WEB-camera scans the floor in the real time and sends information to PC. In Matlab environment the map of the work space is build based on these images. On the map the borders of work environment, obstacles (static and dynamic) as well as the position of objects are marked. The objects are distinguished only by colour (but not by the form), and their locations are marked on the map with the coordinates. The data map is sent from PC to Koala.

Koala, being a “master” in the hierarchy of system, scans the floor (with built in camera), and after localizing the objects (by colours), sends a data to Khepera, which is a “slave”, having the aim to gather objects. After information is submitted, Khepera searches, takes and puts the objects in special nests for every colour ball. When Khepera finishes, it sends the information to Koala and the “master” checks the work of Khepera. After checking process Koala sends a message to PC for testing a quality of Khepera’s activities.

This scenario represents a complex task, consisting of few subtasks - navigation, mapping, localization, foraging, image processing, cooperation, communication and coordination between robots. The order of their execution is presented in following sequence:

1. Mapping of experimental environment – WEB-camera takes images of work environment in the real time, the image processing programs finds: the objects which have to be moved, the nests where the objects have to be put in and the natural obstacles present in the experimental environment. The information is processed in PC (Matlab environment) and is sent to the “master” robot Koala;
2. Localization and Indication of objects – Robot Koala, with built-in Pan-Tilt Camera, moves to each object and to the nest, discerning it and indicating it on the map with their coordinates. In the base of these data (for objects and for nests) PC (or Koala PC) makes the optimal route, which has to be sent to robot Khepera by using radio communication.
3. Cooperation, communication and coordination between robots – The radio communication between two robots is based on radio beacons, the coordination process is done by control of PC and the cooperation finds expression with distributing tasks between robots in the
base of hierarchical behaviour – “master-slave”. Tasks are divided into subtasks, which are executed by steps: foraging of the object, localization, identification, putting the object, transportation to the nest, leaving the object in the nest, checking for the accomplishment of work. In the case of mistakes the robot must repair its work. Except the difference by type, cooperation and coordination algorithms need the presence of move-, put-, leave-algorithms.

Through the complex of experimental works presented here, like functionality and sequence of actions, it is proposed to divide generalized tasks to the subtasks, as for based on different strategies, which has to be executed by a pair of robots. The strategies have to be executed in a strong hierarchal sequence, because only execution of given subtasks gives the possibility to continue of preliminary planning scenario. In this paper we present only the communication structure realized in the base of radio beacons. Other aspects of experimental work are very difficult and are themes of another discussion.

IV. COMMUNICATION STRUCTURE IN THE FRAMEWORK OF THE HOLONIC ARCHITECTURE

The deployment view of CAMRs communication network is shown in Fig.3. The presented structure corresponds to the proposed holonic reference architecture. The task holon (being as a software agent) may share the same computer used by the control holon. The communications between robots (resource holon), control system (control holon) and Order System (task holon) are accomplished on the base of Intranet protocols. The connection between two robots (Khepera and Koala) and workstation (PC) is realized through radio beacons using RS232 interface. The collaboration and sequence diagrams for modelling the communications in the CAMRs are designed on the base of the class diagram representing the suggested holonic reference architecture [5]. They cover the scenario, depending on the appearance or absence of external and internal orders in the system.

Fig.3 represents the sequence diagram of interactions when the CAMRs system is working in a normal mode. Communications between different holons follow the sequence described below:

- The customer submits the complex task to the robotic system. The task is received by the task holon where the complex task is split into subtasks towards the requirements and functionality of the resource. In this case there are described only these tasks for which communication structure is needed;
- So configured task requirements are passed to the control holon (PC) that determines the tasks models (data for objects and order for identification of objects) needed in the control processes;
- The control holon sends the tasks models to the resource holon (part 1- Koala), and the resource holon (Koala) determines the identification of objects by colour with built-in Pan-Tilt camera;
- After identification of objects Koala determines the orders for each object individually (in this case, three separately orders). In the base of information for identification the objects and the nests, where the objects must be put in, Koala passes the needed information to the control holon for solving the optimal transport scheduling task for sequence of orders’ execution;
- The control holon by means of the corresponding optimization algorithms determines the optimal schedule
and transfers them back to the resource holon (Koala);

