Darmstadt Dribblers

Team Description for Humanoid KidSize League of RoboCup 2009

Martin Friedmann, Karen Petersen, Sebastian Petters, Katayon Radkhah, Dorian Scholz, Dirk Thomas, and Oskar von Stryk

Department of Computer Science, Technische Universität Darmstadt,
Hochschulstr. 10, D-64289 Darmstadt, Germany
E-Mail: dribblers@sim.tu-darmstadt.de
Web: www.dribblers.de

Abstract. This paper describes the hardware and software design of the kidsize humanoid robot systems of the Darmstadt Dribblers in 2009. The robots are used as a vehicle for research in control of locomotion and behavior of autonomous humanoid robots and robot teams with many degrees of freedom and many actuated joints. The Humanoid League of RoboCup provides an ideal testbed for such aspects of dynamics in motion and autonomous behavior as the problem of generating and maintaining statically or dynamically stable bipedal locomotion is predominant for all types of vision guided motions during a soccer game. A modular software architecture as well as further technologies have been developed for efficient and effective implementation and test of modules for sensing, planning, behavior, and actions of humanoid robots.

1 Introduction

The RoboCup scenario of soccer playing legged robots represents an extraordinary challenge for the design, control and stability of autonomous bipedal and quadrupedal robots. In a game, fast, goal oriented motions must be planned autonomously and implemented online while preserving the robot's postural stability and adapting them in real-time to the quickly changing environment.

The Darmstadt Dribblers participated in 2004 as the first German team in a soccer competition of the RoboCup Humanoid League (penalty kick) where they reached the semi-final. In RoboCup 2006 the Dribblers reached the 3rd place in the 2-2 games out of 16 teams. In 2007 and 2008 the quarter finals out of 20, resp. 24 teams were reached in the 3-3 games and lost both times in tight games against the later champion. In the technical challenges the Dribblers reached the 4th place in 2007 and the 2nd place in 2008 where they were the only team that completed the passing challenge.

The information processing of the robot's sense-plan-act cycles are realized in two, resp. three layers, cf. Sect. 4. In RoboCup 2009 the Darmstadt Dribblers participate in the Humanoid KidSize League with further enhanced hardware and software based on the achievements of previous years.

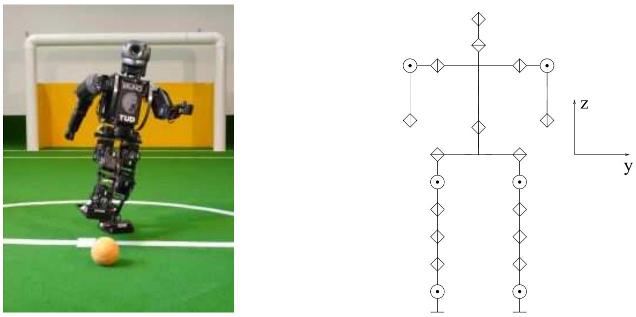

Fig. 1. Autonomous humanoid robot *Bruno* kicking a ball (left), and kinematical structure of the robots (right).

2 Research Overview

The research of the Darmstadt Dribblers in humanoid robotics focuses on

- online-optimization for fast and stable humanoid locomotion, e.g. [1–3],
- alternative humanoid arm and leg kinematics using bio-inspired, elastic kinematics [4, 5] or artificial muscles [6],
- modular, flexible and reusable software and control architectures for cooperating and possibly heterogeneous robot teams [7, 8],
- clocked, hierarchical finite state automata for programming high-level behavior of autonomous robots and robot teams [1, 9],
- modeling, simulation and optimal control of the full nonlinear dynamics of motion of humanoid and four-legged robots [10, 2],
- a real-time software- and hardware-in-the-loop environment simulating humanoid robot kinematics and dynamics as well as external and internal robot sensors for evaluating any onboard software used for image interpretation and perception, localization and control of a humanoid robot [11, 12],
- humanoid perception using an articulated, directed camera mounted on a pan-tilt-joint as well as acoustic communication and localization [13].

