

RoboCup Logistics League Sponsored by Festo: A Competitive Factory Automation Testbed

Tim Niemueller¹, Daniel Ewert², Sebastian Reuter²,
Alexander Ferrein³, Sabina Jeschke², and Gerhard Lakemeyer¹

¹ Knowledge-based Systems Group, RWTH Aachen University, Germany
{niemueller, gerhard}@kbsg.rwth-aachen.de

² Institute Cluster IMA/ZLW & IFU, RWTH Aachen University, Germany
[firstname.lastname]@ima-zlw-ifu.rwth-aachen.de

³ Electrical Engineering Department, Aachen Univ. of Appl. Sc., Germany
ferrein@fh-aachen.de

Abstract. A new trend in automation is to deploy so-called cyber-physical systems (CPS) which combine computation with physical processes. The novel RoboCup Logistics League Sponsored by Festo (LLSF) aims at a such CPS logistic scenarios in an automation setting. A team of robots has to produce products from a number of semi-finished products which they have to machine during the game. Different production plans are possible and the robots need to recycle scrap byproducts. This way, the LLSF is a very interesting league offering a number of challenging research questions for planning, coordination, or communication in an application-driven scenario. In this paper, we outline the objectives of the LLSF and present steps for developing the league further towards a benchmark for logistics scenarios for CPS. As a major milestone we present the new automated referee system which helps in governing the game play as well as keeping track of the scored points in a very complex factory scenario.

1 Introduction

A new trend in automation is to deploy so-called cyber-physical systems (CPS) to larger extents. These systems combine computation with physical processes. They include embedded computers and networks which monitor and control the physical processes and have a wide range of applications in assisted living, advanced automotive systems, energy conservation, environmental and critical infrastructure control, or manufacturing [1]. One application area of CPS are logistics scenarios in automation settings. As production is going to move away from mass production towards customized products, the challenges for the automation process will increase. This will open the floor to mobile robots in order to help with the manufacturing process.

In particular, mobile robots will be deployed for transportation tasks, where they have to get semi-finished products in place to be machined in time. This is right where the novel *RoboCup Logistics League Sponsored by Festo* (LLSF)



Fig. 1. A LLSF competition during the RoboCup 2012 in Mexico City

starts. Teams of robots have to transport semi-finished products from machine to machine in order to produce some final product according to some production plan. Machines can break down, products may have inferior quality, additional important orders come in and need to be machined at a higher priority. Due to increasing demands for flexibility production facilities will become dynamic environments, where shop floor layouts and the number, location and type of the engaged machinery change constantly. The robots therefore need to be able to identify these machines either visually or by direct communication. For the LLSF, a team consisting of up to three robots starts in the game area of about $5.6\text{ m} \times 5.6\text{ m}$. A number of semi-finished products is represented by RFID-tagged *pucks*. Each is in a particular state, from raw material through intermediate steps to being a final product. The state cannot be read by the robot but must be tracked and communicated among the robots of a team. On the playing field are *machines*, RFID devices with a signal light indicating their processing status. When placed on a proper machine type, a puck changes its state according to the machine specification which is communicated via broadcast messages. The outcome and machine state is indicated by particular light signals. During the game a number of different semi-finished products need to be produced with ten machines on the playing field. Orders are posted to the robots requesting particular final products to be delivered to specific delivery gates and in specified time slots. All teams use the same robot base, a Festo Robotino which may be equipped with additional sensor devices and computing power, but only within certain limits.