- Having the whole information needed for the execution of tasks, the resource holon (Koala) communicates with the resource holon (Khepera) by radio beacons and sends orders for each object. In the sequence diagram of interactions (Fig.4) orders are presented generally, like one task;
- When Khepera finishes the tasks, it reports to Koala that orders were completed successfully;
- Koala checks the work of Khepera with Pan-Tilt camera and sends the results to the control holon.

Fig. 4 represents the sequence diagram of detailed interactions in resource holon between Koala and Khepera in the part of communication only for transporting the objects.

The communications between the robots in resource holon follow the sequence described below:

- Koala sends order for object 1 (1 of 3 objects) with optimal schedule path to Khepera;
- Khepera executes the order (object 1) and sends a message to Koala that the task 1 is completed;
- Koala sends to Khepera the order for object 2;
- The procedure is analogous for the next objects. The final operation is:
  - Khepera executes the order (object 3) and sends a message to Koala that the final task 3 is completed.

After checking the work, there is a possibility to have two scenarios:

- The objects are put correctly in the nests – the operation is done;
- The objects are put incorrectly in the nests – follow new orders for “wrong” objects.

Fig. 5 represents the collaboration diagram of detailed interactions in resource holon between Koala and Khepera in the case when tasks aren’t completed.

![Collaboration diagram of interactions between Koala and Khepera after incorrectly work.](image)

Fig. 5. Collaboration diagram of interactions between Koala and Khepera after incorrectly work.

The task can be not completed – the objects are put in the wrong nests or for certain reasons are again on the work floor. In this situation the scenario of sequence actions is identical, to the order for new task. The communications between the robots and PC follow next steps:

- Modification of data for objects;
- Identification of objects (from Koala);
- Request for scheduling (from Koala to control-holon);
- Sending of optimal schedule (from control-holon to Koala);
- Order(s) for object (1, …n) (from Koala to Khepera);
- Completion of the task(s) (from Khepera);
- Sending the message “Task(s) (1, …n) are completed) (from Khepera to Koala);
- Koala checking the work of Khepera and sending the results to control holon.

V. PETRI NET COMMUNICATION MODELS OF CAMR

Petri Nets (PNs) and their modifications provide a very flexible and expressive notation, based on simple formal concepts. They have proven to be useful for the modelling and analysis of several classes of systems, including computer systems, software, communication networks, production/process control systems, knowledge-based systems and manufacturing systems. PNs inherently capture various asynchronous, sequential and parallel interactions between the various system resources and operations [7]. PNs have also been used for performance analysis and evaluation of decision making organizations associated with command, control and communication systems. A few authors have explored different aspects in the field of mobile robots systems - a PN coordination model for intelligent mobile robots [8], topological modelling with Fuzzy PNs for navigation of autonomous mobile robots [9], testing the control system of mobile robot [10], developing the high-level controller with Timed Coloured Petri Nets for searching problems in mobile robots [11].

The dynamic and asynchronous structure of the Petri net is ideally suited to the modelling of communication models for autonomous mobile robots with cooperative functions, when including different type’s tasks, like a navigation, mapping, localization, foraging, image processing, etc. The main contribution of this sub point is to show that Petri Nets can be used as basic modules of a holonic mobile robot system, when a control and communication algorithms for solving the complex task can be efficiently specified within the Petri net framework. The PN model is achieved by analytically modelling various units of the system as Petri net transducers and, explicitly representing the tasks execution and information dependence among them. In our experimental case we will use a hybrid model between Condition/Event and Place/Transition-Net, when the number of rules will be described for each one element of the communication structure in the team of CAMRs.

