1. RFC Stuttgart Team Description 2009

O. Zweigle, U.-P. Käppeler, H. Rajaie, K. Häussermann, A. Tamke, A. Koch, B. Eckstein, F. Aichele, G. Bhalsing and P. Levi

IPVS, University of Stuttgart, Universitätsstraße 38, 70569 Stuttgart, Germany
robocup@informatik.uni-stuttgart.de

Abstract. The 1. RFC Stuttgart robot soccer team is used as a testbed for multi-agent software architecture principles in dynamic real time domains. The current research activities focus on a completely new design of a midsize robot, the enhancement of machine learning algorithms for strategies, new vision methods and a context-aware visualization.

1 Introduction

Since 1999, the 1. RFC Stuttgart - formerly known as the CoPS-Team - successfully took part in RoboCup tournaments. The research objectives in the past year focused on the development of a completley new midsize robot to enhance the overall performance, remove certain hardware design problem and efficiently control robot's movements [RLS+07], [OSBL05] , [OSL05]. Moreover the reinforcement learning approach of basic behaviors presented last year was enhanced to work on a higher level with strategies. Further work was done on the field of vision, context-aware error visualisation and multi-agent task allocation [LSK+08]. Especially on the field of task allocation the work from the past years could be continued e.g. by improving the mathematical method of selection equations [LSZ+07], [SSL+05]. All that work contributes to the continuing research in the field of agent behavior modeling [ZLB+06], situation recognition/learning, distributed world models [BKLL03] and multi-agent systems [KBZ+08], [BKZ+08].

The paper doesn't describe all new innovations, but only focuses on some major issues: The development of the new midsize robot platform (chapter 2), the enhancement of machine learning algorithms for strategies (chapter 3), new vision methods for recognizing arbitrary balls (chapter 4) and a context-aware visualization for monitoring robot and team behavior efficiently (chapter 5). Finally chapter 6 concludes.

2 Development of a new midsize robot platform

The 3D drawing of the new model of the RFC robot which is designed and constructed by the department of Image Understanding of the University of Stuttgart for the RoboCup competitions is shown in figure 1. The robot has four wheels, each driven by a brushless DC motor. The electromagnetic kicker of the robot allows to shoot the ball with adjustable power.

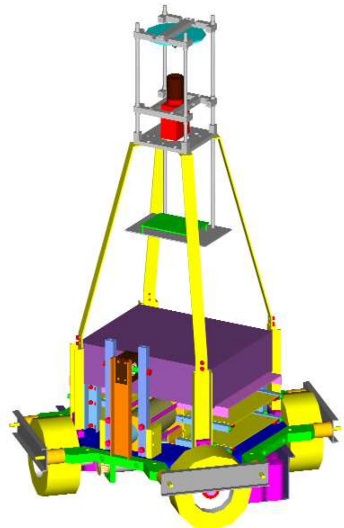

Fig. 1. 3D drawing of the RFC robot model 2009

The distributed control architecture which is used in the RFC robot enables the computer on the robot to delegate a number of control and data acquisition tasks to microcontroller based modules which are mounted on the robot. These tasks include the controlling of the actuators and the measuring of the output signals of the sensors. Though the delegated tasks can be performed independently, the communication between the robot's computer and the microcontroller based modules is essential. The software which controls the whole behavior of the robot in its environment, generates the commands which have to be sent to the microcontrollers through a data communication link. The microcontrollers should also be able to send feedbacks, e.g. the sensor measurements or the status of the microcontroller modules to the computer of the robot. As a data communication link CAN-bus is used. It establishes a connection between the computer and the microcontroller based modules of the RFC robot. The schematic of the distributed control system of the RFC robot is shown in figure 2. As shown in the figure, microcontroller based modules are used in the RFC robot as follows:

- four motor controller modules, each controls a brushless DC motor
- a microcontroller based module for controlling the electromagnetic kicker
- a microcontroller based module for reading the compass sensor
- a microcontroller based module for displaying the information on a LCD