3 Hardware

In 2009 a further improved robot design of the model DD2008 will be used by the Darmstadt Dribblers. The kinematic structure with 21 DOF can be seen in Fig. 1. The robots are equipped with an articulated camera and distributed computing hardware, consisting of a controller-board for motion-generation and stability control and an embedded PC board for all other functions. For motion stabilization 3 1D-gyroscopes and a 3-axes-accelerometer are used.

4 Software

In the current robotic system the computational power for information processing is distributed into basically two, respective three, layers. The lowest, third layer of computation is performed within the 21 servo motors. Each servo motor is equipped with some "intelligence" consisting of adjustable controllers for the joint's position and velocity which operate at a constant rate of about 1 ms (estimate). The motors are also able to monitor their operational environment, e.g. temperature of the motor as well as voltage of the power supply, thus allowing autonomous emergency shutdown in case of overheating motors or discharged batteries. Further hard real-time tasks like motion generation and stability control are executed on a microcontroller board (reflex layer). High level control like vision, world modeling, behavior control and team coordination is executed on a standard embedded PC board (cognitive layer). Both parts of the control software communicate by a serial connection.

The development process for the software is supported by several tools including a graphical user interface (GUI) and a real-time simulation of the robots which can be used to transparently replace a real robot for software-in-the-loop (SIL) tests of the software.

4.1 Low-Level Control Software (Reflex Layer)

The main task of the low-level control software is motion control including the generation of stable walking motions in real-time. To ensure real-time performance it is executed on a microcontroller board allowing a 10 ms control cycle. Motion generation is based on an inverse kinematics model of the 6 DoF robot's legs. For each time-step the pose of the robot's feet and hip is calculated and respective angles for the leg joints are calculated. The basic trajectories of hip and feet are based on ZMP theory and can be parameterized and altered at runtime [1]. Stability control is based on the robot's gyroscopes. Readings of the gyros are used to generate balancing motions, e.g. with the robot's arms, and to calculate offset angles for the leg joints to compensate for disturbances detected by the inertial sensors.

The walking engine's parameters (e.g. different length and time variations during one stride) are well suited for optimization. By applying a newly developed optimization method a maximum walking speed of 40 cm/s in permanent operation was achieved [3]. From the accelerometer the robot detects if it has fallen down and to which side. The robot can stand up autonomously from lying on its back or its front side. The low-level control software also includes several hardware related drivers and a main control function which is executed at the robot's control rate. For software-in-the-loop testing the control function can be re-compiled to a DLL which can be executed within the Darmstadt Dribblers' multi-robot simulator [11].

4.2 High-level Control Software (Cognitive Layer)

RoboFrame. The base of the robot control software is the object oriented and platform independent framework *RoboFrame* (`www.dribblers.de/roboframe`). This robot middleware has been developed to match the special requirements in small sized light-weight robots, both legged and wheeled, with low payload abilities resulting from requirements for dynamical and inertially stabilized locomotion. The framework provides flexible communication connections between the data processing parts of the applications, the so called modules. Currently packet and shared memory based communication is possible. The connections are established during runtime with very little overhead, thus allowing to change the layout of the application very fast. Very different deliberative or reactive behavior control paradigms may be realized on the basis of RoboFrame which has already been employed successfully on a variety of robots with different locomotion and onboard computing properties.

For debugging and monitoring of the software, a graphical user interface based on the platform independent GUI toolkit QT is available. With the GUI it is possible to visualize any kind of data by extending the provided API. TCP based data connections to multiple robots are possible. For further details on the architecture, the framework and the modules see, e.g. [7, 8, 14]

Current modules. Mainly four interacting modules developed on the basis of RoboFrame are used for the Dribblers's humanoid robots: image procession, world modeling, behavior decision and motion control.

