Getting closer: How Simulation and Humanoid League can benefit from each other

Joschka Boedecker[3], Norbert Michael Mayer[1,2], Masaki Ogino[1], Rodrigo da Silva Guerra[1], Masaaki Kikuchi[1], and Minoru Asada[1,2]

[1]Dept. of Adaptive Machine Systems,
[2]HANDAI Frontier Research Center,
Graduate School of Engineering, Osaka University, Osaka, Japan
{ogino, guerra, kikuchi}@er.ams.eng.osaka-u.ac.jp
{norbert, asada}@ams.eng.osaka-u.ac.jp
[3]AI Research Group, Universität Koblenz-Landau, Koblenz, Germany
jboedeck@uni-koblenz.de

Summary. This paper presents the current efforts and ideas of members in the RoboCup Simulation and the Humanoid Leagues to take successful concepts from both environments and extend them in ways so that each league can profit from the results. We describe the ongoing development of the 3D simulator which is being extended to simulate a real humanoid robot. At the same time, we give an insight into the current behavior development framework of the Humanoid League team *Senchans* which makes heavy use of techniques which have been successfully used in the Simulation League before. Furthermore, we give some suggestions for a collaboration between the different leagues in the RoboCup from which all the participants could benefit.

1 Introduction

One big issue that is under heavy discussion amongst members of the RoboCup Simulation League community is the question to what extent the simulation environment should abstract from the real world robots. During a discussion at the RoboCup-2004 in Lisbon, it became clear that a majority of researchers would like to push the 3D simulation towards simulating humanoid robots. It was considered by many as a necessary step towards the declared goal of the RoboCup initiative [6, 5]: to built a team of fully autonomous humanoid robots able to beat the human soccer champion by the middle of the 21st century. The concern was that too much abstraction in the simulation might produce results that are not easily usable in real robots (cf. [1] for a similar discussion). To avoid this, and to start the development in the outlined direction, it is necessary to look at how both leagues can be brought closer together, and how effective concepts and structures can be reused.

At Osaka University, researchers from both leagues joined forces in order to learn and benefit from each other's knowledge. As a result, the structure of the agent code for the Humanoid League team *Senchans* was revised and a very flexible framework for behavior development which had been successfully applied in the Simulation League was adopted. At the same time, the official 3D simulator used in the Simulation League is currently being extended to simulate the robot used in the *Senchans* team: the HOAP-2 robot from Fujitsu Automation Ltd. (see also Fig. 2(right)). This can be seen as first steps into the desired direction of using humanoid robots in the Simulation League, but it can also serve as a prototype platform for rapid controller and behavior development for the HOAP-2 robot. We will address this issue more closely in section 4.

The organization of this paper is as follows. In the next two sections we give two examples of successful collaborations across two different leagues in RoboCup. First, we give a brief overview of the development status of the 3D simulator and outline the work that is currently being done to improve it. Next, we describe the development of the agent structure used in the *Senchans* humanoid team. Finally, we present ideas for a simulation environment that would be attractive for researchers across different leagues and then give some concluding remarks.

2 Development of the 3D simulator

RoboCup-2004 saw the introduction of the new 3D simulator to the Simulation League competitions. It uses the generic building blocks of the *Spark* simulation system [8], the free *Open Dynamics Engine* (ODE) library [10] for realistic physics, and OpenGL for 3D graphics rendering. Furthermore, the simulation is built on top of the middle-ware system *Spades* which provides means for distributed simulation execution and takes into account machine and network load to guarantee reproducible experiments [9]. Fig. 1(left) depicts a screenshot of the current simulator.

While the simulator was successfully used at the RoboCup in Lisbon, it is still under heavy development. In its current state, the simulator employs a very basic representation for the robots. They are simulated as spheres and only have two actuators: a kicking device and a directional motor for locomotion. The only sensor implemented so far is an omnidirectional camera.

For the reasons given in section 1, we are now trying to implement a model of the HOAP-2 robot, as well as appropriate sensors (encoders for the joint angles, force sensors in the feet, gyroscope, and accelerometer), and actuators (motors for the joints) for the simulator. Similar work [2] has been done for the commercial simulator *Webots* [7] by Pascal Cominoli at the Swiss Institute of Technology in Lousanne, Switzerland (see also Fig. 1(right)). However, as the Webots source code is not freely available and the product is not easily affordable by everybody, it was decided to make the effort of implementing the