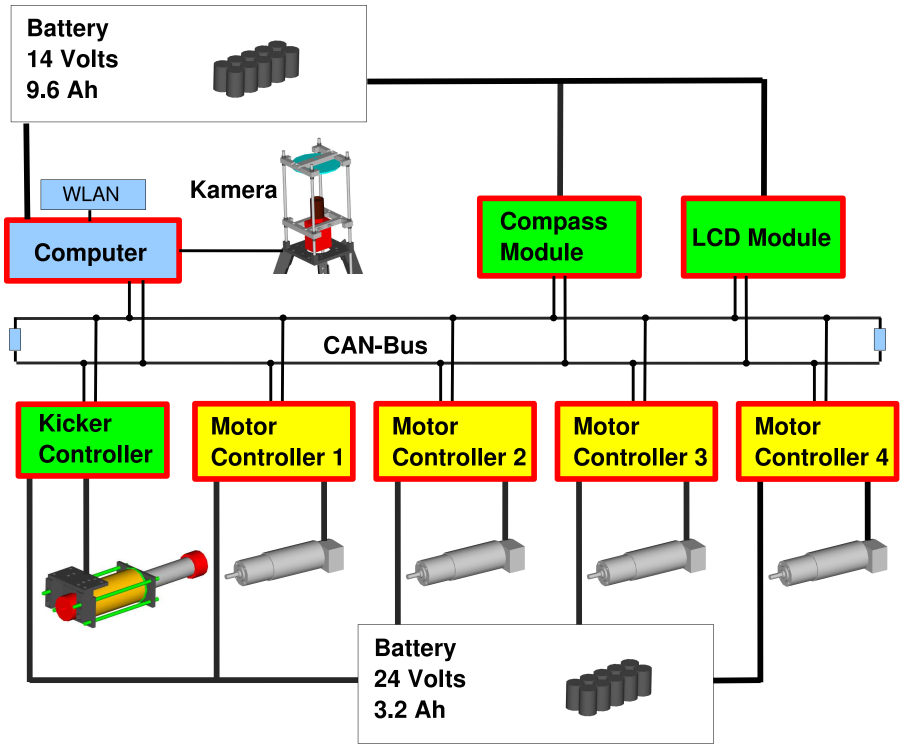

Fig. 2. Schematic of the distributed control system of the RFC robot consisting of 8 nodes on the CAN bus

3 RL-Methods for high-level decision making

Using reinforcement learning methods, in particular model-free temporal difference algorithms, it is possible to improve the behaviour of the robot by learning from its own experiences. With a modification of our existing RL-framework [ZKR+08] we are now able to use our approach not only for tactical lower levels (e.g. for learning optimal movement behaviours) but also for more complex decision-makings on higher strategical levels. For this improvement we modified the RModeler-Module to handle with the needed state-strategy-vectors. Thus the obsolete set of possible actions a (needed for low-level-decisions) is replaced by a predefined fixed set of allowed strategies $S = \{\sigma_1, \sigma_2, ..., \sigma_n\}$. In contrast to the previous version of our framework and due to the use of model free approaches, we drop the calculation of values for (s,a)-pairs (s = state, a = action) and calculate now the expected reward for each $(s, \sigma) - pair$. Both the calculation of the reward for each pair according to the experience and the selection of the best strategy for the current state (using ϵ-greedy-policys), is done online during the game via our modified RL-framework-approach (see figure 3).

Because of the fact that online-learning (during the game) is very critical in

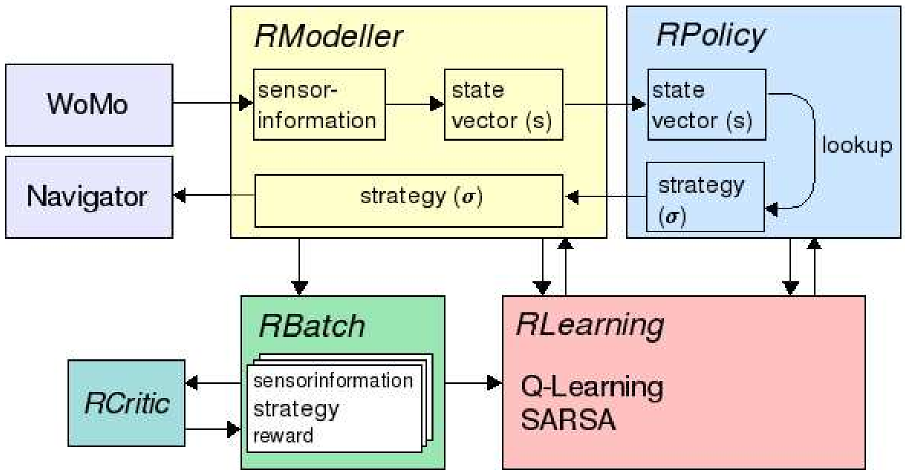

Fig. 3. Architecture of modified Reinforcement Learning System.

