In Baltes, Lagoudakis, Naruse, Gidary, editors, *RoboCup 2009: Robot Soccer World Cup XIII*, pp. 140--152, Springer, 2010.

Three Humanoid Soccer Platforms: Comparison and Synthesis

Shivaram Kalyanakrishnan, Todd Hester, Michael Quinlan,
Yinon Bentor and Peter Stone
Department of Computer Sciences, The University of Texas at Austin
{shivaram, todd, mquinlan, yinon, pstone}@cs.utexas.edu

Abstract. In this article, we provide an overview of three humanoid soccer platforms currently in use at RoboCup: 3D simulation, the humanoid Standard Platform League (SPL), and the Webots-based simulator released with the SPL. Although these platforms trace different historical roots, today they share the same robot model, the Aldebaran Nao. Consequently, they face a similar set of challenges, primary among which is the need to develop reliable and robust bipedal locomotion. In this paper, we compare and contrast these platforms, drawing on the experiences of our team, UT Austin Villa, in developing agents for each of them. We identify specific roles for these three platforms in advancing the overarching goals of RoboCup.

1 Introduction

The long-term goal of RoboCup is to field a team of humanoid soccer players that can compete with the best human teams on a regulation soccer field by the year 2050 [11]. This goal is still very far away, in part because we do not yet have "human-level" humanoid robots. Since the start of RoboCup in 1997, however, steady progress has been made on all aspects of the challenge, which has been achieved by planning, prioritizing efforts, and by imagining the technology of a few years ahead. We attribute the ability of RoboCup to maintain its momentum in advancing the frontiers of technology to two main characteristics:

1. Its competitive structure, which fosters enthusiasm in diverse groups from all over the world, and at the same time, encourages collaboration. As Behnke [1] notes, competitions promote the evaluation of complete systems, providing a standardized testbed on which comparisons are fair. This is supplemented by community-based development, brought about by the sharing of ideas, solutions, and organizational responsibilities.
2. The division of effort into well-defined leagues and challenge problems, each with self-contained, specific focus areas that are challenging by themselves. Currently at RoboCup there are five main leagues: Simulation League, Small Size League, Middle Size League, Standard Platform League, and Humanoid League. Within each of these, there are multiple sub-leagues.

The focus of this paper is in the context of the second characteristic: the leagues that constitute RoboCup. Ideally, these various leagues, while addressing separate challenges, should also be carefully connected so as to address complementary and synergistic research challenges. Based on the extensive firsthand experiences gathered by our team, UT Austin Villa, we have the opportunity to closely examine three platforms (two sub-leagues and one supplement) that are very new, are spawned from different threads of history, and yet have much in common. These are the 3D simulation sub-league within the Simulation league (Sim-3D), the humanoid-based Standard Platform League (SPL), and its accompanying Webots-based simulator (SPL-Sim).

These three platforms trace different origins with different research foci, but they are now converging in the need to tackle bipedal humanoid locomotion. Incidentally, these three platforms share the same robot model, the Aldebaran Nao.[1] At this point, it seems that in any platform that has bipedal locomotion, the dominant challenge is that of stable, fast movement: the team that is able to walk the fastest and kick the most accurately is favored to win, with minimal need for sophisticated higher-level reasoning. However, the goal is to reach the point where locomotion can be taken as mainly a skill to be fine-tuned within the larger context of individual and team decision-making in strategic situations.

An important ongoing effort within the RoboCup community over the past few years has been the inclusion of "road-map" discussions at the symposia, which are the culmination of discussions within the individual leagues. However, there are only a limited (though significant) number of participants bringing experiences from multiple leagues to these discussions. This paper offers a focused road-map proposal pertaining to three closely related platforms within RoboCup Sim-3D, SPL and SPL-Sim are all 1-3 years old, with much potential yet to be realized. We compare and contrast the current state of progress in these three platforms and identify specific roles for them in advancing the overarching goals of RoboCup. We present these recommendations from the point of view of keeping the platforms complementary, each with its important role to fill. We recognize that individual communities may have perspectives that are not entirely consistent with our proposals. Yet, we believe that some level of *inter-league* planning is necessary for the future.

