AT Humboldt in RoboCup-99 (Team description)*

Hans-Dieter Burkhard, Jan Wendler**, Thomas Meinert, Helmut Myritz, and Gerd Sander

Humboldt University Berlin, Department of Computer Science,
Artificial Intelligence Laboratory, D-10099 Berlin, Germany
email: hdb/wendler/meinert/myritz/sander@informatik.hu-berlin.de,
WWW: http://www.ki.informatik.hu-berlin.de

1 Introduction

Our agent team AT Humboldt 99 (AT stands for "Agent Team") was developed as extension of our former team AT Humboldt 98, which became vice champion at RoboCup-98. We started to extend it by improved skills, new options and a larger planning horizon, respectively. So the most features of our current team were already part of AT Humboldt 98 which has been briefly described in [3] and extensive described in [5]. A description of our first soccer team AT Humboldt 97, which became world champion at RoboCup-97, can be found in [1].

We are interested in virtual soccer for the development and the evaluation of our research topics in artificial intelligence which concern the fields of

- Agent oriented techniques,
- Multi-Agent Systems,
- Case Based Reasoning.

The results of our research in these areas can be found in [1, 2, 4, 7, 8]. Thus many aspects of our soccer program are heavily influenced by these fields, but it is important not to consider these fields in isolation: to create our soccer agents, we also needed a lot of contributions from other fields of computer science (e.g. programming techniques, synchronisation, concurrency) and from mathematics. Thereby we gain deeper insights for integration AI techniques in software development. This aspect is especially important for the education of our students.

2 Team Development

The virtual soccer teams "AT Humboldt" are implemented by our AI group at the Department of Computer Science at the Humboldt University Berlin. The

* This work was partly sponsored by TecInno GmbH Kaiserslautern, Daimler Chrysler AG Research & Technology Berlin and PSI AG Berlin

** This work has been partially supported by the German Research Society, Berlin-Brandenburg Graduate School in Distributed Information Systems (DFG grant no. GRK 316).

M. Veloso, E. Pagello, and H. Kitano (Eds.): RoboCup-99, LNAI 1856, pp. 542–545, 2000.

work is done by groups of students in practical exercises during the summer semester. A core group of up to three students maintains the coordination and the programs. Besides the experiments with AI methods, the project is also a challenge for software development by changing teams. The source code of AT Humboldt 99 has about 28.000 lines of code. It is a non trivial task to maintain the introduction of new ideas during extremly short time intervals by changing teams. To support the concurrent development we use the freely available source code management system CVS [9] and the documentation system doc++ [10].

Team Leader: Prof. Hans-Dieter Burkhard
Team Members:

Prof. Hans-Dieter Burkhard
- leader of the AI group
- did lead the development and did consulting
- did attend the competition

Jan Wendler
- PhD student
- did consulting and some debugging
- did attend the competition

Thomas Meinert, Helmut Myritz and Gerd Sander
- undergraduate student
- did the design, implementation and debugging of the new ideas
- did attend the competition

Web page: http://
www.ki.informatik.hu-berlin.de/RoboCup/RoboCup99/index_e.html

3 World Model

Our world model uses object representations of situations, implemented by a class called Situation. For any given time an object of this class will be generated which consists of object represention for teammates, opponent players, the ball and the agent itself. Flags are only used to determine the own absolute object representation which consists of the absolute player position, speed, body direction and of the relative face direction. With this data the absolute objects representations of the other players and the ball are calculated. So the agent can get a new Situation-object from sensor information. Another way to get a new Situation-object is the simulation of the own actions of the agent, as it is done in the *SoccerServer*. So after every sensor information we have two concurrent Siuation-objects which are merged together by finding correspondig player and using the best information of both situations for the new one.

4 Communication

We have done only small efforts into communication. For the cooperation among the teammates no communication is neccessary because our agents model their teammates and can predict their behaviour very well. The agents only use communication to broadcast their world model among their teammates .

5 Deliberation and Strategy

Our agent architecture uses a mental deliberation structure which is best described by a belief-desire-intention architecture (BDI) [6]. Distinct from other (e.g. logically motivated) approaches our approach is closely related to procedural thinking, and we use object oriented programming for the implementation.

BDI reasoning means to us the rational choice of promising options to become desires and intentions. The main idea of our reasoning process is its decomposition into modular heuristic options and a generic control process.

Our BDI structure uses a set of independent options which can be differentiate into active and passive options. Each option returns a value of its utility based on the current situation of the world and obeying certain constraints as *ConserveStamina* and *AvoidOffside*.