terms of time, it is important to keep the possibility of a full exploration of our new state-strategy-space in time. This is why a small state-strategy-space is necessarily needed. In the current configuration we use three possible states and three different strategies. Thus the state-strategy-space would consist of nine elements. But for an additionally minimization, the state-strategy-space can also be limited in a way that only four reasonable state-strategy combinations will be allowed in the selection. Depending on the oppenent team and due to the flexibility of this approach we will extend the state-strategy-space in the future to up to six states and up to six different strategies.

The states we use for selecting the best strategy will be identified by a special helping-function H. The function H is also used to do a coarse generalization of the input-sensor-values. Thereby H holds a datastructure of the simplyfied playing field (only of the opponent half), which divides the playing field into a number of m freely definable tiles. Thus each tile is equivalent to one analogous state.

For the calculation of the reward of the $(s, \sigma)-pairs$, the reward-function (implemented in the RCritic-Module) will use the sensor-information of the World-Model(WoMo). It is specified as follows: An episode runs for the whole time during the robot has the ball or tries to shot a goal. If the robot loses the ball, or dribbles out of the field or missed the goal the robot will be punished. In case of a successful goal-shoot the robot will be rewarded accordingly.

As usual the RPolicy-Module manages the policy locally on each robot. The policy is needed to realize a fast mapping of each state s to the best strategy σ^*. If a better strategy has been explored during the game by one robot the new policy is distributed to all the other robots.

4 Vision

4.1 Recognition of arbitrary balls

The RFC Stuttgart has developed a novel method for ball detection to deal with the challenge of recognizing arbitrary colored balls. The obsolete color segmentation based algorithm has been replaced by a 2-phase method. In phase 1 the image is scanned for circles in a predefined range of radii. In phase 2 features of each circle are extracted for comparison with features of the calibrated ball. For each phase several algorithms have been tested to achieve the individual requirements of its phase. The Standard Hough transform for circles [KBS75], the 2-1 Hough transform [YPIK90] and the generalized symmetry transform [RWY95] were reviewed for phase 1, SIFT features [Low99] and color histograms [SB91] for phase 2. In the 1st phase the most accurate result was obtained by the generalized symmetry transform. Due to real time constraints during the game and quite good results of the Standard Hough transform in less amount of time the generalized local color symmetry was used for the non time critical calibration step and the Standard Hough transform during the game. The best results for the 2nd phase are achieved by color histograms. Altogether the process of the arbitrary ball detection is as follow:

Calibration (see figure 4)

– Phase 1: Find a circle (the ball) in a predefined region of interest by generalized symmetry transform.
– Phase 2: Extract color histogram of this circle

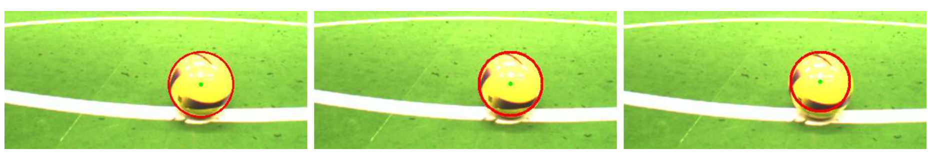

Fig. 4. Calibration step; perspective camera; left: generalized symmetry transform; middle: 2-1 Hough transform; right: Standard Hough transform

During Game

– Phase 1: Find all circles by Standard Hough transform
– Phase 2: Extract color histogram for each circle and compare it with the color histogram of the calibrated ball

The arbitrary ball detection was tested with seven different balls, three different distances (3m, 5m and 7m, see figure 5) and the usage of two camera systems (omnidirectional and perspective). We were able to detect all balls with the perspective camera in each tested distance. With the omnidirectional camera we were able to detect every ball in 3m and 5m. In a distance of 7m the probability of success is 57%.