This paper is organized as follows. In Section 2 we survey the current state of the three platforms, following which, in Section 3 we describe our experiences in developing agents for each platform. Section 4 lays down some of the long-term challenges facing humanoid robotics, and Section 5 earmarks specific roles to each platform in moving forward. We summarize the paper in Section 6.

2 Overview of Platforms: Past and Present

In this section, we present the history and the current challenges facing each platform. Figure 1 shows snapshots of the Nao robot from the three platforms.

2.1 Sim-3D

Sim-3D has evolved from the 2D Simulation sub-league, one of the earliest competitions in RoboCup [3], having been in existence since 1997. The 2D Simulation sub-league simulates a 2D world, with cylindrical robots that have access to abstract actions such as Turn, Kick, Dash, and Catch. It has models of noisy,

[1] See: http://www.aldebaran-robotics.com/eng/

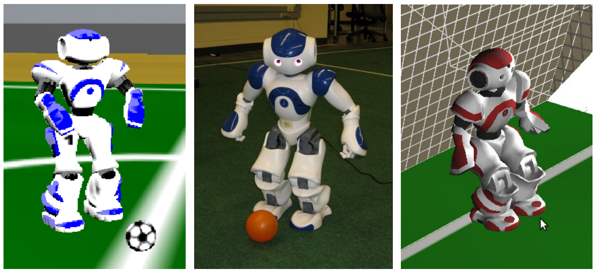

Fig. 1: Pictures of the Nao robot in the various leagues. From left to right: Sim-3D, SPL, and SPL-Sim.

asynchronous sensation and actuation, real-time decision making, restricted vision and hearing, and player stamina. In short, it incorporates an extensive range of realistic considerations in simulation. Stable "11 versus 11" simulations can be run on current desktop hardware, with agents able to communicate with the simulation server through a network interface following a well-defined protocol. While the first few years of the 2D simulation sub-league witnessed an emphasis on developing agent skills (such as reliable passing and ball interception), its focus soon progressed to high-level strategy. Not only do player formations and communication play an important role in games today, teams even employ strategic reasoning for determining whether to play offensively or defensively depending on the goal difference and time left in the game.

Despite all the realism modeled in the 2D simulator, there exists one significant omission: the third dimension. To address this issue, the 3D Simulation sub-league (Sim-3D) was introduced into RoboCup in 2004. The migration to a 3D world called for physical simulation engines to replace the simple physics models of the 2D simulation. With the intent of staying ahead of the hardware leagues and to encourage high-level reasoning, Sim-3D adopted agents in the form of spheres, with access to abstract commands for kicking, turning and dashing. However, it soon became apparent that while interesting in itself, this version of Sim-3D was lacking direct relevance to the long-term goal of playing soccer with humanoid robots. Indeed in their 2007 paper, Mayer *et al.* [13] argued that the simulation league should embrace humanoid robots as early as possible. To this end, in 2007, Sim-3D transitioned to a more realistic and challenging humanoid model: the Soccerbot, based on the Fujitsu HOAP-2 robot [2]. This step marked a defining moment in the history of the Simulation League: for the first time, robots had to be programmed through the low-level interface of controlling motor torques. While this has caused inevitable backtracking in terms of the performance levels of games, its long-term benefits will be significant.

In the initial days of development, the Soccerbot-based simulation encountered a string of systems-related problems, such as unstable physics simulation and performance issues. To correct this, the Soccerbot model was constantly changed, and by the 2007 RoboCup competitions, it was 5 meters tall! However, subsequent contributions from the community towards developing the server have increased its reliability, as has the change of the robot model from Soccer-

[2] See: http://jp.fujitsu.com/group/automation/en/services/humanoid-robot/hoap2/.

bot to the Aldebaran Nao for the 2008 competitions. In the 2008 competitions, complete, noise-free world information was provided to the agents. In addition, there was no actuator noise. As a result, there has been a quick development of locomotion skills. Both in 2007 and in 2008, skills (in particular, walking speed) have been a major factor determining team performance. It must be noted, however, that some passing behavior began to emerge during RoboCup 2008.

The 3D simulator is under active development. Earlier the platform used the SPADES timer [21], and it still uses the SPARK simulation engine [17]: both of these were developed by the RoboCup community. The simulation server code is fully open source. Apart from the annual RoboCup competitions, Sim-3D is now also popular at the regional open competitions in Iran, Germany, and China.

