

# Robot Sports Team Description Paper

Ton Peijnenburg<sup>1</sup> (ed.), Jürge van Eijck<sup>2</sup>, et. al.

<sup>1</sup>VDL Enabling Technologies Group (VDL ETG), Achtseweg Noord 5, 5651 GG Eindhoven, The Netherlands, <sup>2</sup>Independent Freelancer, The Netherlands

[Ton.Peijnenburg@vdletg.com](mailto:Ton.Peijnenburg@vdletg.com), [Jurge@barbatrick.nl](mailto:Jurge@barbatrick.nl)

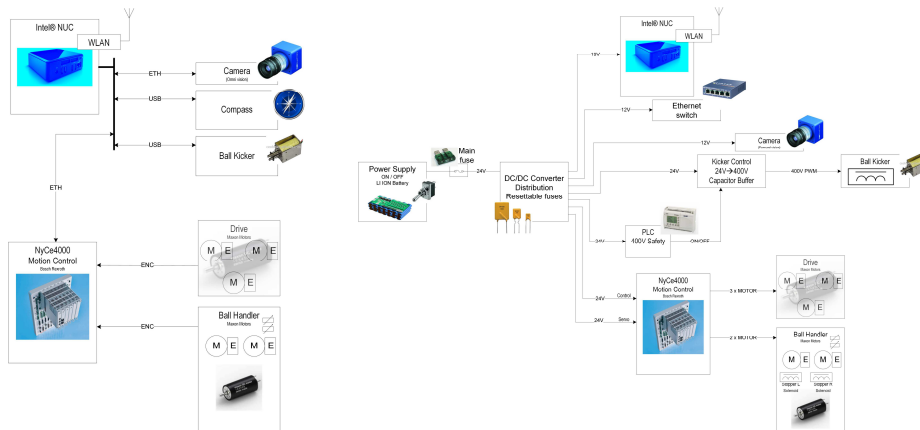
**Abstract.** Robot Sports is a restart of the Philips RoboCup Team, which participated in the MSL competition from 2002 till 2006, when it achieved a best qualification of 3<sup>rd</sup> place during the world championships. Like the Philips RoboCup team, it is an open industrial team, meaning that its participants are all employed by or have retired from various high-tech companies in the Dutch high-tech ecosystem. The team will participate with the intent to develop additional skills that must be added to traditional engineering practices for high-end mechatronic equipment to develop autonomous robotic systems, or teams of autonomous robotic systems. Robot development practices from the domain of Artificial Intelligence in turn may be used to improve high-end equipment development effectiveness and efficiency, e.g. the robustness of equipment control.

## 1. Introduction

The Robot Sports Team uses new robots, developed as a mix of the Philips robot design used in the MSL competition [1], design advancements developed by the Philips team after the last tournament participation and the Tech United TURTLE robot design as used during the 2012 tournaments [2].

## 2. Player architecture

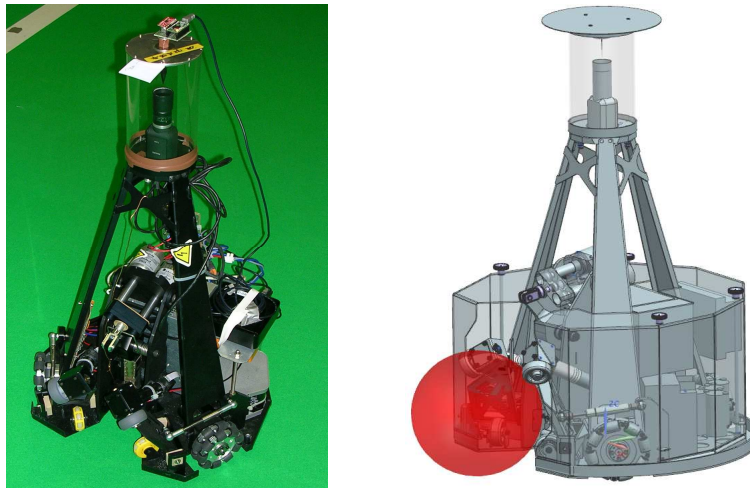
An important break with previous architectures used by our team is that we will no longer use a system wide controller with a real time operating system. Instead we rely on clock synchronization and explicit time parameterized communication. This allows us to use non-real-time components such as out of the box Linux on an x86 processor and Ethernet. Only the motion control tasks still need strict real time behavior; these will be handled by a separate industrial motion controller.