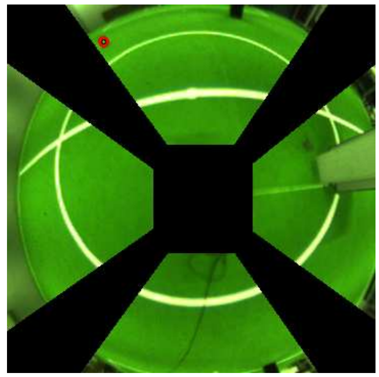

Fig. 5. Ball detection during game; omidirectional camera; ball distance left: 5m

5 Context aware visualisation

Due to recent changes in the RoboCup rules it's not possible to have any interaction with a control notebook. That means for example that error messages etc. can't be displayed in a secure shell (ssh) window because then the user sometimes has to scroll through the window to read certain messages. This will be impossible as touching the mouse or the keyboard is not allowed at all. Due to these restrictions a completely new context aware visualization architecture was created. As an assumption the GUI should be as modular as possible so that it is not only be used for error visualization but also for testing purposes. As a consequence a modular plug-in compatible application was created for implementing several different test and visualization applications in one GUI. One of the advantages is that every user can create and save screen setups according to his needs. Therefore it is possible to adopt the GUI to different situations where for example during a game only an error monitor and a visualization of the field is needed while during testing some other tools like a refbox command simulator or a behavior monitor have to be used. The application has a standard interface for data In-/Output accessible through sockets. This allows to exchange data locally as well as over the network without any changes in the code. Furthermore it is essential that according to the current situation (e.g. testing, real match, test match etc.) only selected information from the robot should be sent over the network and visualized on an external notebook. A decision component on the robot can dynamically decide which information is important and implements the context aware approach. The basic principle of the architecture is shown in figure 6.

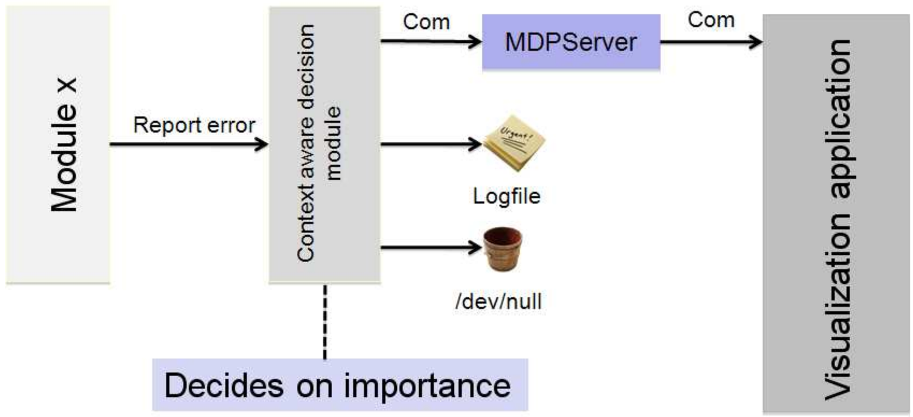

Fig. 6. Data flow of context aware error visualization

6 Conclusion and Outlook

The RFC team has focused this year on several topics in its research efforts. At first a new hardware platform was built that is able to satisfy the requirements of a modern, dynamic and easy to use robot usable for different scenarios. Furthermore the reinforcement learning approach of basic robot behaviors has been transferred to the higher strategy level and allows to dynamically adapt to the strategies of an opponent using a distributed learning algorithm. In the field of vision a stable system for recognizing arbitrary balls was implemented and moreover a highly adaptable context aware error visualization component was introduced.

References

[BKLL03] T. Buchheim, G. Kindermann, R. Lafrenz, and P. Levi. A dynamic environment modelling framework for selective attention. In Visser et al., editor, *Proceedings of the IJCAI-03 Workshop on Issues in Designing Physical Agents for Dynamic Real-Time Environments: World Modeling, Planning, Learning, and Communicating*, 2003.

[BKZ+08] Ruben Benkmann, Uwe-Philipp Käppeler, Oliver Zweigle, Reinhard Lafrenz, and Paul Levi. Resolving Inconsistencies using Multi-agent Sensor Systems. In Luis Seabra Lopez, Filipe Silva, and Vitor Santos, editors, *Proceedings of the 8th Conference on Autonomous Robot Systems and Competition: Robotica 08; Aveiro, Portugal, April 2nd, 2008*, pages 93–98, Aveiro, April 2008. Universidade de Aveiro.

