

Strategy Extraction for Mobile Embedded Control Systems Apply the Multi-agent Technology

Vilém Srovnal¹, Bohumil Horák¹, Radim Bernatík¹, and Václav Snášel²

¹ Department of measurement and control, FEECS, VŠB-Technical University of Ostrava,
17.listopadu 15,CZ-708 33 Ostrava-Poruba, Czech Republic
vilem.srovnal@vsb.cz

² Department of computer science, FEECS, VŠB-Technical University of Ostrava,
17.listopadu 15,CZ-708 33 Ostrava-Poruba, Czech Republic
vaclav.snasel@vsb.cz

Abstract. Mobile embedded systems belong among the typical applications of the distributed systems control in real time. An example of a mobile control system is the robotic system. The proposal and realization of such distributed control system represents a demanding and complex task of real time control. In the process of robot soccer game application extensive data is accumulated. A reduction of such data is a possible win in game strategy. SVD decomposition of matrixes is used in data reduction.

1 Introduction

A typical example of a distributed control system with embedded systems is the proposal of the control system of mobile robots for the task robot-soccer game. The selection of this game for a laboratory task was the motivation both for students and for the teachers as well because this was a question of proposing and realizing a complicated multidisciplinary task, which can be divided into a whole number of partial tasks (the evaluation of visual information and processing of an image, the hardware and software implementation of a distributed control system, wireless data transmission and processing of information and the control of robots). For the development of the game strategy itself the multi-agents method is considered with the use of opponent strategy extraction algorithms. The results are used as a control action database for the decision agent.

Embedded systems are represented by up to 11 own and 11 opponent autonomous mobile robots. The core of an embedded control system is a digital signal processor Motorola - DSP56F805. PWM output of the signal processor are connected to a pair of power H-bridge circuits, which supply a pair of DC drives with integrated pulse encoders. For communication the communication module is used with the control IC Nordic nRF2401, which ensures communication with a higher level of the control system.

The higher level of control system is represented by a personal computer. In the PC the signal is entered, which represents the picture of a scene with robots scanned with an above the playground placed CCD camera. At the output is connected radio line which transmits commands for all own mobile robots.

The software part of a distributed control system is realized by decision making and executive agents. The algorithm of agents cooperation was proposed with the control agent on a higher level. The algorithms for agents realized in robots are the same. The control agent determines the required behavior of the whole control system as the response to the dynamic behavior of robots and to the one's own global strategy in the task and knowledge about the last situations, which are saved in the database of the scene. The agent on a higher level controls the other agents [5].

The separate task is the transformation which converts the digital picture into the object coordinates (robots and ball in the task of robot soccer) which are saved in the database of the scene [4]. This database is common for all agents in the control system. Each agent sees actual the whole scene and is capable of controlling its behavior in a qualified way. The basic characteristic of a control algorithm of a subordinate agent is the independence on the number of decision making agents for robots on the playground.

Both agent teams (one's own and the opponent's) have a common goal, to score a goal and not to have any against. For successful assertion of one's own game strategy the extraction and knowledge of an opponent's game strategy is very important. From object coordinates of the picture scene strategy extraction algorithms are created from the opponent's game strategy database.

2 Control Agent

The control agent (Fig. 1) goals (score the goal into an opponent's goal, defend one's own goal area against the opponent players) can be achieved by the correct selection of cooperating agents (division of tasks among agents). The decision making agents are capable of selecting the correct tasks for themselves and further to select the executive agents.

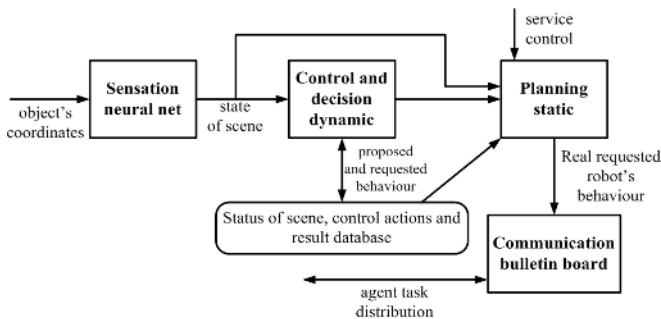


Fig. 1. Control agent scheme

The sensation module contains the neuron network which evaluates the actual state of the scene. The output vector (state of scene) does not describe the position of robots but gives the information about the capability of robots to move into places with critical situations. The neuron network proposed is suitably capable to correctly evaluate any situation.