**Fig1.** Robot architecture with main building blocks, data and power flows

## 2.1 Frame

The robot frame is designed entirely in sheet aluminum, which provides rigidity at the lowest possible weight, while keeping cost down. Box structures are used to provide maximum torsional stiffness, e.g. for mounting the kicker mechanism.



**Fig2.** Robot mechanical layout around sheet aluminum frame

## **2.2 Motion and ball handler control**

The robots use the NYCe4000 motion control system to control and power the robot's three drive motors. The system is a modular industrial solution that can be used to control multi-axis system. The drive wheels are not placed symmetrically, but the two front drive wheels are at a smaller angle such that the forward drive force is increased.

The NYCe4000 system also controls the ball handling actuators and the kicker. Being pioneered by the Philips RoboCup Team [3], the ball control mechanism and the powerful kicker have been perfected by the Tech United team, supporting their outstanding results in the world championships for the last decade.

## **2.3 Kicking device 2014**

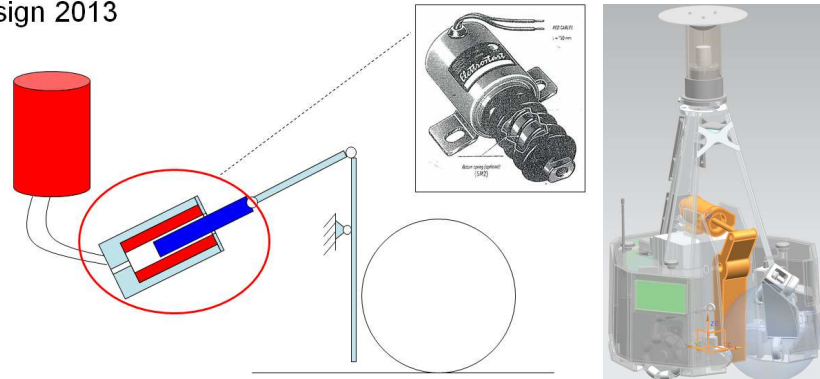
During the period 2002 – 2005 the kicker used a mechanical spring that was compressed by a spindle and released on command. Although this mechanism was innovative at the time and contributed significantly to the successes of the team in those years, electromagnetic kicking mechanisms have become more widely used over time. Advantages are:

- Smaller volume claim
- Shorter recovery time after shooting
- Lower cost of goods
- Better control of shooting power

In our new kicker, automotive solenoids are used for actuation of an adjustable lever that is the "foot" that will kick the ball. Multiple geometries for this "foot" have been tried in an experimental approach, supported by FEA calculations modeling the "foot"-ball interaction. The lever adjustment allows for three discrete vertical positions of the "foot" to vary between low and elevated shots.

A capacitor stack is charged by a circuit originating from a Philips consumer electronics product; discharge is done through a custom IGBT based switch that can be pulse modulated to control shooting power and –duration. The shoot control is implemented on a Teensy 3.1 microcontroller that interacts with the general purpose PC.

design 2013



**Fig3.** A schematic and a physical representation of the new shooting device

## 2.4 Sensing

In our current robot, a catadioptric sensor will be used. The mirror profile will not be a hyperbolic one, but optimized to maximize covered field area on the image sensor. The images are acquired with a GigE camera from Point Grey with a high resolution and framerates up to 60 fps.