[KBS75] Carolyn Kimme, Dana Ballard, and Jack Sklansky. Finding circles by an array of accumulators. *Commun. ACM*, 18(2):120–122, 1975.

[KBZ+08] Uwe-Philipp Käppeler, Ruben Benkmann, Oliver Zweigle, Reinhard Lafrenz, and Paul Levi. Resolving Inconsistencies in Shared Context Models using Multiagent Systems. In Rüdiger Dillmann and Wolfram Burgard, editors, *Proceedings of the 10th International Conference on Intelligent Autonomous Systems: IAS-10; Baden Baden, Germany, July 23-25, 2008*, pages 93–98, Baden Baden, Juli 2008. Springer-Verlag.

[Low99] David G. Lowe. Object recognition from local scale-invariant features. In *ICCV '99: Proceedings of the International Conference on Computer Vision-Volume 2*, page 1150, Washington, DC, USA, 1999. IEEE Computer Society.

[LSK+08] Reinhard LAFRENZ, Michael SCHANZ, Uwe-Philipp KÄPPELER, Oliver ZWEIGLE, Hamid RAJAIE, Frank SCHREIBER, and Paul LEVI. Evaluating robustness of coupled selection equations using sparse communication. In Burgard et al., editor, *Intelligent Autonomous Systems 10 (IAS-10)*, pages 272–277. IOS Press, 2008.

[LSZ+07] Reinhard Lafrenz, Frank Schreiber, Oliver Zweigle, Michael Schanz, Hamid Rajaie Uwe-Philipp Käppeler, Paul Levi, and Jens Starke. Evaluating coupled selection equations for dynamic task assignment using a behavior framework. *Berns, K. (ed.); Luksch, T. (ed.): Autonome Mobile Systeme 2007*, 1(1), 2007.

[OSBL05] M. Oubbati, M. Schanz, T. Buchheim, and P. Levi. Velocity control of omnidirectional robocup player with recurrent neural networks. *Proceedings of the RoboCup Symposium*, 1(1), 2005.

[OSL05] M. Oubbati, M. Schanz, and P. Levi. Fixed-weight rnn adaptive controller for an omnidirectional robot. *Proceedings of the 9th International Conference on Engineering Applications of Neural Networks (EANN05)*, 1(1), 2005.

[RLS+07] H. Rajaie, R. Lafrenz, F. Schreiber, U.-P. Käppeler, O. Zweigle, T. Rühr, A. Tamke, and P. Levi. Physical simulation of the dynamical behavior of three-wheeled omni-directional robots. In *submitted to: RoboCup Symposium 2007, Atlanta*, 2007.

[RWY95] Daniel Reisfeld, Haim Wolfson, and Yehezkel Yeshurun. Context-free attentional operators: the generalized symmetry transform. *Int. J. Comput. Vision*, 14(2):119–130, 1995.

[SB91] Michael J. Swain and Dana H. Ballard. Color indexing. *Int. J. Comput. Vision*, 7(1):11–32, 1991.

[SSL+05] M. Schanz, J. Starke, R. Lafrenz, O. Zweigle, M. Oubbati, H. Rajaie, F. Schreiber, T. Buchheim, U.-P. Käppeler, and P. Levi. Dynamic Task Assignment in a Team of Agents. In P. Levi et al., editor, *Autonome Mobile Systeme*, pages 11–18. Springer, 2005.

[YPIK90] H. K. Yuen, J. Princen, J. Illingworth, and J. Kittler. Comparative study of hough transform methods for circle finding. *Image Vision Comput.*, 8(1):71–77, 1990.

[ZKR+08] O. Zweigle, U.P. Käppeler, H. Rajaie, K. Häussermann, R. Lafrenz, A.Tamke, F. Schreiber, M. Hoeferlin, M.Schanz, and P. Levi. Cops-team description. In *RoboCup 2008 electronic publication*, 2008.

[ZLB+06] Oliver Zweigle, Reinhard Lafrenz, Thorsten Buchheim, Uwe-Philipp Käppeler, Hamid Rajaie, Frank Schreiber, and Paul Levi. Cooperative Agent Behavior Based on Special Interaction Nets. In Tamio Arai, editor, *Intelligent Autonomous Systems 9 (IAS-9)*, 2006.