2.2 SPL

The Standard Platform League is unique in that all the teams use the same standardized hardware, making it essentially a *software* competition. SPL allows teams to work with an affordable humanoid robot platform without having to invest as much time or money as the other humanoid leagues [2]. The current SPL sub-league (using humanoids) emerged from the Sony Aibo "Four-legged" league. Areas of research in SPL include vision [9, 22], localization [6, 7], and motion and skill development [12]. Since the robots are fully autonomous and all processing is performed on-board, CPU cycles need to be used efficiently.

In 2006, Sony stopped manufacturing the quadrupedal Aibo robot; consequently, the SPL switched to the Aldebaran Nao for 2008. The Nao is a two-legged humanoid robot, on which balance, walking and other soccer skills are more difficult to implement than on the Aibo. Teams competing in the SPL served as beta testers for the Nao; the original robots were very fragile and motors would frequently overheat or break. The robot also was unable to see its own feet with its camera without bending over, which made teams spend significant amounts of time searching for and lining up to the ball.

The Nao has 25 degrees of freedom, compared to the 20 on the Aibo. The Nao robot, a biped, stands 57cm tall, while the Aibo, a quadruped, measures 28cm tall. The V3 version of the Nao has two color cameras in its head, each with a much higher resolution than the Aibo. The Nao has a 500 MHz AMD Geode processor and 256 MB of RAM, which is more sophisticated than what the Aibo uses (a 576 MHz RISC CPU and 64 MB of RAM).

The first competition with the Nao robot was held at RoboCup 2008 in Suzhou, China. Since the robots broke frequently, many teams arrived at the competition with very little code tested on actual robots. In Suzhou, robots were fixed as they broke, giving some teams their first opportunity to test code on the robots. It was a struggle for teams to even "close the loop" on robot behaviors, i.e., reliably score a goal on an empty field. There were only 14 goals scored in total in the 29 games during RoboCup 2008 (including one own goal), even though most teams had very minimal defense strategies. In stark contrast, the Aibo competition the same year witnessed a total of 151 goals scored in 28 competitive games.

2.3 SPL-Sim

Two companies released Nao models in their respective simulators after the SPL humanoid sub-league was introduced. These were Microsoft Robotics Studio [3] and Cyberbotics Webots [15]. Both simulators are more sophisticated than Sim-3D, using commercial physics software and providing useful tools for programming the agent, such as for visualization and debugging. The simulators aid code development for the physical robots in the SPL, proving especially useful in making initial progress in developing skills, as the robot itself is quite fragile.

For code development on the Nao, UT Austin Villa employed the Webots simulator, which we denote SPL-Sim. The simulator was helpful in developing code for SPL, especially since the robots were unavailable or unusable for long durations. While the physics were not very realistic, and therefore the learned motions were not directly transferable to the robot, the simulator was useful in testing vision, localization and behaviors. The simulator was also useful for learning parameters for motions, which made good starting points for motions on the physical robot. Although there is no official competition at RoboCup using Webots, there was an Internet-based competition called "ROBOTSTADIUM" [4], as well as an informal competition at RoboCup 2008. About eight teams played 4 versus 4 soccer games, exhibiting superior skills compared to those from the SPL games. Versions of Webots with the Nao model were made available free to the teams. UT Austin Villa did not take part in this Webots-based competition.

3 Experiences in Agent Development

The agent architectures that UT Austin Villa developed were similar across the three platforms. At the low level, we have PID control for the joints, as well as inverse kinematics for arms and legs. The main thrust of our effort was in developing skills, such as walk, kick, turn, and get-up. These in turn were tied together by fairly simple high-level behavior. Here we summarize the salient aspects of our agent behavior, which is described in detail in our technical reports [5, 10].

For SPL, we developed a walk that comprises four key-frame states, inspired by Yin *et al.* [24]. Each state is represented by the spatial coordinates of the feet relative to the hips. Joint angles are calculated from these spatial coordinates using inverse kinematics. The motion between states is interpolated and the transition to the next state begins either at a specific time or after getting close to the current target state. The specific coordinates of the key-frames are determined through a set of 10 walk parameters, such as step height and length. When turning, the "HipYawPitch" angle is set to turn the robot's legs. In addition, the robot's shoulder joints are moved to help balance the robot during walking. We used the Downhill Simplex algorithm [19] to learn the best walk parameters through trials in the Webots simulator (SPL-Sim).