The LLSF has a number of very interesting research questions to be addressed. The robots need to be autonomous, detect the pucks, detect and identify the light signals from the machines, know where they are and where to deliver the final product to. These are the basic robotics problems such as localization, navigation, collision avoidance, computer vision for the pucks and the light signals. Of course, all these modules have to be integrated into an overall robust software architecture. On top of that, the teams need to schedule their production plan. This opens the field to various concepts from full supply chain optimization approaches to simpler job market approaches where a master robot

gives sub-tasks to other robots. If the robots are able to prioritize the production of certain goods they can maximize the points they can achieve. In order to avoid resource conflicts (there can only be one robot at a machine at a time), to update other robots about the current states of machines and pucks, and to delegate (partial) jobs of the production process the robots must communicate. The only means allowed is via a wireless network. Since at RoboCup events there are so many wireless networks, connectivity cannot be guaranteed and is even likely to be interrupted every now and then. Hence robots must have both, useful cooperative behavior, but also sensible single-robot fallbacks. Robots that can recognize each other can have a decisive advantage. In the first phase exploration and roaming concepts must be implemented. A lot of these tasks present cutting edge research problems for the fields of factory automation and CPS, especially in the logistics sector. We intend to emphasize these problems in future tournaments to improve the attractiveness of the league as a benchmark for research and development in these fields.

The LLSF competition started in 2012 for the first time awarding a first RoboCup champion. In order to develop the league further, a number of rule changes have been proposed and implemented such as allowing more computing power on- and off-field, more diverse production plans or the introduction of a Technical Challenge. Another major novelty is the introduction of the referee box which, as in the other leagues, governs the automated game play. A particular benefit in the LLSF is the fact that the game is easy to observe automatically, and hence a large autonomy of the game play can be achieved. To foster the acceptance and implementation of the connection by the teams—while still being backward compatible for this year—certain aspects of the rules have been added and modified to grant bonus points for robots that can communicate with the refbox. For example, during the exploration phase, teams must announce their findings to the refbox, only then they can score.

This paper presents the LLSF as a competitive RoboCup league and details the objectives as well as the ensuing research questions. We outline the leagues' road-map for the next years which we intend to gradually increase in complexity, presenting the automated referee system as a major milestone. The rest of the paper is organized as follows. In the next section, we review cyber-physical systems and their connection to logistics problems arising in automation processes. In Section 3, we discuss the current version of the rules and detail the game objectives. In Section 4, we present the automated referee box (refbox). We first sketch the CLIPS rule base which the refbox is based on, before we describe the tasks and the implementation of the refbox in detail. After proposing possible future advancements of the league in Sect. 5 we conclude in Sect. 6.

2 CPS and Logistics Challenges

Digital devices have infused and changed the private and industrial world. Almost every process is supported, controlled or monitored by a digital device. Taking this a step further leads to the development of so-called cyber physi-

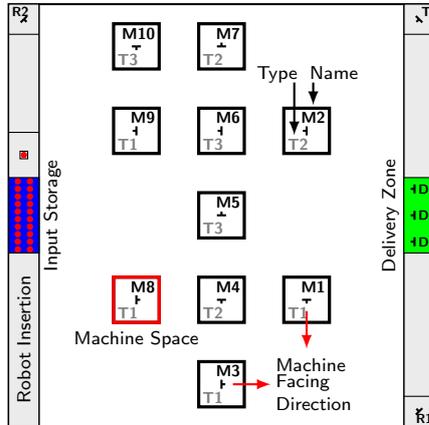
cal systems (CPS) [1]. Here, every entity of a system is equipped with suitable hardware and software for carrying relevant information and for communicating autonomously with its environment to exchange these information [2]. In this way, the entity is transformed from a passive element into an acting agent, which directly interacts with and manipulates its surrounding. This allows for decentralized control as well as increased flexibility and robustness of the affected processes. Examples for possible applications include a.o. the fields of advanced automotive systems, process control, manufacturing, and distributed robotics [3]. Especially logistic scenarios will benefit from CPS by promoting transparency and robustness due to decentralized control. Shipping goods or material can carry address information, handling instructions, or even directly instruct operating facilities. Additionally, due to throughout interconnections and built-in sensors, real-time tracking and tracing becomes possible in so far unrivaled accuracy and quality. However, to allow for large-scaled, reliable, and flexible CPS, research must still find answers to a number of challenges. According to [4] these include finding solutions for

- situation recognition,
- planning and anticipatory, partially or completely autonomous behavior,
- cooperation and negotiation,
- strategies of self-organization and adaptation.

