Robot Soccer
An Example for
Autonomous Mobile Cooperating Robots

Gregor Novak[1]

[1]Vienna, Austria
novak@robotsoccer.at

Abstract — *The aim of this paper is to give a short introduction to an autonomous robot system on the practice-oriented example of robot soccer as a test bed for the implementation of Multi Agent Systems. It is a great opportunity to work with robot soccer, because it is covering almost every parts of an autonomous system and gives an idea how such a system can be realized.*

1 Introduction

The topic ”autonomous mobile cooperating robot systems” is one of the most seminal technologies. This is indicated by the fact that leading nations like the USA and Japan have adopted it to one of their national research topics. Research in this field needs knowledge of different technical sub domains. These are mechanical engineering (e.g. for mechanical construction), control engineering (e.g. control algorithms), electrical engineering (e.g. for hardware design, electric drive, communication) and computer science (e.g. for data analysis, modelling and representation of the application, intelligent selection of the actions, programming). This paper deals with the design of a robot system by means of a robot soccer system.

Autonomous mobile cooperating robots belong basically to the group of Multi Agent Systems (MAS). The term Multi Agent Systems (MAS) is one of the two sub-fields of Distributed Artificial Intelligence (DAI), which is on its part a sub field of Artificial Intelligence (AI). DAI is concerned with systems that consist of multiple independent entities interacting in a domain. The second sub-field is Distributed Problem Solving (DPS).

MAS deal with the behavior management in collection of several independent entities, or agents. An agent must have at least the following features [5]:

- It perceives the world in which it is situated.
- It has the capability of interacting with other agents.
- It is pro-active in the sense that it may take initiative and pursues persistently its own goals.

– It is an entity with goals, actions and domain knowledge situated in an environment.

The keyword Multi Agent System was introduced approximately 30 years ago. It is defined as a collection of, possibly heterogeneous, computational entities, having their own problem-solving capabilities and which are able to interact in order to reach an overall goal.

Working with robot soccer is a great opportunity to deal with a lot of different kinds of technical subject areas. Although it is fascinating to see small robots playing soccer, robot soccer seems on the first view to be a simple stupid game. However it only seems so. If going into detail the whole complexity turns out. It is possible to deal with every technology, which is necessary for an autonomous system. Starting with the sensor system e.g. vision system, encoders, etc., which are necessary to bring the important environmental facts in a form, which the control system understands. The prediction and position estimation is another important part. Coming to the control mechanism, which has to generate tasks by using neural networks, fuzzy, or simple if - then rules. Out of these tasks a trajectory for the robot is generated. The whole control mechanism has to be covered in a program framework, which provides the possibility of distributed development and of stepwise testing. Because of the speed of the game the cycle time of the control mechanism is very short, which leads to real time problems. The moving platform of such a robot has to fulfill its task, which is at least to follow a given trajectory with maximum precision and to reach the desired speed as fast as possible. This requires knowledge in mechanical engineering and electrical engineering. Furthermore it requires the knowledge of programming a micro controller and of course of control engineering.

1.1 System Configuration

Generally spoken in robot soccer regarding the division of labor between the components of a soccer team, namely between the host computer system and the autonomous mobile robots, three system configurations are defined:

– Remote brainless system
  In this configuration, an external workstation is used to process the data obtained from a vision system and to send the resulting commands to the different robots on the field. The robots just contain modules for propulsion, communication and controlling.

– Robot-based system
  This configuration is clearly the goal for all technical research in this field. The robots act completely autonomously, thus processing data from their onboard sensors. Furthermore they communicate with the other cooperating agents and generate own strategies towards their goal.

– Vision-based system
  This system can be described as the step from the remote brainless to the robot-based system, as some of the intelligence is transferred from the main computer to the single agents, but the control of the vision system and the strategic coordination still remain tasks of the host unit.

The system described in this paper was designed in the way that it is possible to transform stepwise from a vision-based system to a robot-based system. For all the wealth on problems in detail and complexity of a robot soccer game there was never lost the sight of the idea to design a universally applicable system. The basic idea was to develop a fully autonomous robot. For this purpose a layered model with corresponding universal interfaces was conceived. The top layer of this model is a platform for decision finding.

