HULKs - Team Research Report 2014

Finn Poppinga, Florian Bergmann, Stefan Kaufmann
roboting@tuhh.de
http://www.hulks.de

Hamburg University of Technology

January 6, 2015

Abstract. This paper describes the decisions and development of the code that was used by the team HULKs in 2014's RoboCup World Championship. We talk in detail about the approaches we chose to accomplish the technical challenges and perform the soccer games as well as the drop-in challenges.

1 Introduction

1.1 The Team

The SPL-Team HULKs formed in April 2013 and thus is one of the newer teams in the Standard Platform League (SPL). The team itself is a department of the supporting association, the RobotING@TUHH e.V. which has currently more than fifty members, most of them being undergraduate students. The members are split into four divisions, Brain, Motion, Vision and Marketing. Each division has its own area of responsibility and is supervised by a respective team leader who coordinates the development process.
We participated in the RoboCup World Championship for the first time in 2014 and took part in the RoboCup German Open 2014 to prepare for the World Championship. As expected, most of the more experienced teams were much better than us in playing soccer and as a result we have been eliminated in the preliminary games.
Our team also participated in the *Technical Challenges* (see section 4), where we have been much more successful. The HULKs presented their *FallManager* which was developed for the RoboCup 2014 [1]. This and the good performance in the *Sound Recognition Challenge* made us come out third in the overall ranking of the challenges and thus pre-qualified us for the RoboCup World Championship 2015.

2 HULKs Codebase

The HULKs codebase for the RoboCup 2014 is completely self-developed by the members and former members of the team. We follow this policy of code development, because we think it is important that the team has a good knowledge of how and why the codebase was implemented as it is. This ensures that we can fully understand what is going on in our code without the need to reverse-engineer complex algorithms or intelligent design that was done by other teams. On the other hand, RoboCup 2014 has shown, that there is a lot of work to be done by the HULKs developers to close the gap between our and the other teams' football skills.

2.1 Brain

Behaviour. As stated in the team description paper [1], the HULKs behaviour is based on state machines developed with the `boost::msm` state machine engine in C++. As of 2014 the behaviour was developed on a very high level, i.e. every player role was implemented as a single state machine, only reusing a small number of lower-complexity sub state machines.
The behaviour that controls the head and the general gaming behaviour are split into two subsequent state machines that run independently. This enables us to execute different actions that only need to control the head, e.g. searching for the ball, localizing at the same time as the respective body behaviour controls the overall behaviour.
To switch between different behaviours, a so-called *State-Machine-Switcher* was designed, that can change the currently executed behaviour dynamically and very fast at runtime. This concept gave us the possibility of reducing the complexity of the state machines and instead build a behaviour on top of it that switches into more flexible and reusable sub state machines on the fly. However, this technique was not used in the RoboCup 2014 as any implementation of it was nowhere near done.

Communication. Our team uses the standardized SPL-Message without any appended data to communicate. At the moment we do not really use this data for other reasons than debugging. It is planned to use the SPL-Message for in-team-communication for the RoboCup 2015.

2.2 Motion

Walking. We have been investigating the two most common approaches for implementing a robust and fast walking algorithm for biped robots which are the 3 Dimensional Linear Inverted Pendulum Mode (3DLIPM) [2] and the cart-table model [3]. Both methods show advantages and disadvantages, and we developed a hybrid models, which takes the advantages of both models. The 3DLIPM does not require knowledge of the Zero Moment Point (ZMP) [4] as well as knowledge of the future pattern and is used for the lateral motions. The cart-table model is used to ensure stability for sagittal motions. Our current implementation is a fully adaptive walk which uses sensor feedback from the joint position sensors as well as from the IMU.

The current implementation is mainly based on the control algorithms proposed in [5] where the point in time for a support change is calculated by a filtered estimation of the pendulums position and velocity in y-direction (see Figure 1). Since the proposed method only takes the y-direction of the model as a basis for changing the support foot, this can lead to very large step sizes in x-direction when the robot is pushed. Hence, we developed alternative methods

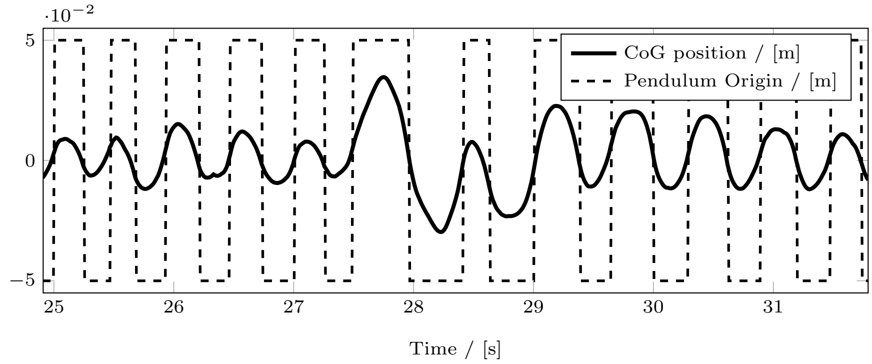

Figure 1: Adjustment of support change time in case of pushes.

for computing the step sizes to stabilize the robot when being pushed. Since the approach in [5] suggests to simply follow the natural movement of the pendulum model, the ability of applying torques to actively influence the pendulum motion is neglected. Therefore we tried to actively apply torques within the foot which has shown to significantly improve the walking stability.

StandUp and FallManager. We developed a very fast key frame based stand up algorithm, that enables our robots to get back in the game in approximately 4 seconds. This is a very common approach in the SPL – however our StandUp-Module works together with our *FallManager* (see 4.1) and enables us to recover from the safe pose, the *FallManager* generates when stopping the robot from falling.
In the future we are going to extend the number of poses the *FallManager* can use to protect the robot from falling and implement an adaptive algorithm to detect such poses and stand up from those in a especially designed manner.