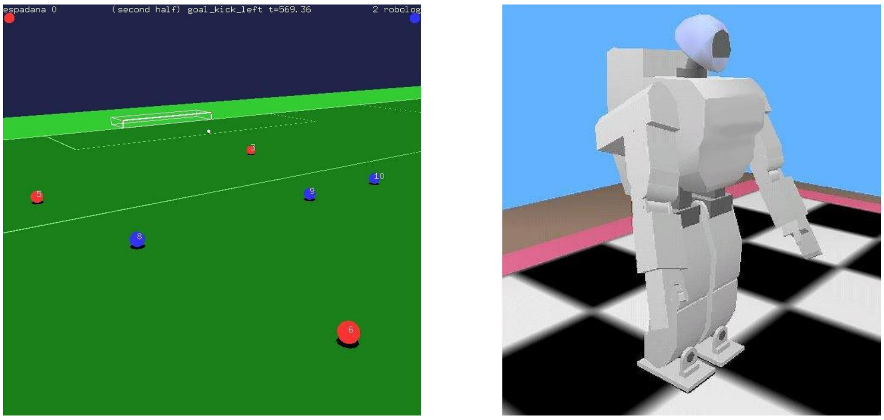

Fig. 1. (left) Screen shot of the current 3D simulation. So far, the agents are represented as spheres and have rather simple sensors and actuators. (right) Simulation of the HOAP-2 robot in the webots simulator

robot simulation in the official RoboCup Simulation League simulator (which is freely available at [4]). Luckily, we can to some extent build on the work[1] in [2]. Once the robot is implemented in the simulation, we plan to introduce an adaption layer to convert between the commands sent to the robot and a more generic command format used in the simulator. This will allow to reuse controllers developed for the real robots in the simulation and vice versa.

3 Transfer of Agent Modeling Knowledge to the Humanoid Robots

As the RoboCup Humanoid League progresses and becomes more sophisticated (the first matches of 2 vs. 2 robots will be played at the RoboCup in Osaka this year), the need for modular and extensible software architectures arises. While the low-level control of the robots is still the prevalent issue in this league, tactics and team strategies will become more and more important. This is why for the software architecture of the *Senchans* team, we adopted an object-oriented design similar to the one presented in [3]. It had been very successfully used in the RoboCup 2D Simulation League before and seemed general enough to be used in a team of humanoid robots, too. The different layers of the architecture provide functions to communicate with the robot, provide high- and mid-level skills like approaching the ball or passing, and implement the action selection for the robot. An overview of the architecture is given in Fig. 2(left). For the sake of clarity, we omitted several classes in the diagram, but the core classes of the system are shown.

[1] Thanks go to Pascal Cominoli for providing us with the Webots file he used in his work

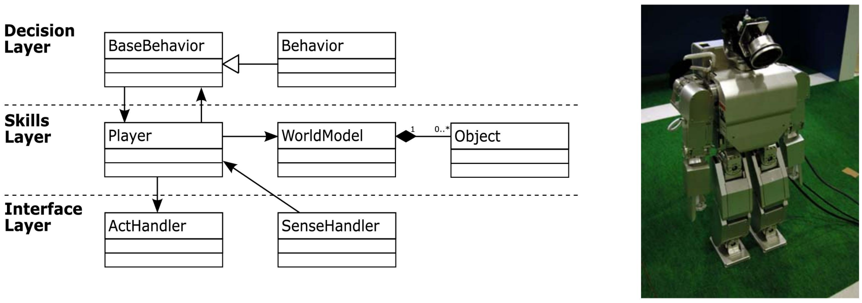

Fig. 2. (left) UML class diagram of the agent architecture used in the Senchans team. It shows the core classes of the system in the different layers. The modular design allows for easy extensibility. All sensory information from and to the robot is handled by the interaction layer. The information stored in structures of the skills layer, which also provides methods for predicting the future world state. These are used by classes in the decision layer to determine the best action of the agent for the current situation. (right) The HOAP-2 robot used in the Senchans team.

The design is based on modules with a high inner cohesion and low coupling between the different modules. The behaviors are implemented as dynamic modules which can be loaded at runtime. Because of this plug-in mechanism, the other classes of the architecture don't need to have any knowledge about the inner workings of the behaviors. The robots' behavior can be changed easily, without having to recompile any code, by simply loading a different behavior module according to the present situation.

4 Building a Simulation Environment for Researchers from Different Leagues

Simulation and the real-world robotics are different in several aspects. Even though there are increasing efforts to build more realistic environments for simulation, there are other central issues that distinguish the two worlds. Among these issues the authors emphasize the following:

- *Simplistic models* – Even for quite realistic simulated scenarios, the boundaries upon which the simulation teams can explore near-singular situations are much broader than it is possible in the real world. Robustness of hardware, for instance, is not simulated yet and is a central concern in the real-robot leagues. Even if some randomness is introduced into the simulated environment, those probabilistic models are bound to, and – for the sake of tractability – to some extent have to be more simple than what can be observed in the physical world. In the real-robot leagues, disturbances are relatively unknown (there is no probabilistic model available), therefore very unpredictable, and most important of all: *they depend on design* (i.e. the used hardware).