1.2 System Components

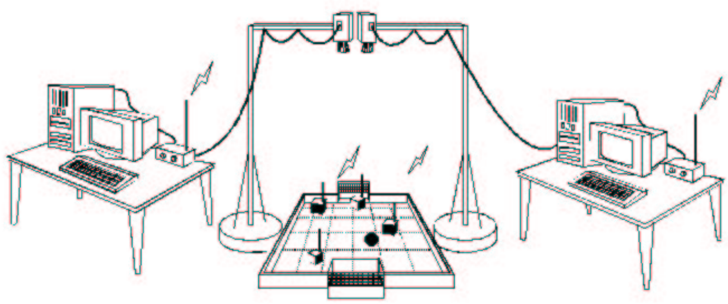

Figure 1: Overall System

In robot soccer different kinds of system configurations exist. The configuration focused in this work is called MiroSOT (Micro Robot Soccer Tournament). In this configuration two teams play soccer against each other on a black playground with a golf ball. The dimensions of the playground are 1.5m of length and 1.3m of width. A team consists of three robots and three additional team members. One or two host computers per team control the robots. The size of the robots is limited to a cube with an edge length of 75mm. The height of the antenna will not be considered when calculating the robot's size. Furthermore each robot has to be marked on its top with at least one color. This is the team color and has to be according to the rules blue or yellow. In addition the robots can be marked with other colors in order to differentiate them within the team on the one hand and on the other hand to detect not only their position, but also their orientation on the playground. The color of the ball, which is a golf ball, must be orange. The positions of the moving objects on the playground are detected by the aid of their color information. 2m above the playground a camera is mounted, which transmits pictures to the host computer. In the host computer a vision system detects the position and orientation of the moving objects. This information is the basis for decision-making. With a radio communication the desired movement of the robots is sent back to the robots. Rules regarding the game, which are defined in the regulations of the FIRA (http://www.fira.net/games/mirosot.html), include the definition of scoring methods, fouls, game interruption and special kicks awarded to a team (Free Kick, Penalty Kick or Free Ball, just to mention a few). As host computer serves a usual in trade Pentium PC. The tasks of this computer are image processing, communication with the robots as well as planning of the next movements and the control of the players. A game has two half times with five minutes net playtime each. The teams have the possibility to take one timeout per halftime.

2 The System

In the described system (Figure 2) was set a great store to have the possibility of an easy and independent further development of the components [7]. The components themselves should abstract the parts of a robot. The component model and the class model were developed in imitation of a soccer playing human. For instance self-observation for the whole movement control has led to the following result. For a movement you are looking for a checkpoint. For example the position where a ball will be within a short time, or a position beside an opponent player whom you have to avoid, or a position where you can block an opponent player, etc. Furthermore you have trained a specific behavior how to shoot a goal, how to give a pass, how to defend, how to stop a ball, etc. All these behaviors are trained and automated but always stamped by a given strategy from the trainer and the behavior of the teammates and opponent players.

This autonomous robot consists of four components:

- The strategy (e.g. strategy robot 1)
- The movement control (e.g. kinematics robot 1)
- Data transmission (communication between the parts)
- The actors and sensors (e.g. robot 1 , vision system)