For the application of CPS in manufacturing systems, the requirements for communication within the smart factory are detailed in [5]: CPS must allow for horizontal and vertical communication within the automation pyramid. *Vertical communication* refers to information exchange between the different levels of manufacturing management and control. Nowadays, these are associated to organization-wide enterprise resource planning systems (ERP), manufacturing execution systems (MES) for detailed production planning across machines and the underlying supervisory control and data acquisition (SCADA). While the boundaries between the levels are not fixed, in the long run CPS are expected to completely obliterate them. CPS must therefore be able to exchange information on different levels of abstraction. Thus, non-hierarchical communication structures will evolve which shift the production from a centralized production control to a production controlled by the products which are to be manufactured [6]. *Horizontal communication* refers to data exchange on the same level along the production chain. Thereby, information is handed to downstream production processes which will adapt their production parameters to the individual characteristics of the currently manufactured product. As we will show in the following sections, these challenges are similar to the ones faced in the LLSF.

3 Game Rules and Objectives

As presented in Sect. 2, the deployment of CPS in the logistics of production sites opens a new field of challenges for factory automation. To recommend the



(a) LLSF Competition Area



(b) Machine with light signal, RFID device, and game puck

Fig. 2. LLSF Competition Area and field machine

LLSF as a suitable testbed for research and education regarding CPS in logistics, in Sect. 3 we will present the ideas of the LLSF in more detail.

In 2012 the Logistics League Sponsored by Festo (LLSF) was officially founded. The general intention is to create a simplified and abstracted *factory automation scenario* with an emphasis on logistics applications. Teams of up to three robots operate in a fenced area of about $5.6\text{ m} \times 5.6\text{ m}$ as shown in Fig. 2(a). Fig. 3 shows the original Festo Robotino and a modified version of the Carologistics RoboCup team⁴. The task is to complete a production chain by carrying a (semi-finished) product (a puck in the game) along different machines (signal lights on the playing field as shown in Fig. 2(b)). Points are awarded for intermediate and completed products.

On the field two margin areas on opposite sides contain the puck input storage, delivery zone, and several machines that deal as delivery gates for the final products or as recycling stations. Each puck has a programmable radio frequency identification (RFID) chip with which the different product states S_0, S_1, S_2 , and P_1, P_2, P_3 are distinguished. Initially, all pucks are in state S_0 . In the enclosed inner field, ten signals equipped with an RFID device mounted on its front represent production machines. Each machine is assigned a random but defined type out of the types T_1 – T_5 , which is initially unknown to the robots. The type determines the input and output of a machine. Pucks transition through their states by being processed through machines. The complete production tree is shown in Fig. 4. Circular nodes indicate a puck’s state and rectangular nodes show the respective machine type. For example, the T_1 machine in the upper branch takes an S_0 puck as input with an S_1 puck as output. If a machine, like T_2 , requires multiple inputs, these can be presented to the machine in any order.

⁴ <http://www.carologistics.org>

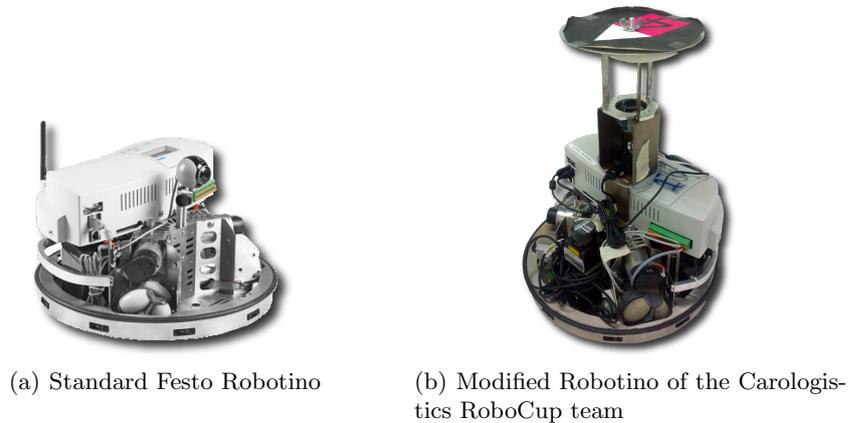


Fig. 3. Standard Festo and modified Carologistics Robotino

However, until the machine cycle completes, all involved pucks must remain in the machine space. The last input puck will be converted to the output puck, all others become junk and must be recycled at a recycling station.