- *Diversity of design* – So far, in the case of the Simulation League, the interface of the team to its environment is rather standardized (except for some room for customization using the heterogeneous players in the 2D league). In contrast to this, real-robot leagues have the design of the robot itself as a main research topic. The *environment* is standardized but not the interfaces to it. As real-robot leagues use real hardware, there are several design issues on choosing hardware parts, such as price, efficiency/processing power, weight, size, and so on. These issues are not easily dealt with, and often a game ends up being lost because of a lack of robustness rather than strategy.

In general, researchers within the Simulation League have traditionally focused on *game strategy*, real-robot leagues have a major concern on *robot design and control.* Quite often, simulation team members actually chose to engage in the Simulation League rather than real robots in order to be able to concentrate on strategy and not having to spend a lot of time designing their robots. The reverse statement could also be true, although not very probable, as up to now it is too difficult to reach the simulation level of strategy development with the current hardware technology.

However, one idea that has come up in discussions among members of the different leagues, and which could bring both worlds closer to each other, is to build a kind of *parts and controller repository* within the official simulator. This way, researchers who are interested in design and low-level control of the robots could contribute new parts and test their ideas for controllers, while others who want to focus more on strategies and multi-agent coordination would use those provided parts to build custom robots for the simulation, or even use pre-assembled, standard robot models. Teams would be allowed to develop their own hardware for integration into the simulation, as long as they provide it for all teams within the repository. Of course, these repository items should have constraints about their realism, and perhaps some other tradeoffs could be introduced to prevent teams from having impossible designs. The Simulation League technical committee could make the final decision whether a proposed item is fair enough to be approved. By doing so, both simulation and non-simulation leagues could have a common environment in which they could benefit from each other – new design ideas and strategies from simulation could be more easily taken into the real world and vice-versa.

5 Conclusion

We presented two concrete examples of successful collaborations of members working in different leagues within RoboCup. A lot of what was described is ongoing work, and at this point it is still too early to present an evaluation of the ideas presented. However, we are confident that both leagues will be able to profit from the developments described above. Furthermore, we gave a possible direction for further developments of the simulation environment used in the Simulation League that could attract researchers from different leagues, and provide a tool that could speed up the progress across those leagues.

6 Acknowledgements

This work was partially made possible through financial support from the Heinz Nixdorf Foundation which we gratefully acknowledge here.

References

1. R. A Brooks. Intelligence Without Representation. *Artificial Intelligence Journal*, 47:139–159, 1991.
2. P. Cominoli. Development of a physical simulation of a real humanoid robot. Master's thesis, EPFL, Swiss Federal Institute of Technology, 2005.
3. R. de Boer and J. Kok. The Incremental Development of a Synthetic Multi-Agent System: The UvA Trilearn 2001 Robotic Soccer Simulation Team. Master's thesis, University of Amsterdam, 2002.
4. Soccer Server Maintenance Group. The Robocup Soccer Simulator Webpage. `http://www.sf.net/projects/sserver/`, 2005.
5. H. Kitano and M. Asada. The Robocup humanoid challenge as the millennium challenge for advanced robotics. *Advanced Robotics*, 13(8):723–736, 2000.
6. H. Kitano, M. Asada, Y. Kuniyoshi, I. Noda, E. Osawa, and H. Matsubara. RoboCup: A Challenge AI Problem. *AI Magazine*, 1997.
7. O. Michel. Webots: a Powerful Realistic Mobile Robots Simulator. In *Proceeding of the Second International Workshop on RoboCup*, Lecture Notes in Artificial Intelligence. Springer, 1998.
8. Oliver Obst and Markus Rollmann. SPARK – A Generic Simulator for Physical Multiagent Simulations. In Gabriela Lindemann, Jörg Denzinger, Ingo J. Timm, and Rainer Unland, editors, *Multiagent System Technologies – Proceedings of the MATES 2004*, volume 3187 of *Lecture Notes in Artificial Intelligence*, pages 243–257. Springer, September 2004.
9. Patrick Riley. SPADES: A System for Parallel-Agent, Discrete-Event Simulation. *AI Magazine*, 24(2):41–42, 2003.
10. Russell Smith. ODE webpage. `http://www.ode.org/`, 2005.