In the case that the dynamic control and decision making system will not find the solution for a certain critical situation, then it will try to find the partial solution from several situations. The controller selects the relevant tasks for the agents for these situations. The controller saves or updates the information about the scene state and control in the relevant database which can be used for static control.

3 Decision Agent

The main task of the decision agent (Fig. 2) is to schedule the relevant task. The agent will schedule the tasks in relation to the environment and the internal state of the relevant robot. The module of perception provides the information about the environment.

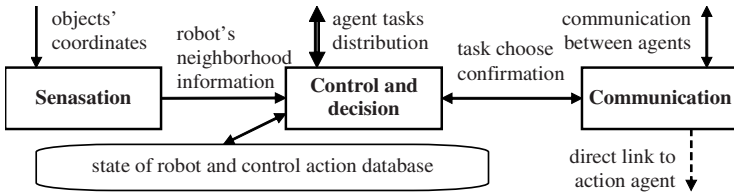


Fig. 2. Decision agent scheme

In the case that more than one agent will select the same task the decision agent must evaluate the probability of the action's success. The agent with the highest probability of success will schedule the actual task. The task selected is handed over to the executive agent which realizes it. The executive agent will receive also the information from cooperating agents for optimization of the robot's movement.

4 Action Agent

The activity of the action agent (Fig. 3) is simple. The agent moves with the robot from an actual position to a new position. The information concerning the actual position is obtained from the sensation module.

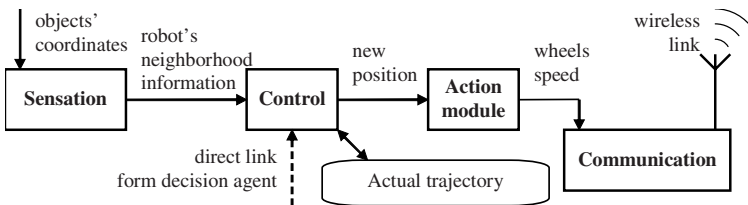


Fig. 3. Action agent scheme

A new position for further moment (for instance another frame of the camera) is evaluated by the control module. The new position is calculated on the basis of a pre-calculated trajectory. The advantage of pre-calculated trajectories is the possibility of the realization of the task of controlling position and time which is useful in avoiding other robots.

After completion of the trajectory calculation the agent will determine the position for next frame (new position) and will transmit it to the action module

5 Strategy Extraction Method

One's own and an opponent's robots create a very dynamically changed environment. This environment is scanned by a CCD camera with a sample frequency (in the present time) up to 50 fps. The picture sample with objects coordinates before processing is demonstrated in Fig.4.