While the walk we developed worked well in simulation, it did not work consistently on the physical robots, likely because of lag and jitter introduced

[3] See: http://msdn.microsoft.com/en-us/robotics/.

[4] See: http://www.robotstadium.org/.

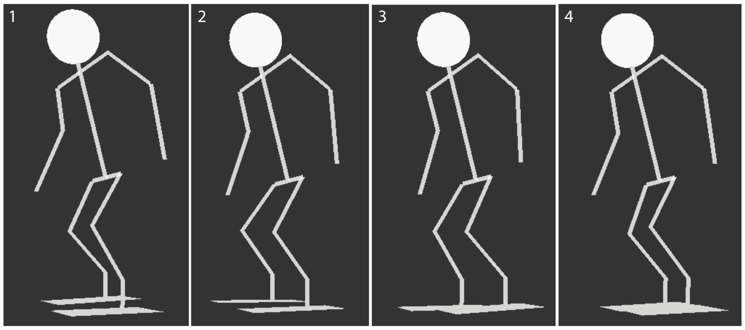

Fig. 2: States from our SPL(-Sim) walk engine. In the state 1, the robot shifts its weight to its left leg and lifts its right leg. In state 2, it rebalances its weight and brings the right leg forward and down. In states 3 and 4, the robot repeats this motion with the legs reversed.

by the interface we used to send commands to the robot. Our Sim-3D agent too employed a similar state-based approach for developing skills, although with vastly different parameter settings. Specifically, Sim-3D used higher gains for motor torques in order to realize fast, dynamically stable motions.

We summarize the different walking speeds and kick distances that we achieved on the different platforms in Table 1.[5] The results on the different platforms are not directly comparable, as they were obtained on different surfaces, with different parameters, different physics, different code, etc. We used our own code for all the Sim-3D skills and for the kicks in SPL(-Sim). We used the walk engine provided by Aldebaran with parameters that we tuned for the walking experiments in SPL(-Sim). For comparison, the robot walked at 28.56 ± 1.31 mm/sec in SPL-Sim using our walk engine. The results in SPL and SPL-Sim were taken over 5 trials each, and the ones from Sim-3D over 10 trials.[6]

Table 1: Performance statistics for skills of the Nao robot on different platforms.

Skill	Sim-3D	SPL	SPL-Sim
Forward walking: linear velocity (mm/sec)	144.27 ± 1.22	82.40 ± 1.54	91.62 ± 2.77
Side walking: linear velocity (mm/sec)	62.80 ± 2.00	18.13 ± 2.66	24.48 ± 2.23
Turning: angular velocity (deg/sec)	19.96 ± 3.15	27.20 ± 0.54	18.83 ± 0.21
Kicking: distance reached by ball after kick (mm)	3122.68 ± 14.47	1200.00 ± 312.64	5122.12 ± 756.78
Get-up: time to rise after falling forwards (sec)	10.21 ± 0.94	10.56 ± 0.27	NA
Get-up: time to rise after falling backwards (sec)	23.14 ± 0.81	11.30 ± 0.37	NA

[5] The authors acknowledge Hugo Picado from the FC Portugal robot soccer team for suggesting this set of relevant statistics.

[6] Videos of several skills are available from our team website: http://www.cs.utexas.edu/~AustinVilla/

There is a significant difference between the results from SPL-Sim and SPL, as we switch from simulation to reality. For example, although the same kick was used on the real robot and in SPL-Sim, the ball traveled an average of about four times as far in the simulator because it has less friction than the real carpet. The main difference between the Sim-3D and SPL-Sim is in the magnitude of the gains. Interestingly, the real robot turns much faster than either simulated robot. The turn is implemented by moving the HipYawPitch joint to turn the legs relative to each other. This is more effective on the real robot than in simulation because the extra friction arrests sliding of the feet. We note that the reported statistics are not representative of all soccer teams; for example, some Sim-3D teams obtained very fast walks through optimization. Yet, these results showcase some key differences in the platforms and the behaviors developed on them.