The machines indicate their state after processing a puck using light signals. A green signal means that the particular machine production cycle has been completed, i.e., all required input products have been presented one after another, and now the puck has been transformed to the machine's respective output; for instance, after a T_1 machine transformed a puck from state S_0 to S_1 , an orange light indicates partial success (more pucks are required).

Besides typical robotics tasks such as motion planning or self-localization, the robot needs to plan an efficient sequence of actions to produce as many products as possible in a fixed amount of time. Moreover, the robot has to deal with incomplete knowledge as it is not known in advance what machine has which type. Thus, the robots need to combine *sensing and reasoning* to incrementally update their belief about the world. Based on the knowledge gained, it has to find a strategy to maximize its production output, ideally minimizing costs such as travel distance.

4 Autonomous Referee Box

The LLSF game comprises a rather constrained task for now. Yet overseeing the game requires tracking of more than 20 pucks and their respective states, watching machine areas of 10 machines to detect pucks that are moved out of bounds, placing late order pucks for visual triggering at specified places at certain times, and overseeing completion of the production chain awarding points and keeping a score. This can easily overwhelm a human referee and make the competition hard to understand for a visitor. In fact, in 2012 we needed to review a camera recording of a game to award points in hindsight because a situation was

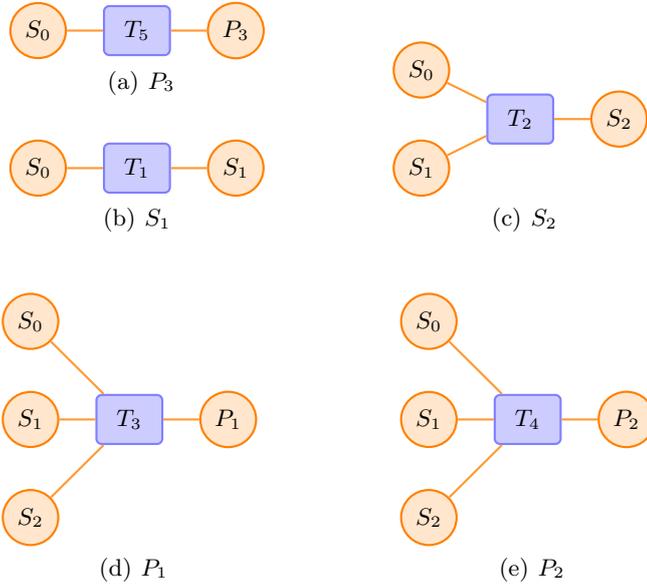


Fig. 4. Production Chain Diagrams showing the machines and inputs relative to their outputs.

overseen by the two human referees. Therefore, we strive for the implementation of a (semi-)autonomous referee box (refbox). The refbox shall control, monitor, and score the overall game. The human referees instruct the refbox (e.g., to start or pause the game), which then communicates with the robots on the field. The humans act as a second tier referee correcting for misjudgment of yet unforeseen situations. Additionally, full autonomy cannot be achieved, yet, because there is no sensor to detect pucks moved out of a machine area.

In the following we will first introduce CLIPS, a logic-based system used to implement the core functionality of the refbox and to formally describe the competition rules. We then give details about the implementation of the refbox.

4.1 The CLIPS Rule-based Production System

CLIPS is a rule-based production system using forward chaining inference based on the Rete algorithm [7]. The CLIPS rule engine [8] has been developed and used since 1985 and is thus mature and stable. It was designed to integrate well with the C programming language⁵, which specifically helps to integrate with the refbox. Its syntax is based on LISP as it was conceived to replace LISP-based systems. The Carologistics team has already used CLIPS to implement the task coordination component [9]. Therefore it appeared suitable for the refbox implementation.