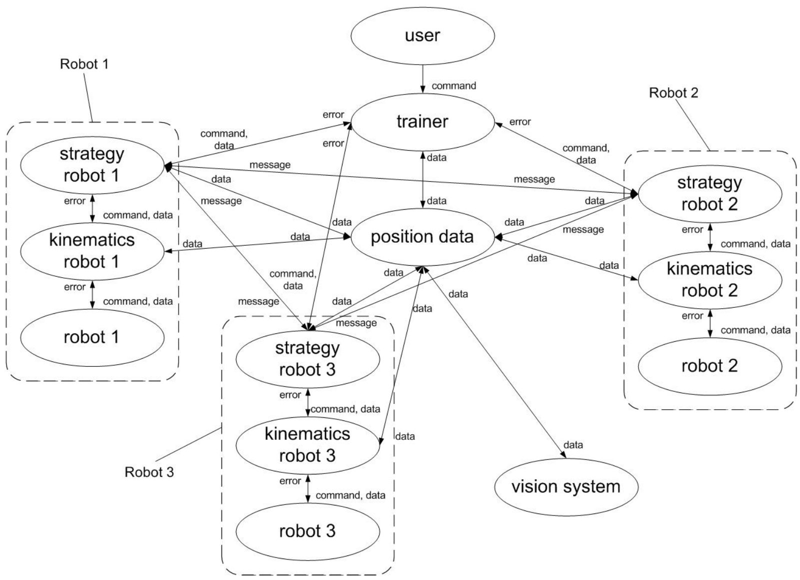

Figure 2: Component Overview

In general a hierarchical controlling structure of a robot is built up as shown in Figure 3 (concerning to [4]). In case of robot soccer the task is created by the strategy (e.g. shoot a goal). This task is dissected in level 3 to positions the robot has to reach. Out of these positions there is built up a trajectory. Partial constant velocities and angular velocities define this discontinuous trajectory stepwise. In level 1 these velocity values are used as demand values for the speed controller on the robot. This speed controller generates out of reference signals from two digital encoders the manipulated variable a PWM signal for the DC - motors.

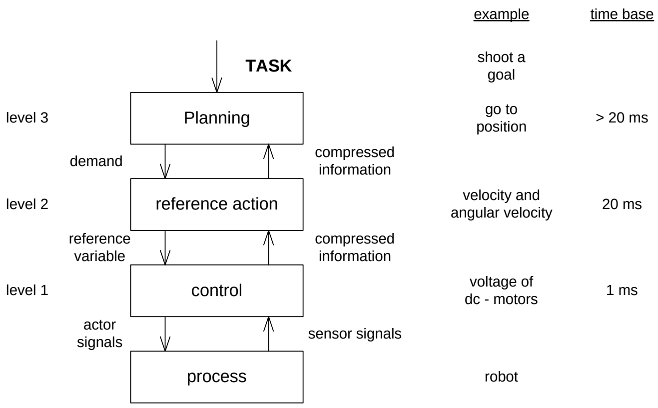

Figure 3: Hierarchical Three-Layer Control Structure

2.1 Object Oriented Concept

The common overview of the object-oriented concept is shown in Figure 4.

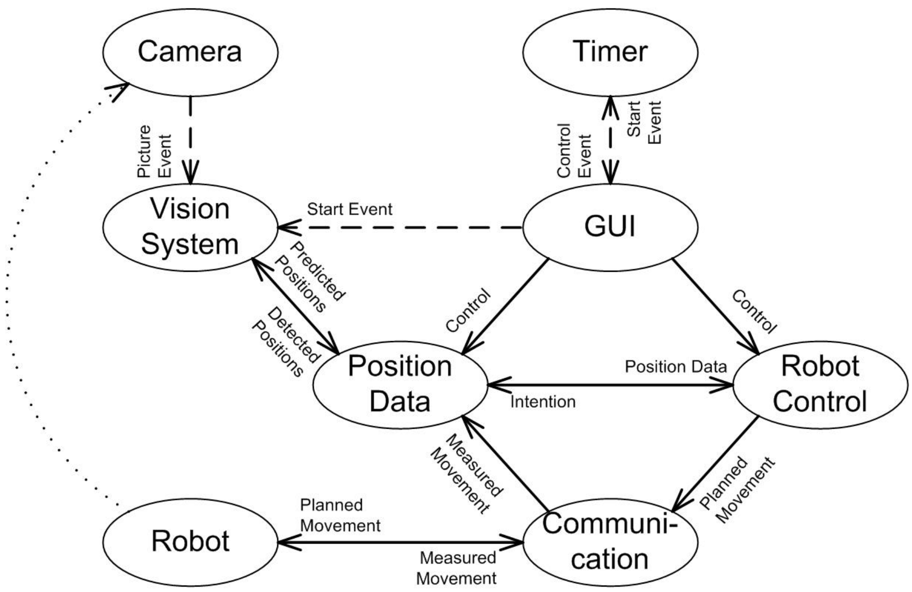

Figure 4: Program Overview

The system consists of:

- The GUI (Graphical User Interface),
- The Vision System,
- The Position Module,
- The Robot Control Module,
- The Communication Module and
- The Mobile Platform (Mini Robot).