When the agent has control over the ball one of its active options will have the highest utility and therefore he will try to pass to a teammate or to dribble in a preffered direction or to kick a goal. It depends on the utilities of the active options which one of these behaviour is chosen by the agent.

If the agent doesn't have control over the ball the passive options are candidates for the highest utility. The agent may decide to intercept the ball, or try to get a good position to get a pass from a teammate, or go back to his home region, or just collect data if already there.

Precise information about our deliberation structure can be found in our paper "BDI Design Principles and Cooperative Implementation in RoboCup" [7], which can be found in this book.

6 Skills

After a soccer agent has decided what to do, he can use its skills to fullfil its intention. All skills generate a plan which can be of any size up to the fullfilment of the intention. The most important skills are:

Goto Region: With this skill an agent is able to run to a destination region, which is given by a circle. Because we are invoking all skills with a copy of the current situation, obstacles will be avoided exactly. If a stamina limit is set, the agent will never fall below this limit of stamina. This skill can be also used to generate a plan to run backwards.
For the interception of the ball this method is used as well. The determination of the target region is done in the corresponding option.

Kick Ball: Up to now this skill kicks the ball with a demanded power into a demanded direction. Depending on the speed and the direction of the ball, it is kicked several times to reach the demanded values. This skill was implemented by using mathematical calculations.

Dribbling: We have implemented two kinds of dribbling. The simple dribble method just kicks the ball with low power to the destination direction and runs after the ball. In the second implementation, the agent puts the ball beside his body and dribbles forward, never loosing control over the ball.

Whereas the first method is stable against synchronisation problems it lacks at safety over ball control. Just the opposite holds for the second method, so we used the first one in defense and the other one in midfield and attack.

7 Conclusion

In this paper we gave a glimpse of our agent team AT Humboldt 99, which is an extension of our team AT Humboldt 98. More information about our teams can be found in [1–5, 7, 8].

Our main goals for further extensions are: the introduction of longer planning intervals (we already introduced a longer planning interval, but only for the option FreeKick), the extension of the "Emergent cooperation" of our agents by explicitely plannable cooperation, the use of learning methods for ball kicking and dribbling skills, and the modelling of opponents in the frame of the BDI architecture to predict their behaviour in advance.

Furthermore we want to support our sony legged robot team, the Humboldt Hereos, with techniques already used by our simulation team.

References

1. Burkhard, H. D., Hannebauer, M., and Wendler, J.: AT Humboldt — Development, Practice and Theory. RoboCup-97: Robot Soccer World Cup I, Springer 1998, 357–372.
2. Burkhard, H.-D., Hannebauer, M., and Wendler, J.: Belief-Desire-Intention Deliberation in Artificial Soccer. *AI Magazine* 19(3): 87–93. 1998.
3. Gugenberger, P., Wendler, J., Schröter, K., and Burkhard, H. D.: AT Humboldt in RoboCup-98 (Team description). RoboCup-98: Robot Soccer World Cup II, Springer 1999, 358–363.
4. Hannebauer, M., Wendler, J., Gugenberger, P., and Burkhard, H. D.: Emergent Cooperation in a Virtual Soccer Environment. RoboCup Papers at ICRA-98 and DARS-98, 1998, 72–81.
5. Müller-Gugenberger, P. and Wendler, J.: *AT Humboldt 98 — Design, Implementierung und Evaluierung eines Multiagentensystems für den RoboCup-98 mittels einer BDI-Architektur*. Diploma Thesis. Humboldt University Berlin, 1998.
6. Rao, A. S. and Georgeff, M. P. : BDI agents: From theory to practice. In V. Lesser, editor, *Proc. of the First Int. Conf. on Multi-Agent Systems (ICMAS-95))*, pages 312–319. MIT-Press, 1995.
7. Wendler, J., Burkhard, H. D., and Hannebauer, M.: BDI Design Principles and Cooperative Implementation in RoboCup. RoboCup-98: Robot Soccer World Cup III (this book), Springer.
8. Wendler, J. and Lenz, M.: CBR for Dynamic Situation Assessment in an Agent-Oriented Setting. In D. Aha and J. J Daniels, editors, *Proc. AAAI-98 Workshop on Case-Based Reasoning Integrations, Madison, USA, 1998.*
9. CVS: http://www.loria.fr/ molli/cvs-index.html
10. DOC++: http://www.zib.de/Visual/software/doc++/index.html