⁵ And C++ using clipsmm, see <http://clipsmm.sf.net>

CLIPS has three building blocks [10]: a fact base or working memory, the knowledge base, and an inference engine. *Facts* are basic forms representing pieces of information which have been placed in the fact base. They are the fundamental unit of data used by rules. Facts can adhere to a specified template. It is established with a certain set of slots, properties with a specified name which can hold information. The *knowledge base* comprises heuristic knowledge in the form of condition-action rules, and procedural knowledge in the form of functions. *Rules* are a core part of the production system. They are composed of an antecedent and consequent. The antecedent is a set of conditions, typically patterns which are a set of restrictions that determine which facts satisfy the condition. If all conditions are satisfied based on the existence, non-existence, or content of facts in the fact base the rule is activated and added to the agenda. The consequent is a series of actions which are executed for the currently selected rule on the agenda, for example to modify the fact base. *Functions* carry procedural knowledge and may have side effects. They can also be implemented in C++. In our framework, we use them to utilize the underlying robot software, for instance to communicate with the reactive behavior layer described below. CLIPS' *inference engine* combines working memory and knowledge base. Fact updates, rule activation, and agenda execution are performed until stability is reached and no more rules are activated. Modifications of the fact base are evaluated if they activate (or deactivate) rules from the knowledge base. Activated rules are put onto the agenda. As there might be multiple active rules at a time, a conflict resolution strategy is required to decide which rule's actions to execute first. In our case, we order rules by their salience, a numeric value where higher value means higher priority. If rules with the same salience are active at a time, they are executed in the order of their activation, and thus in the order of their specification. The execution of the selected rule might itself trigger changes to the working memory, causing a repetition of the cycle.

4.2 Tasks of Referee Box

The refbox has several tasks it must fulfill. It must control the game, communicate with the robots, represent the current state of the game, and interface with the devices on the playing field. We will now briefly explain the four tasks before we describe some aspects of the implementation.

- *Control*. The refbox must oversee the game implementing the rules defined in the rule-book⁶. For this very purpose we use CLIPS. This part is also responsible for awarding points if the robots accomplished a (partial) task.
- *Communication*. It must communicate with the robots on the field to provide information, send orders, and receive reports.
- *Representation*. A textual or graphical application is required to visualize the current state of the game and to receive command input from the human referees. A simplified visualization can be used to explain the game to visitors.

⁶ The current rules can be found at <http://www.robocup-logistics.org/rules>

- *Interfacing.* The referee box needs to communicate with the programmable logic controller (PLC) which is used to set the light signals and read the RFID sensors on the pucks.

4.3 Implementation

The referee box has been implemented by members of the LLSF TC. Its infrastructure is written in C++ and the game controller core in CLIPS. It uses Boost for some of its internals, for example asynchronous I/O and signal propagation.

The base program creates the environment for the CLIPS core, in which the actual game controller is implemented. This core is a knowledge-based system. The facts in the working memory are used to keep track of the state of the game and to communicate within the core. Rules trigger on specific conditions and events, for example the reception of a message, or the completion of a production cycle of a machine. A time fact is periodically asserted (currently at 25 Hz) to allow for time-based triggering, such as in the case of the production completion. This allows us to specify durative actions.

There are currently two interfaces to represent the game state and to accept commands. A textual shell which uses the ncurses library is used for quick operation by the human referee. It shows the most important information and accepts commands. A graphical user interface (GUI) has been implemented using the Gtkmm library. It features a visual display of the playing field and will be focused on visualization and explanation of the game to the audience in 2013.