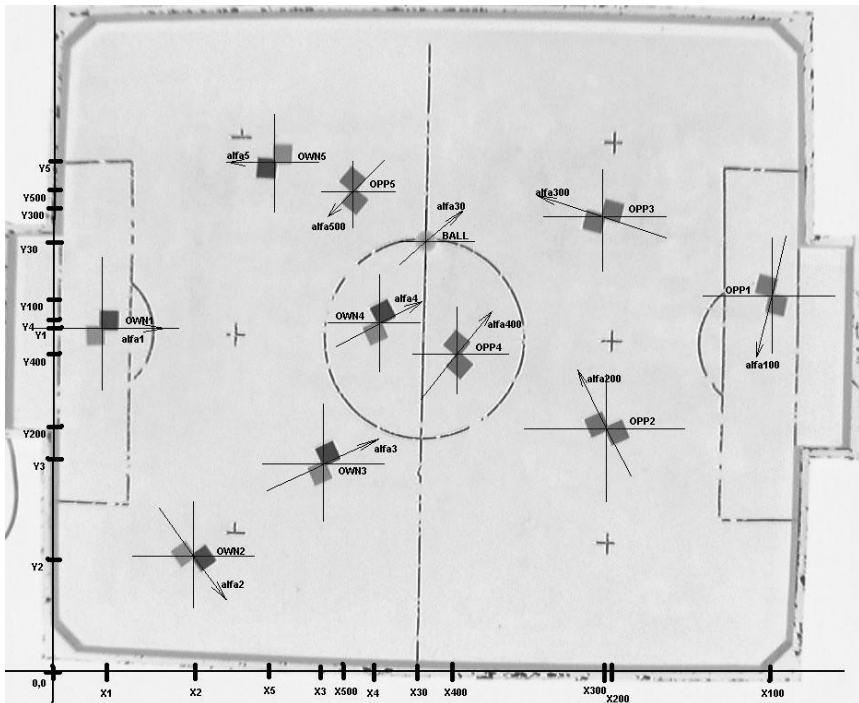


Fig. 4. Picture sample from CCD camera – playground with coordinates of mobile robots

The neuronal net of a control agent in a sensation module process the picture signal and encoded information (position, orientation) saved in one of output vectors. The time sequence of these vectors builds a very wide matrix (up to 420000 vectors - processed samples in the time of a game). This matrix inputs in the proper extraction of the game strategy process by using latent semantic analysis (LSA).

LSA is a statistical model of word usage that permits comparisons of the semantic similarity between pieces of textual information. It was originally designed to improve the effectiveness of information retrieval methods by performing retrieval based on the derived "semantic" content of words in a query, as opposed to performing direct word matching. LSA was used for extracting semantics in many other situations, see [1], [3], [7].

In this paper, LSA is used as a tool for the strategy extraction problem. In our approach, a game is coded as a game matrix.

We can extract vector V_t in time t from game:

$$V_t = \left\{ \begin{array}{l} t, X_1, Y_1, \alpha_1; \\ X_2, Y_2, \alpha_2; \\ X_3, Y_3, \alpha_3; \\ X_{100}, Y_{100}, \alpha_{100}; \\ X_{200}, Y_{200}, \alpha_{200}; \\ X_{300}, Y_{300}, \alpha_{300}; \\ X_{30}, Y_{30}, \alpha_{30} \end{array} \right\} \quad (1)$$

Where are:

X_i is x coordinate of one's own robot i
 Y_i is y coordinate of one's own robot i
 α_i is angle of orientation of one's own robot i

X_{i00} is x coordinate of an opponent's robot i
 Y_{i00} is y coordinate of an opponent's robot i
 α_{i00} is angle of orientation of an opponent's robot i

X_{30} is x coordinate of the ball
 Y_{30} is y coordinate of the ball
 α_{30} is angle of motion α ball

A game matrix (in next GM) we can define in the following way:

$$GM = (V_0^T, V_1^T, \dots, V_{n1}^T, V_n^T) \quad (2)$$

The results in [1], [7], [8], [9] and [6] indicate that LSA can perform matching based on semantic content. The game matrix is analyzed by LSA and semantic information about game is obtained. This semantic information can be interpreted as a strategy. This strategy is use for for agent management see [10] and [11].

LSA is based on singular value decomposition (SVD). The SVD is commonly used in solution of unconstrained linear least squares problems, matrix rank estimation, and canonical correlation analysis [8]. The singular value decomposition takes a rectangular $m \times n$ matrix A and calculates three matrices U , S , and V . S is a diagonal $m \times n$ matrix (the same dimensions as A). U and V are *unitary* matrices with sizes $m \times m$ and $n \times n$ respectively. The matrices are related by the equation $A = USV^T$. Diagonal elements matrix S are called the singular value of the matrix A . Informally, we can say that greater singular value correspond to greater semantic information.

The algorithms of an opponent's game strategy extraction proceeds parallely and independently at the main control process. Exploits of vector (matrix) representation of picture analysis results in going to the extract general elements of an opponent's

game strategy. Knowledge of one's own and an opponent's strategy allows the building of an opponent's game strategy dependent on the decision rules for task scheduling for individual action agents. These rules are gradually saved in the decision rules database by the decision agent. In this way the possibility of temper game state is increased.

6 Software Development

The technology of agents enables the solving of the proposal and implementation of the distributed embedded control system. The embedded systems are used for controlling of mobile robots. For generation of the source code of DSP Motorola processor Metrowerks CodeWarrior can be used. It enables the simulation in real-time and simulation of the hardware.