Image processing. To achieve a modular and extendable vision system for different camera types, the vision module can process images in different color spaces with different resolutions by choosing a highly object oriented approach which allows rapid prototyping of new image processors while providing the possibility for code optimizations for high computational efficiency. Image processing is split into two parts: a common pre-processing stage and several exchangeable modules for object recognition. Object recognition, done by so called perceptors, can work with multiple image types, such as pre-processed segmented or gray scale images, or the unprocessed raw image. This way, depending on the object and underlying recognition algorithms, the proper level of abstraction can be used by each perceptor while keeping the pre-processing efforts at the required minimum. The perceptors developed up to now detect field lines, line crossings, the center circle, the ball, goals, poles and obstacles.

World modeling. The world model consists of a set of models which are updated by different modellings using the detected percepts from the vision module. One part of the world model is a self localization, which is accomplished by Markov localization with particle filtering [15]. Additional for nearly every percepted object a modeling exists, for example the ball and the obstacles. A selected subset of informations from the models is exchanged between all robots in the scenario via wireless LAN using an UDP broadcast. These informations are integrated into the various models, for example if no ball is seen by a robot, it uses the ball position communicated by its team players to start its own ball search. Additionally these informations are used in a role model to dynamically selected the different player roles of the field players.

Behavior decision. The data provided by the world model is used to plan a more complex behavior such as it is required for playing soccer autonomously. The main task is separated into subtasks until they can be described as a set of atomic actions which can be executed by the humanoid robot. This is done by a hierarchical state machine implemented in XABSL [9]. The basic motion actions are transferred to and interpreted by the motion module, other basic actions are processed in further modules.

Motion control. The current motion module is mainly used to calculate walking trajectories (see Sect. 4.1) and to control the neck joints with two DOF depending on the robot type. The control of the other joints in the arms is mainly for balancing aspects during walking or kicking.

4.3 Simulation

Developing and testing the key modules of autonomous humanoid soccer robots (e.g., for vision, localization, and behavior control) in software-in-the-loop (SIL) experiments, requires real-time simulation of the main motion and sensing properties. These include humanoid robot kinematics and dynamics, the interaction with the environment, and sensor simulation, especially the camera properties. To deal with an increasing number of humanoid robots per team the simulation algorithms must be very efficient. The simulator framework `MuRoSimF` (Multi-Robot-Simulation-Framework, `www.dribblers.de/murosimf`) has been developed which allows the flexible and transparent integration of different simulation algorithms with the same robot model. These include several efficient algorithms for simulation of humanoid robot motion kinematics and dynamics (with $O(n)$ runtime complexity), collision handling, and camera simulation including lens distortion. A simulator for teams of humanoid robots based on `MuRoSimF` has been developed [11, 16, 12]. A unique feature of this simulator is the scalability of the level of detail and complexity of motion and sensor simulation which can be chosen individually for each simulated robot and tailored to the requirements of a specific SIL test. Currently up to ten humanoid robots with 21 degrees of freedom, each equipped with an articulated camera can be simulated in real-time on a moderate laptop computer.

Acknowledgement. The team *Darmstadt Dribblers* currently consists of students and researchers of the Technische Universität Darmstadt, namely Armin Berres, Astrid Wolff, Barbara Pfister, Dirk Thomas, Dorian Scholz, Georg Stoll, Jochen Mück, Karen Petersen, Katayon Radkhah, Martin Friedmann, Sebastian Jakob, Sebastian Petters, Stefan Kohlbrecher, Thilo Molitor, Thomas Hemker, and Oskar von Stryk.

Further information (including preprints of publications as well as videos) is available online for download from our website `www.dribblers.de`.

References

1. M. Friedmann, J. Kiener, S. Petters, H. Sakamoto, D. Thomas, and O. von Stryk. Versatile, high-quality motions and behavior control of humanoid robots. *International Journal of Humanoid Robotics*, 5(3):417–436, September 2008.
2. M. Hardt and O. von Stryk. Dynamic modeling in the simulation, optimization and control of legged robots. *ZAMM: Zeitschrift f. Angewandte Mathematik und Mechanik*, 83:648–662, 2003.