There are two ways of communication: client-server stream connections using TCP, and peer-to-peer broadcast communication using UDP. This might be changed to multicast at a later point in time. The stream connections are used to connect the refbox tools like the shell or GUI. Broadcast communication is used to communicate with the robots. Both protocols use Google protocol buffers (protobuf) for message specification and serialization. A small framing protocol allows for transmitting messages of different types over the same connection. This is particularly important in RoboCup, where network resources are scarce and combining messages and reducing the number of connection handshakes is beneficial. Using protobuf for message specification gives a very efficient serialization in terms of message size, and allows for forward compatibility, i.e. older clients can still read the messages as new fields are added, they are simply ignored. It also allows for optional fields, further reducing the amount of data required to be sent.

The refbox has been released as Open Source software and can be downloaded at <http://www.robocup-logistics.org/refbox>. The page also contains links to the documentation of the refbox.

5 Gradual Advancement of the Logistic Scenarios

In this section we will propose possible advancements of the LLSF to keep the league challenging for existing teams while preserving a sufficiently low entry

barrier for new teams. The basic idea for developing the league further is to gradually increase the complexity of the game. The objective is to create a benchmark for logistics tasks scenarios addressing the challenges mentioned in Sect. 2. Therefore, we need to push the boundaries towards more realistic scenarios in the following years. We envision challenges in the following areas.

Basic robotic problems. For now, only one team is in the arena at a time. Mostly it is sufficient to navigate with odometry or correct the pose estimate by the unambiguous positions of the observable landmarks. At some point, an opposing team could be in the arena at the same time. It could even try to obstruct machines and hinder the opposing team (within certain limits). With opponents, the whole setup becomes much more complicated and teams need to have solutions for increased situation awareness, collision avoidance, navigation, and self-localization.

Mobile manipulation. A realistic logistics robot will most probably be equipped with a handling device. In the next few years, the pucks will not only be pushed around by the robots, but also have to be lifted up and put into the machine or a conveyor belt. The possibilities to extend the concept of a “machine” as we have it right now are manifold. They would need to be in such a way that a manipulator arm or a fork lift mechanism on the robots needs to be installed. The interaction with machines must not stay limited to one interacting robot. Also cooperative scenarios, where one robot moves the puck, while a second simultaneously activates the machine in a given way or tasks which have to be executed nearly in parallel on different machines are imaginable.

Multi-robot coordination. At this time multiple robots can be advantageous if they split the serial task and complete parts of it with each robot. In the future, we could require multiple robots to cooperate to be able to complete the task at all. For example, two goods may be required to be delivered to two machines in a small time window. Also cooperation of multiple teams would be very interesting, as it would require to agree on a common communication protocol. For instance, one team could produce an item that another has to deliver, and only completing both steps by robots of different teams would score.

Logistics management. The game setup allows for numerous interesting scenarios for logistics management. While the current task still is fixed in many aspects, this should change in the future. Production tasks, machine and product specifications could be changed dynamically, as well as a shift of the optimization goals from a purely market driven maximization of the production output to a cost driven maximization of machine utilization. Also the playfield structure should allow for changes by rotating machines as a first step towards a change of the factory layout, e.g., from a workshop-based layout to a flow-production system. While a part of these changes is communicated by the rebox (see Sect. 4),

other changes have to be recognized by the robots. Competitive teams should have to build more autonomous robotic agents that can cope with such an environment are able to exchange relevant information and react as a team.

Robot hardware restrictions. Currently, the Robotino robot by Festo is the only legitimate robot platform that can be used for the game. Step-wise, hardware restrictions could be dropped to move towards more realistic scenarios. One way for the league could be to proceed towards a number of autonomous fork-lift systems in a real factory scenario.

Referee box development. The current referee box is the first of its kind in the RoboCup domain which has achieved such a high level of autonomy. For the future we want to further increase this autonomy. By adding an overhead camera over the field, for instance, we could automatically track the pucks and robots on the field. This would put the refbox in a position to automatically declare foul play if a puck is moved out of a machine area while it was still needed.

The rules should be implemented in a *rule tic-toc* way. This means that major changes to the rules only become implemented every two years. Major changes will be announced in one year, but only the year after this rule will come into effect. This will give all the teams enough time to tackle the new problem and work towards solutions that are required by the rule change.