2.3 Vision

Cameras. We have been working on directly accessing the cameras and getting the images over Video4Linux instead of using the module that is provided by Aldebaran. This allows for more control over the settings of the camera. In addition, we are now able to detach our own software further from NAOqi which gives us more flexibility and robustness.

New Concept. After having gathered some experience, we realised that our Vision needs some major changes. We used to give each Module (like the BallFinder) a copy of an image. This is a very inefficient process and we did not use the knowledge that was gained by other modules for later algorithms. Therefore, we decided to rewrite the whole code and use a different concept. We are currently in the process of doing that. Our new concept enables us to use the knowledge of where the field ends for finding the ball in a smaller part of the image. In addition, we are not checking every pixel any more but instead only look at regions of the image that are of interest.
The first version of our new concept is planned to be used in the GermanOpen 2015.

3 DropIn Challenge

3.1 Strategy

Our this year's drop in player was selected after evaluating the performance of our normal player roles in the first preliminary games. It showed, that the so called *SupportStriker* performed best. The strategy of the *SupportStriker* is, to place at an intelligent position in the opponent's half of the field. It then patrols in an area around 1.5m in front of the opponent's goal area and waits for the ball. If it sees the ball, the patrolling will be stopped and the robot tries to score a goal.

3.2 SPL Standard Message

We create SPL Standard Messages as specified in the respective rules during DropIn matches. During the competition we developed a JavaScript based visualizer which enables us to see the communication that is broadcasted by our team and display all information in a graphical user interface utilizing a web-browser of choice.

Our evaluation showed, that not all of the teams are sending either correct SPL Standard Messages or correct data in those messages. At the point of the competition we had no plausibility checks and decided to drop all incoming messages, i.e. playing the game quasi-alone.

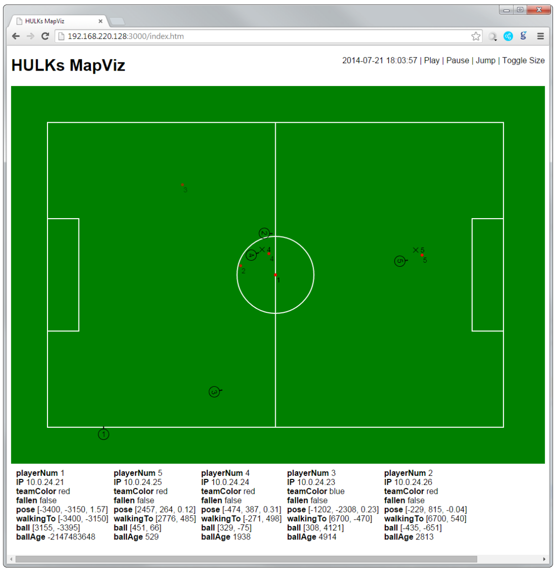

Figure 2: HULKs SPL-Message Viewer

4 Technical Challenges

4.1 Open Challenge

In the *Open Challenge* we presented our *FallManager*. This is a module that utilizes filtered sensor data to detect when the robot is falling. The algorithm

uses a ring-buffer system, that accumulates orientation data of the robot's torso for a number of samples. From the vector difference of theses angular data one can then determine the approximate direction of the robots swing for each individual sample.
If the robot's torso-orientation leaves a specified pyramid, the *FallManager* recognizes this and initiates a motion that is intended to protect the robot from falling with the direction of the fall given as a parameter. As of the RoboCup 2014 the falling motion was not dependant on the determined falling direction. We plan to research this topic in a lot more detail for RoboCup 2015, also discussing control strategies and dynamic modeling of the robot in this field.

4.2 Sound Recognition Challenge

As opposed to the technique described in our team description paper [1], we went for a simpler approach as the supplied audio data was not very complex and it was not required to decide between the two given signals. We instead performed a Power Spectral Density (PSD) analysis and searched for a rise in power at the frequencies related to the given audio sample.

4.3 Any Place Challenge

Due to the amount of work that was needed to be done before the beginning of the RoboCup World Championship 2014, there was no time to prepare the code for the *Any Place Challenge* prior to the competition. A hastely implemented goal-recognition was the base of the robot behaviour. We would find any goal posts in the image and shoot in the direction of any found post.
It turned out, that our ball-recognition was quite stable even in the environment that was showed to us in this challenge and our main difficulty was to be able to locomote on the carpet ground in that specific place.

References

[1] S. Kaufmann, O. Tretau, N. O. Lüders, A. Wurl, F. Patschkowski, N. Riebesel, P. Eisele, J. Nissen, and R. Oehlmann, "HULKs-SPL Team Description Paper 2014," *The 18th annual RoboCup International Symposium*, 2014.

[2] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, and H. H. K. Yokoi, "A realtime pattern generator for biped walking," in *IEEE International Conference on Robotics & Automation*, May 2002.

[3] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa, "Biped walking pattern generation by using preview control of zero-moment point," in *International Conference on Robotics & Automation*, September, 14-19 2003.

[4] M. Vukobratović and B. Borovac, "Zero-moment-point - thirty five years of its life," *International Journal of Humanoid Robotics*, vol. 1, no. 1, pp. 157–173, 2004.

[5] C. Graf and T. Röfer, "A center of mass observing 3d-lipm gait for the robocup standard platform league humanoid," in *RoboCup 2011: Robot Soccer World Cup XV* (T. Röfer, N. Meyer, J. Savage, and U. Saranli, eds.), 2011.