For development of control software of the distributed system is proposed by the unified modeling language (UML). UML has the corresponding output for servicing the requirements and analysis of models on a higher level of abstraction but it is also connected to the detailed proposal of the conception as threads, multitasking, resource control, etc. [2], [3].

7 Conclusion

The algorithm of the control system should be proposed in a such way so that it would ensure the requirements for the immediate response of control, so that the system with robots would be controlled in real-time. That is why, it is very important so that the algorithm for critical speed would be optimized. The system response should be shorter than the time between two frames from a camera. In the event that this limit is exceeded, the frame is cut out and the control quality is decreased.

The main possibilities of algorithm adjustment are as follows:

- Dynamic control in the control and decision module of a control agent.
- The control and decision modules and communication protocol of the decision agents.
- The strategy of planning in the control model of the action agent.
- Extraction of an opponent's game strategy and using the extraction results for decision rules generation as a part of the rules decision database of a decision agent.

It is necessary to know that the system response should take a shorter time than the time between the frames from the PAL movie camera, e.g. 20 ms. If this limit is exceeded, the frame is dropped and the control quality decreases.

Parallel processing of extraction algorithms of an opponent's game strategy allows adding and refining information of opponent game strategy. This way expands the rule database of a decision agent for decision making within the bounds of one's own game strategy given by a control agent.

Acknowledgement. The Ministry of Education of Czech Republic supplied the results of the project CEZ: J17/98:272400013 with subvention. The Grant Agency of Czech Republic supplied the results of the project 102/02/1032 with subvention.

References

1. Berry, M. W., Browne, M.: Understanding Search Engines: Mathematical Modeling and Text Retrieval. SIAM Book Series: Software, Environments, and Tools, (June 1999), ISBN: 0-89871-437-0.
2. Adam G.K., Grant E. and Adam K.: Qualitative Modelling and Control of Industrial Processes, in Proceedings of the IASTED International Conference on Modelling and Simulation (MS'2000), Acta Press, ISBN: 0-88986-284-2, pages 477-482, May 15-17, 2000, Pittsburgh, Pennsylvania, USA.
3. Douglas B.P.: The UML for system engineering, I-Logix Rhapsody. White-paper, USA 2000
4. Bernatik,R., Horak,B., Kovar,P.: Quick image recognize algorithms. In: Proceeding International workshop Robot-Multi-Agent-Systems R-MAS 2001. VSB Ostrava 2001, Czech Republic, ISBN 80-7078-901-8 p.53-58
5. Srovnal V., Pavliska,A.: Robot Control Using UML and Multi-agent System. In: Proceeding 6th World Multiconference SCI 2002. Orlando, Florida, USA, ISBN 980-07-8150-1, p.306-311
6. Húsek D., Frolov A. A., Řezanková H., Snášel V.: Application of Hopfield-like Neural Networks to Nonlinear Factorization. COMPSTAT 2002, Proceedings in Computational Statistics (Eds.: Härdle W., Rönz B.), 177-182. Physica-Verlag, Heidelberg 2002. ISBN 3-7908-1517-9.
7. Berry, M. W. (Ed.): Survey of Text Mining: Clustering Classification, and Retrieval. Springer Verlag 2003.
8. Praks P., Dvorský J., Snášel V.: Latent Semantic Indexing for Image Retrieval Systems. SIAM Conference on Applied Linear Algebra (LA03) The College of William and Mary, Williamsburg, U.S.A. 2003.
9. Praks P., Dvorský J., Snášel V., Černohorský J.: On SVD-free Latent Semantic Indexing for Image Retrieval for application in a hard industrial environment. IEEE International Conference on Industrial Technology - ICIT'03, Maribor 2003.
10. Smid, J., Obitko, M., Snášel, V.: Communicating Agents and Property-Based Types versus Objects. Sofsem MatfyzPress 2004.
11. Obitko, M., Snášel, V.: Ontology Repository in Multi-Agent System. IASTED, International Conference on Artificial Intelligence and Applications (AIA 2004), Innsbruck, Austria.