Another way to push boundaries further is the introduction of Technical Challenges. In the 2013 competition, the TC announced three technical challenges where teams need to show their technical skills in the field of robotic navigation, localization and computer vision. As in other RoboCup leagues, participation in the Technical Challenge will not be mandatory, but teams will very likely use this possibility to showcase their latest development there. This way, after one particular Technical Challenge has been solved by most of the teams, it can be replaced by a more complicated task. Additionally an open challenge, where teams can show a task of their choosing, can serve as a showcase to propose ideas for future development of the league and its rules. This concept has already been applied successfully to other RoboCup leagues with a strong Technical Committee which is pushing boundaries from year to year.

6 Conclusion

In this paper we gave an overview of the new RoboCup Logistics League Sponsored by Festo (LLSF). The idea is to simulate a factory automation scenario with mobile robots. A team of robots has to “machine” different semi-finished products in order to produce some end-product and then deliver it at a certain delivery point. The team needs to identify machines, derive feasible production plans and realize them in due time. Additionally, they need to cope with failures in the production process that may occur, products need to be recycled and high

priority orders may be placed arbitrarily. All this must be dealt with to optimize the outcome of the factory (and earn as many points as possible).

With this league, exactly the challenges that arise in logistics setting of cyber-physical systems (CPS) are covered. The requirements of the systems are that they enjoy among others planning abilities and autonomous behavior, self-organization strategies or cooperation facilities. Therefore, the LLSF is very well suited to become a benchmark for logistics scenarios of CPS.

We outlined the rules of the league and presented a road-map which will develop the league further towards this eligibility as a benchmark. Requirements towards cooperation and communication, both within the team and with external entities will play an increasing role. A number of rule changes have already been proposed such as allowing more computing power and fostering autonomous behavior with several technical challenges. A major improvement, which we described is the introduction of the referee box, an autonomous referee for the LLSF. Different than in other leagues where referee boxes are mainly used to control the flow of the game, in LLSF the referee box keeps track of the machines types and the produced items and awards the points to the teams as the factory scenario is very complex for a human referee to follow. In the near future, we also want to extend the referee box with an automated robot and product tracking system to take another step towards the automated factory.

References

1. Lee, E.: Cyber Physical Systems: Design Challenges. In: 11th IEEE International Symposium on Object Oriented Real-Time Distributed Computing (ISORC-08). (2008) 363–369
2. Marwedel, P.: Embedded System Design. 1 edn. Springer (December 2005)
3. Lee, E.A.: Computing foundations and practice for cyber-physical systems: A preliminary report. University of California, Berkeley, Tech. Rep. UCB/EECS-2007-72 (2007)
4. Broy, M., Cengarle, M., Geisberger, E.: Cyber-physical systems: Imminent challenges. In Calinescu, R., Garlan, D., eds.: Large-Scale Complex IT Systems. Development, Operation and Management. Volume 7539 of Lecture Notes in Computer Science. Springer Berlin Heidelberg (2012) 1–28
5. Vogel-Heuser, B., , Witsch, M., eds.: Erhöhte Verfügbarkeit und transparente Produktion. Volume 2 of Embedded Systems – I Tagungen und Berichte. kassel university press, Kassel (2011)
6. Hensel, R.: Industrie 4.0 revolutioniert die produktion. VDI Nachrichten (2012)
7. Forgy, C.L.: Rete: A fast algorithm for the many pattern/many object pattern match problem. *Artificial Intelligence* **19**(1) (September 1982)
8. Wygant, R.M.: CLIPS: A powerful development and delivery expert system tool. *Computers & Industrial Engineering* **17**(1–4) (1989)
9. Niemueller, T., Lakemeyer, G., Ferrein, A.: Incremental Task-level Reasoning in a Competitive Factory Automation Scenario. In: Proc. of AAAI Spring Symposium 2013 - Designing Intelligent Robots: Reintegrating AI. (2013)
10. Giarratano, J.C.: CLIPS Reference Manuals. (2007)
<http://clipsrules.sf.net/OnlineDocs.html>.