Processing of the video stream is done using primitives from the `OPENCV` library and executed on the system's general purpose PC, an Intel NUC with an i5 processor. The primitives are modified for optimal execution speed; at the moment we achieve  $\sim 20$  fps on a full image of  $1280 \times 1024$ .

The main processing is based on non-linear filtering of the image data followed by data reduction steps. To resolve north south playfield ambiguity, a compass unit is used. Self-localization is done based on a Hough transform that operates on white-line points filtered from the video stream. Compass orientation information is used to steer the selection of the Hough lines. Position data is subsequently fused with an encoder based odometer and data from the inertial measurement unit. Ball detection is based on a color segmentation process with filtering on size and position in field coordinates. Balls close by will get priority over ball detections far away.

Obstacle detection and/or competitor detections are filtered based on shape descriptors like area and elongation. Obstacles inside the playing field and close by will get priority over other obstacle detections.

Clustering of features is achieved using proven algorithms from Tech United, both for ball tracking and player tracking. These algorithms have been imported on C code level and their parameters are currently being re-tuned for our specific sensor characteristics.

## **2.5 Shared world model**

The Robot Sports Team uses RTDB [4, 5] to exchange and synchronize data between team players, which results in a fast and accurate shared world model.

## **2.6 Reasoning**

We believe that the reasoning that is required for soccer is much more reactive than the typical AI tasks of the past. Therefore we do not plan to use a lot of processor time to come up with the best possible response for a given situation, but instead invest in behavior that more quickly makes a non-optimized but appropriate decision. This is a tradeoff between timing and quality. In equipment design, all equipment motion is planned ahead of time and tightly controlled to adhere to planned trajectories. Also, collision avoidance is done ahead of time. The soccer robots will be more reactive, which is one of the key interests of the Robot Sports team, i.e. to see how new robot control approaches may benefit equipment control situations. In turn, this will allow equipment designers to work with different design paradigms and optimize robotic equipment for new generations of soccer robots.

The robot behavior is implemented in a set of executable skills. These skills have dedicated responsibilities and effectively run parallel to each other. A domain specific language called STACTICS has been developed already in the days of the Philips team to describe the robot state behavior. The name STACTICS comes from *state* and *tactics*. It allows to express robot behavior in a higher language. Therefore it is not necessary to know C or C++. The STACTICS compiler generates a state list file, which is run-time configure the finite state machine on the robot. This language and finite state machine was used in all the games of the former Philips RoboCup Team.

STACTICS consists of two types of skills: action and decision skills. At the highest level, a finite state machine (FSM) is present with fixed transitions between states. It controls the highest-level states of the robot. The STACTICS

FSM is quite similar to Maes' action selection dynamics [6,7]. The FSM decides when and which transition is made. When a transition is made the set of skills that are relevant for that state are made active. The STACTICS language and machine are now extended to allow for simultaneous execution of skills in an asynchronous manner.

Our robot planner is a variation of the visibility graph [8], which was used on the first general purpose mobile robot Shakey [9], fitted for the soccer domain. On the edges of the created graph (robot planner) by the visibility graph heuristic functions can be added. Via this mechanism opponents can be avoided, while keeping distance to the field boundaries. Restricting the edges to the target vertex and extra costs the approach ball can be influenced. Also the robot's own velocity vector can be taken into account. Via constraint based optimization the best path is determined.

The result of the robot planner is a list of (x,y) points. This describes a rough path. This rough path is smoothed in the smoother step. The smoother splits the path in many small vectors, and adds the velocity and orientation to the path. To limit calculation time, only a limited part of the path is smoothed. The smoothed path can be executed by the motion system of the robot. In an alternative solution, we rely on the robot motion system to provide the smoothing of the path.

We are using a heuristic based team planner, which uses the robot planner to calculate for every available player a path to an objective, till no players are available. The team planner combines dynamic role assignment and strategic positioning. Our decentralized team planner makes the next level of strategic game-play possible. It allows to execute different strategies.