The key element in our architecture for SPL(-Sim) was to enforce that the environment *interface*, the agent's *memory* and its *logic* were kept distinct (Figure 3). In this case, *logic* encompasses the vision, localization, behavior and motion modules. The main advantages of our architecture are:

Consistency The *core system* remains identical irrespective of whether the code is run on the robot, in the simulator or inside our debug tool. As a result, we can test and debug code in any of the three environments without fear of code discrepancies. The robot, simulator and tools each have their own *interface* class which is responsible for populating *memory*.

Flexibility We can "plug & play" modules into our system by allowing each module to maintain its own local memory and communicate to other modules using the common memory area. For example, a Kalman Filter localization module would read the output of the vision module from common memory, work in its own local memory and then write object locations back to common memory.

Debug-ability At every time step only the contents of current *memory* are required to make logic decisions. We can therefore save a "snapshot" of the current memory to a log file and then examine the log subsequently in our debug tool. The debug tool not only has the ability to read and display logs, it can also take logs and process them through the logic modules. As a result we can modify code and watch the full impact of the change in our debug tool before testing it on the robot or in the simulator.

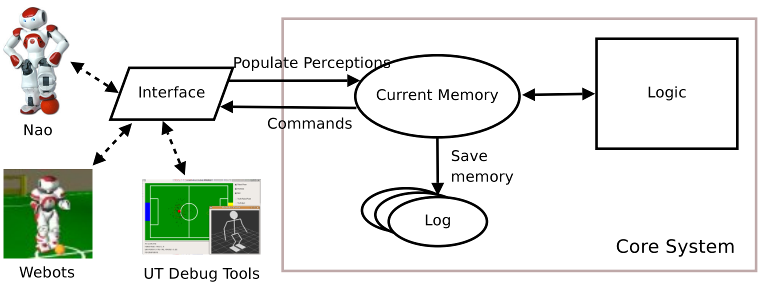

Fig. 3: Agent architecture for SPL(-Sim).

4 Shared Challenges

Sim-3D, SPL, and SPL-Sim are all just coming out of their nascency. In this section, we discuss some of the challenges facing humanoid robotics, which should be treated as long-term issues on all these platforms.

The key concern at this stage for all three platforms is also the most basic: reliable, robust bipedal locomotion. Locomotion is a prerequisite for a large number of activities performed by a humanoid robot, and its role is particularly important in soccer. The RoboCup community has successfully addressed the challenge of quadrupedal locomotion [12], but bipedal locomotion requires solutions of a more complex nature and scale. Bipedal locomotion offers great flexibility in developing a wide range of stepping patterns, including climbing stairs and lateral motion. Yet, this flexibility comes at the price of constantly having to balance. While balance is almost like a second nature to human beings, the state-of-the-art with humanoid robotics is yet to provide satisfactory solutions to this problem [23].

Pratt's thesis [18] provides useful insights into the nature of bipedal locomotion. Whereas human beings rely extensively on the natural dynamics of walking (uncontrolled "falling" for some part of the gait, followed by a swift catching step), most algorithms for walking today control every part of the walk cycle. This places a severe limit on the robustness of the developed walking gait: even slight variations in the cycle can potentially cause fall. Another fallout is the energy efficiency of walking. Fully controlled walks spend roughly an order of magnitude more energy for walking than human beings with comparable masses [20]. Thus, developing robust, energy efficient walks is of utmost importance for any humanoid platform in RoboCup. Approaches such as passive dynamic walking [14] are likely to play a key role in realizing "natural" patterns of bipedal locomotion, and need to be considered in the future.

Another challenge facing humanoid robotics is the need to develop humanlike models. Currently, the body components of the Nao robot in Sim-3D are implemented as cuboidal elements: this hinders the development of natural gaits as the feet are flat. Simulating complex mesh geometries can be time inefficient; at the same time, they might be *necessary* for developing robust locomotion skills. Another related possibility is to explore the use of more human-like actuators to augment the motors available at the joints. For example, Hosada *et al.* [8] have demonstrated that pneumatic artificial muscles can lead to gaits that require minimal correction. Likewise, it is relevant to note that the Nao (and most humanoid robots) do not have a flexible spine. Flexible spines such as the ones being developed by Mizuuchi *et al.* [16] allow the robot to have more natural movement and enable it to absorb shocks much better. Indeed a major problem in developing walks for SPL is ensuring that the robot's feet touch the ground softly and do not create oscillations in the robot's movement.