Complete communication between individual elements in the team of CAMRs can be presented through Generalized Communication Model, building with Petri Nets (Fig. 6).
The model describes globally the communication structure of CAMRs, when in places of the set in generalized type are presented single elements of the system (robots, host computer and web-camera), in transitions the activation conditions are defined [12].

In general the Generalized Model presents type conditions of the actions and communications between simple components of CAMRs. The places P0, P1, P2 and P3 represent every basic element in the communication structure. Each of these places can be described like a segregate Petri net model. The communication structure of CAMRs has four basic elements – Web-camera, Host Computer, Robot Koala and Robot Khepera. For each of these components Petri net model is developed and verification of models through rechiability graphs are made.

After developing the component models with their specifications, the Global Petri net model of CAMRs is developed (Fig. 7).

This model is used for simulation on thereby building communication structure and has next advantages:

- The Global model has possibilities to make simulation of building communication structure;
- In the base of model control algorithms for specific subtasks can be developed in details, as well as possibilities to generate control communication control algorithms;

In the base of this simply Petri net model (developed in environment of HPsim) can be used like a basis Petri net model to developed complex models in another environment, like a Coloured Petri net.

VI. CONTROL ALGORITHMS IN THE COMMUNICATION STRUCTURE

The complex task is composed of a number of subtasks. Refer to Fig. 3 for a diagram of interactions in normal mode of the communication task. The communication task is composed of six main subtasks: Localization of Objects, Identification of Objects, Path Planning, Order Execution, Check Execution and Model Update.

The Localization of objects task is to localize the objects, the nests and the robots on the map of experimental environment in the real time. The localization algorithm requires two pieces of information upon initialization: the notional size $m \times n$ of the environment, and a position reference $(x, y)$ in Cartesian coordinates of the robots (Koala and Khepera), objects...
The Identification of Objects task is to identify the objects and the nests by colour and to update data on the map [17]. This task uses the subtasks: Identification algorithm – developed in the base of Image processing (Pan-Tilt camera), Motion algorithm for Pan-Tilt Camera and Motion Algorithm for Koala.

The Path Planning task is to supply a goal location to the map model of the environment and to produce the optimal solution path from any location in the environment to the goal [6]. Optimal Scheduling Algorithm is developed in the base of coordinate’s data for Khepera, objects and nests.

The Order Execution task has to locate the start position in the environment, to find the objects, to get directions to it, to transport it and to put it in the place of “right” identified nest. This task uses the subtasks: (Move algorithm, Location Algorithm (for Khepera), Get Algorithm and Put Algorithm). The Move subtask controls the motion of the Khepera robot. The Location subtask controls location of objects and nests with the robots onboard sensors. The Get and Put subtasks controls the Gripper, which is a turret that plugged on the basic configuration of Khepera and can be made the object manipulation to be possible.

The Check Execution subtask (realized from Koala) is to locate the objects and the nests in the work environment, using again same identification algorithm.

The Model Update task is invoked when the execution process detects in the case of internal or external disturbances. Update algorithm resets the execution process to eliminate the problems, updates the data for the condition of the robotics system and puts into the operation the modified execution process.

VII. CONCLUSION

This paper presents and discusses an implementation of the communication network of a CAMR using Intranet and RS232 interfaces. The communication models for different situations of experimental scenario are composed on the base of proposed holonic reference architecture and through using of object-oriented (UML) and formal (Petri nets) methods for modelling and verification. The main benefits in using the proposed approaches for design of communication structure of CAMRs may be summarized as follows:

- the formal models are used like a part of methodology for development of CAMRs, which consist of three basic levels – “System”, “Component” and “Cooperation, Communication and Control”;
- the UML communication models, representing different situations of interactions between holons (basic components of CAMRs) and defined their functions and possibilities;
- The Petri net models introduce generalized, component and global models of the system, when in each elements are defined states in every moment of communication, as the conditions, which required for their realization. The developing simulation model can be implemented like a conceptual model in a process of development of the control system of the team of mobile robots.

ACKNOWLEDGMENT

This work has been support by the Lithuanian state science and studies foundation.

REFERENCES