### **3. Research goals**

Robot Sports is an industrial team. Its research goals are tied to the industrial activities of its members. Team members come from a variety of companies that are active in high-end equipment and –modules for markets such as semiconductor electronics, healthcare diagnostic equipment and analytical instruments. It is expected by all team members that developments in autonomous systems in general and the RoboCup initiative more specifically, will find their way to industrial applications sooner or later.

The development of high-end equipment currently is mostly based on a design paradigm that uses a pre-defined nominal behavior description to be executed by the equipment. Provisions for dealing with disturbances and exceptions are added (later) to make the equipment robust. Failure Mode and Effect Analysis (FMEA) techniques are used to identify up-front the exceptional situations that may occur. By strictly control of the environment, the magnitude and impact of disturbances and exceptions are minimized.

As a contrast, in RoboCup new and different design paradigms are used. There is no nominal behavior that a functional design process can use as a starting point, rather a single global objective: to win the game. Team performance is determined by emergent behavior, and disturbances and exceptions are a fact of life, to be dealt with in a gracefully degrading manner.

One of the goals of the Robot Sports team is now to investigate how the differences between the design paradigms can support each other. We want to investigate methods that reduce the sensitivity of the system to the uncertain environment. Being able to create systems that act predictably in uncertain environments is essential for RoboCup and is a desirable answer for many industrial equipment design tasks as well.

An observation when looking at achievements in the RoboCup initiative, is that realistic soccer matches are played in the Simulation League, whereas the hardware leagues evolve but struggle with hardware related limitations. It would be of value to port agents, which have been trained in the Simulation League, onto the hardware players in the Middle Size League. The Robot Sports team has made preparations to be controlled by such agents.

#### **4. Support**

The Robot Sports team is supported by the following companies: VDL Enabling Technologies Group, Maxon Motor Benelux, Bosch Rexroth and HQ Pack. Participants come from these and other companies. The team maintains two locations in Veldhoven and Almelo, both in the Netherlands. The Veldhoven location is shared with the ASML Falcons team.

## 5. Conclusion

Although the RoboCup community is dominated strongly by university teams, with a significant participation from the AI community. There are a number of good reasons for an industrially based team to participate in RoboCup. Our Robot Sports team will bring its collective background of decades of equipment engineering know-how and will strive to find the best match between the findings of modern AI research and new development paradigms from multi-agent system development.

The RoboCup challenge combined with a team of industrial design engineers with a variety of backgrounds and experience, provides for an excellent innovative environment. Industrial applications for autonomous systems are still limited but expected to grow, just like the use of domain-novel AI technologies in industrial equipment. In the Robot Sports team, industrial design engineers can become familiar with new design approaches and AI technologies, and be prepared to apply them in their applications.

## 6. References

- 1 Philips RoboCup Team Description Paper,
- 2 <http://www.roboticopenplatform.org/wiki/TURTLE>
- 3 A.T.A. Peijnenburg, T.P.H. Warmerdam et.al.: Philips CFT RoboCup Team Description. In: preliminary proceedings 2002 RoboCup conference, July 2002
- 4 Santos, F. , Almeida, L., Pedreiras, P. ; Lopes, L.S.: A real-time distributed software infrastructure for cooperating mobile autonomous robots. In Proceedings of 14<sup>th</sup> IEEE International Conference on Advanced Robotics, Munich, Germany (2009)
- 5 RTDB website: <https://code.google.com/p/rtdb/>
- 6 P. Maes: Artificial Life Meets Entertainment: Interacting with Lifelike Autonomous Agents. In: Comm. ACM, Vol. 38, No. 11, Nov. 1995, pp. 108-114.
- 7 P. Maes: A bottom-up mechanism for behavior selection in an artificial creature, 1991
- 8 Lozano-Pérez, Tomás; Wesley, Michael A. (1979), "An algorithm for planning collision-free paths among polyhedral obstacles", Communications of the ACM 22 (10): 560–570, doi:10.1145/359156.359164
- 9 <http://www.ai.sri.com/shakey/>