In the following section, we argue that in coming to terms with the multiple challenges ahead, the three platforms considered in this paper should concentrate on separate issues, in order that their *combination* will be most effective in advancing towards the goals of RoboCup.

5 Roles for Platforms in Future

As Mayer *et al.* [13] observe, it will become important to start integrating the various leagues in RoboCup; in the year 2050, one "league" will stand to summarize the progress made over the years. At this stage, humanoid robotics still faces a broad range of problems, chief among which is reliable bipedal locomotion. Yet, there is the need to look ahead and preempt problems that are bound to arise in future. In this section, we identify specific roles for Sim-3D, SPL, and SPL-Sim such that they divide their attention in doing so. Encouraging diversity in the platforms is important at this juncture because it will alert us to a wider range of issues, which would remain occluded if our focus remains narrow. However, we note that the desire for diversity needs to be balanced with serving the interests of the members of the individual leagues.

There is the need to study problems in hardware because simulation is invariably subject to unrealistic modeling assumptions, and of course, the ultimate goal of RoboCup needs to be realized in hardware. However, hardware has the disadvantage of being brittle, expensive, and tedious to work with. Thus, it becomes efficient to bypass it and make progress in simulation environments. For example, simulation is far more convenient as a platform for learning and optimization of skills and behaviors. Ultimately, simulation needs to become as close to reality as possible. As long as simplifying assumptions have to be made by simulators, we argue that Sim-3D and SPL-Sim should maintain separate foci: Sim-3D should stay ahead of current possibilities of hardware, and SPL-Sim should become as close to reality as possible. We also believe that each platform should continue to get gradually more difficult as earlier challenges are met.

5.1 Sim-3D

We believe that as with the 2D simulation sub-league that spawned it, Sim-3D should stay ahead of the hardware by a few years and realize what is currently possible *only in simulation*, such as sophisticated coordination strategies among the players and complex soccer skills such as heading, kicks in the air, and the interception of balls in a 3D trajectory. These aspects do not manifest in today's hardware platforms, but it would be inadvisable to wait for the hardware to catch up before devoting research to these issues, which are integral to soccer. Such research might guide the evolution of the hardware itself.

Among the current simulation platforms, Sim-3D has developed the best robotic skills to date, which are evolving at a steady rate. Also Sim-3D has the support of a very active development community. Thus, we propose that Sim-3D consciously make the effort to tackle increasingly complex skills and behavior that humanoid robots will ultimately possess. In so doing, we need to exercise intelligence in choosing parts of reality to approximate and parts to model exactly. For example, we do not need to develop perfectly realistic models of the cameras of today, for it is very likely that in future, robots will be equipped with superior vision systems. However, it would be unwise to persist with very high joint torque limits because these are unrealistic, and will remain so until the year 2050.

5.2 SPL

Unlike Sim-3D, it is not possible to look very deep into the future of SPL, because SPL is inherently restricted by the limits of hardware. At this point, the main focus of the SPL is on important low-level tasks such as vision, localization, and bipedal control. The SPL humanoid sub-league is more challenging than the SPL four-legged (Aibo) sub-league owing to the difficulties involved in developing robust motions on two legs. For the short term, it will be quite an achievement to be able to reproduce the proficiency level of the Aibo robots on the Nao robots. This would entail developing algorithms to for walking stably, robustly, and fast; kicking optimally; ball interception, etc. It might not be worthwhile to plan beyond such a level of proficiency for the Nao robots because they may undergo fundamental changes to the hardware in a few years' time.

5.3 SPL-Sim

Simulators are essential in modern scientific research, as it is time-consuming, expensive, and tedious to conduct several types of useful experiments in the real world. Ultimately, it will be beneficial to have simulators that are as close to reality as possible, for they can function as a more convenient substitute. We propose that at least one thread of research within RoboCup pursue the goal of developing accurate simulators. The simulator being developed need not be specific to the SPL; any hardware league can profit from the use of a realistic physical simulator. But it is especially important to have simulators of humanoid robots, which are going to be principal in the future development of RoboCup.