3. T. Hemker, H. Sakamoto, M. Stelzer, and O. von Stryk. Efficient walking speed optimization of a humanoid robot. *International Journal of Robotics Research*, 28:303 – 314, Feb 2009.
4. S. Klug, O. von Stryk, and B. Mhl. Design and control mechanisms for a 3 dof bionic manipulator. In *Proc. 1st IEEE / RAS-EMBS Intl. Conf. on Biomedical Robotics and Biomechatronics (BioRob)*, number 210, Pisa, Italy, Feb. 20-22 2006.
5. A. Seyfarth, R. Tausch, M. Stelzer, F. Iida, A. Karguth, and O. von Stryk. Towards bipedal jogging as a natural result for optimizing walking speed for passively compliant three-segmented legs. *International Journal of Robotics Research*, page to appear, 2009.
6. R. Kratz, S. Klug, M. Stelzer, and O. von Stryk. Biologically inspired reflex based stabilization control of a humanoid robot with artificial SMA muscles. In *Proc. IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 1089–1094, Kunming, China, December 17-20 2006.
7. M. Friedmann, J. Kiener, S. Petters, D. Thomas, and O. von Stryk. Reusable architecture and tools for teams of lightweight heterogeneous robots. In *Proc. 1st IFAC Workshop Multivehicle Systems*, pages 51–56, Salvador, BR, Oct. 2-3 2006.
8. S. Petters, D. Thomas, and O. von Stryk. Roboframe - a modular software framework for lightweight autonomous robots. In *Proc. Workshop on Measures and Procedures for the Evaluation of Robot Architectures and Middleware of the 2007 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, San Diego, CA, USA, Oct. 29 2007.
9. M. Lötzsch, M. Risler, and M. Jüngel. XABSL - a pragmatic approach to behavior engineering. In *Proceedings of IEEE/RSJ International Conference of Intelligent Robots and Systems (IROS)*, pages 5124–5129, Beijing, China, 2006.
10. M. Hardt and O. von Stryk. The role of motion dynamics in the design, control and stability of bipedal and quadrupedal robots. In G.A. Kaminka, P.U. Lima, and R. Rojas, editors, *RoboCup 2002 International Symposium (Robot Soccer World Cup VI)*, volume 2752 of *Lecture Notes in Artificial Intelligence*, pages 206–223, Fukuoka, Japan, June 24-25 2003. Springer-Verlag.
11. M. Friedmann, K. Petersen, and O. von Stryk. Tailored real-time simulation for teams of humanoid robots. In U. Visser, F. Ribeiro, T. Ohashi, and F. Dellaert, editors, *RoboCup 2007: Robot Soccer World Cup XI*, volume 5001 of *Lecture Notes in Computer Science/AI*, pages 425–432, Berlin/Heidelberg, 2008. Springer.
12. M. Friedmann, K. Petersen, and O. von Stryk. Adequate motion simulation and collision detection for soccer playing humanoid robots. *Robotics and Autonomous System Journal (Elsevier)*, 2009. to appear.
13. D. Becker and M. Risler. Mutual localization in a team of autonomous robots using acoustic robot detection. In *RoboCup International Symposium*, Suzhou, China, July 15-18 2008.
14. S. Petters, D. Thomas, M. Friedmann, and O. von Stryk. Mutlilevel testing of control software for teams of autonomous mobile robots. In S. Carpin et al., editor, *Simulation, Modeling and Programming for Autonomous Robots (SIMPAR 2008)*, number 5325 in Lecture Notes in Artificial Intelligence, pages 183–194. Springer, November 2008.
15. S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics.* MIT Press, 2005.
16. M. Friedmann, K. Petersen, and O. von Stryk. Simulation of multi-robot teams with flexible level of detail. In S. Carpin et al., editor, *Simulation, Modeling and Programming for Autonomous Robots (SIMPAR 2008)*, number 5325 in Lecture Notes in Artificial Intelligence, pages 29–40, Venice, Italy, November 2008. Springer. Best paper award.