We believe that SPL-Sim should adopt the aim of becoming more realistic, partly because it possesses superior systems performance at this stage, which make it the more promising alternative for modeling complex mesh geometries (including feet), simulating realistic collisions (which are common in soccer), and modeling surface properties. Currently Webots already has a reliable and accurate physics engine, developed based on the Open Dynamics Engine (ODE).[7]

Gaps between simulation and reality need to be closed; likely, they will get more pronounced when there are more collisions and robots move faster. Recently, Hebbel and Laue [4] have proposed an interesting idea for doing so: by optimizing simulator parameters based on a fitness function that is evaluated in reality. They demonstrate significant results on the Aibo platform. It would be ideal if efforts such as theirs are complemented by code development dedicated to realizing more realistic simulations for RoboCup. It will also be good for the community to develop open source packages. We could start from the current 3D simulator, SPARK, and evolve distinct threads for Sim-3D and SPL-Sim.

6 Summary

UT Austin Villa's participation in Sim-3D and SPL, as well as our use of the SPL-Sim to develop our code gives us a unique perspective on these platforms. There

[7] See: http://www.ode.org/.

are many similar challenges on these platforms: developing a good code base that is easily debug-able and extendable, developing good robust bipedal motions, and creating good soccer skills. There are differences as well; for instance, SPL(-Sim) has to cope with vision and localization issues, while Sim-3D currently does not.

For RoboCup to continue its progress towards the goal of fielding a human-level team in 2050, we believe that it is important to continue with different leagues that focus on separate aspects of the problem. SPL should continue to deal with problems related to using a real two-legged robot, such as vision, localization, and bipedal motion. We believe that there should be a push towards more realistic simulators to allow more development in simulation instead of on fragile and expensive robots. Finally, the Sim-3D should continue to focus on problems a few years ahead of the hardware leagues, such as coordination, teamwork, intelligence, and developing skills on advanced robot models. All the platforms should continue to gradually increase in difficulty as teams progress.

Acknowledgments

The authors thank anonymous reviewers for their comments. This work has taken place in the Learning Agents Research Group (LARG) at the Artificial Intelligence Laboratory, The University of Texas at Austin. LARG research is supported in part by grants from the National Science Foundation (CNS-0615104, EIA-0303609 and IIS-0237699), DARPA (FA8750-05-2-0283, FA-8650-08-C-7812 and HR0011-04-1-0035), General Motors, and the Federal Highway Administration (DTFH61-07-H-00030).

References

1. S. Behnke. Robot competitions - ideal benchmarks for robotics research. In *Proc. of IROS-2006 Workshop on Benchmarks in Robotics Research*, October 2006.
2. S. Behnke, M. Schreiber, J. Stückler, R. Renner, and H. Strasdat. See, walk, and kick: Humanoid robots start to play soccer. In *Proc. of IEEE-RAS Int. Conf. on Humanoid Robots (Humanoids'06)*, December 2006.
3. M. Chen, E. Foroughi, F. Heintz, Z. Huang, S. Kapetanakis, K. Kostiadis, J. Kummeneje, I. Noda, O. Obst, P. Riley, T. Steffens, Y. Wang, and X. Yin. Users manual: RoboCup soccer server — for soccer server version 7.07 and later. *The RoboCup Federation*, August 2002.
4. M. Hebbel and T. Laue. Automatic parameter optimization for a dynamic robot simulation. In *RoboCup 2008: Robot Soccer World Cup XII*, Lecture Notes in Artificial Intelligence. Springer, 2009. To appear.
5. T. Hester, M. Quinlan, and P. Stone. UT Austin Villa 2008: Standing on Two Legs. Technical Report UT-AI-TR-08-8, The University of Texas at Austin, Department of Computer Sciences, AI Laboratory, November 2008.
6. T. Hester and P. Stone. Negative information and line observations for monte carlo localization. In *Proc. of IEEE International Conference on Robotics and Automation (ICRA)*, pages 2764 – 2769, May 2008.
7. J. Hoffmann, M. Spranger, D. Göhring, and M. Jüngel. Negative information and proprioception in monte carlo self-localization for a 4-legged robots. In *Proc. of*

Nineteenth International Joint Conference on Artificial Intelligence, Workshop on Agents in Real-Time and Dynamic Environments, 2005.

8. K. Hosada, T. Takuma, A. Nakamoto, and S. Hayashi. Biped robot design powered by antagonistic pneumatic actuators for multi-modal locomotion. *Robotics and Autonous Systems*, 56(1):46–53, January 2008.
9. M. Jüngel. Using layered color precision for a self-calibrating vision system. In D. Nardi, M. Riedmiller, and C. Sammut, editors, *RoboCup 2004: Robot Soccer World Cup VIII*, volume 3276 of *Lecture Notes in Artificial Intelligence*, pages 209–220. Springer Berlin / Heidelberg, 2005.
10. S. Kalyanakrishnan, Y. Bentor, and P. Stone. The UT Austin Villa 3D Simulation Soccer Team 2008. Technical Report AI09-01, The University of Texas at Austin, Department of Computer Sciences, AI Laboratory, February 2009.
11. H. Kitano, M. Asada, Y. Kuniyoshi, I. Noda, and E. Osawa. RoboCup: The robot world cup initiative. In *Proceedings of The First International Conference on Autonomous Agents*, pages 340–347. ACM Press, 1997.
12. N. Kohl and P. Stone. Machine learning for fast quadrupedal locomotion. In *The Nineteenth National Conf. on Art. Intelligence*, pages 611–616, July 2004.
13. N. M. Mayer, J. Boedecker, R. da Silva Guerra, O. Obst, and M. Asada. 3D2Real: Simulation league finals in real robots. In G. Lakemeyer, E. Sklar, D. G. Sorrenti, and T. Takahashi, editors, *Proceedings of RoboCup Symposium*, volume 4434 of *Lecture Notes in Artificial Intelligence*, pages 25–34, Berlin, Heidelberg, 2007. Springer.
14. T. McGeer. Passive dynamic walking. *Int. J. Rob. Res.*, 9(2):62–82, 1990.
15. O. Michel. Webots: Professional mobile robot simulation. *Journal of Advanced Robotics Systems*, 1(1):39–42, 2004.
16. I. Mizuuchi, S. Yoshiada, M. Inaba, and H. Inoue. The development and control of a flexible-spine for a human-form robot. *Advanced Robotics*, 17(2):179–196, 2003.
17. O. Obst and M. Rollmann. SPARK – A Generic Simulator for Physical Multiagent Simulations. *Computer Systems Science and Engineering*, 20(5):347–356, September 2005.
18. J. Pratt. *Exploiting Inherent Robustness and Natural Dynamics in the Control of Bipedal Walking Robots*. PhD thesis, Massachusetts Institute of Tech., 2000.
19. W. Press, S. Teukolsky, W. Vetterling, and B. Flannery. *Numerical Recipes in C*. Cambridge University Press, Cambridge, UK, 2nd edition, 1992.
20. S. Ramamoorthy and B. Kuipers. Trajectory generation for dynamic bipedal walking through qualitative model based manifold learning. In *IEEE International Conference on Robotics and Automation*, pages 359–366, 2008.
21. P. Riley and G. Riley. SPADES — a distributed agent simulation environment with software-in-the-loop execution. In S. Chick, P. J. Sánchez, D. Ferrin, and D. J. Morrice, editors, *Winter Simulation Conference Proceedings*, volume 1, pages 817–825, 2003.
22. C. J. Seysener, C. L. Murch, and R. H. Middleton. Extensions to object recognition in the four-legged league. In D. Nardi, M. Riedmiller, and C. Sammut, editors, *RoboCup 2004: Robot Soccer World Cup VIII*, volume 3276 of *Lecture Notes in Artificial Intelligence*, pages 274–285. Springer Berlin / Heidelberg, 2005.
23. P.-B. Wieber. Viability and predictive control for safe locomotion. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, September 22-26, 2008, Acropolis Convention Center, Nice, France*, pages 1103–1108, 2008.
24. K. Yin, K. Loken, and M. van de Panne. Simbicon: Simple biped locomotion control. *ACM Trans. Graph.*, 26(3):Article 105, 2007.