For controlling the whole system, the main program is responsible. The vision system detects the positions of the objects moving on the playground. The position module has the task to convert the position data detected by the vision system to an appropriate form for strategy. It has the capability for the estimation of position and the correction of position errors from the vision system. The position data, which were detected by the vision system, are processed in the way that the actual position is valid for the time when the robot can first act on a new command. The strategy module generates the new movements of the robots step by step and delivers them to the communication module. The controller of the robot, which is located in the control module, generates out of its main goal first a sub goal, then a task and out of the task a trajectory. The communication module sends the motion commands of each robot, namely the velocity and angular velocity, to the mobile platform by using radio frequency. For closing the control loop a robot controlled by an onboard micro controller tries to fulfill its given task.

2.2 Vision System

The vision system consists of a camera, a frame grabber, a connection between camera and frame grabber and the detection algorithm. It is currently the main sensor of the whole robot soccer system. Its task is the detection of the positions and the orientations of the objects on the playground. The accuracy and speed of the vision system is one of the most important criteria for a winning robot soccer team. The vision system takes a picture of the whole playground including the moving objects and identifies the objects on the playground. The position of the objects and their orientation is transferred to the control mechanism. The detection of the objects takes place on the bases of color information.

2.3 Sensors

Beside the vision system other sensors can be used. They can be split in external and internal sensors. Internal sensors are mounted on the robot, whereas external sensors are not fixed on the robot. The camera used in robot soccer is an example for an external sensor. It functions as a local positioning system. Internal sensors, for example a co - channel encoder is used for controlling the DC motors of the mini robot. In addition acceleration sensors provide information of the behavior of the whole robot in order to prevent slipping wheels. Gyro sensors and earth magnetic field sensors help to detect the actual movement. Distance sensors could measure the distance between the robot and other players or the distance between the robot and the wall. Color sensors can help to identify the ball and let the robot follow the ball independently. The implementation of small CMOS cameras on the robots let them gather their own view of the world. In combination with a communication on a high technical level the robots can also use the view of the world of the other team members to improve their own view. Every additional sensor makes the whole system more independent of the global camera and makes a robot become more autonomous.

2.4 Position Module

Usually the pictures of a camera are noisy, the detection algorithm never works perfectly and the detected positions are older than the position on which the robots can first act. Therefore the position data have to be corrected. Additionally there are other sensor signals possible which can improve the positions detected by the vision system. For filtering the position data for example the least square algorithms can be used. Good mechanical, kinematical and dynamical models as well as an impulse model are necessary for a proper estimation of the real positions.

The position information detected by the vision system is inaccurate and not actual. That means after a picture was shot, time is required for calculating until the robot can react first. This leads to a dead time. Therefore the transmitted information has to be filtered and the current position estimated. For example the least square algorithm can be used in order to filter the transmitted position information and estimate the current position. For each coordinate value, besides the ball a polynomial of second order is used. For the ball a polynomial of the first order is used. For the robot as well as for the ball is valid:

For the x value:

$$x(t) = a + b \cdot t + c \cdot t^2 \tag{1}$$

The same applies to the y value and the angle.

The least square algorithm is described by the following matrix:

$$\begin{bmatrix} 1 & t_0 & t_0^2 \\ 1 & t_1 & t_1^2 \\ 1 & t_2 & t_2^2 \\ \vdots & \vdots & \vdots \\ 1 & t_n & t_n^2 \end{bmatrix} \cdot \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \tag{2}$$

$$\underline{A} \cdot \underline{r} = \underline{p} \tag{3}$$

$\underline{A}(n \text{ x } 3)$ time matrix

Whereas the time entries are calculated as following

$$\begin{aligned} & t_0 = 0 \\ & t_1 = \Delta t_1 \\ & t_2 = \Delta t_2 + \Delta t_1 \\ & t_3 = \Delta t_3 + \Delta t_2 + \Delta t_1 \\ & \vdots \\ & t_n = \sum_{i=1}^{n} \Delta t_i \end{aligned} \tag{4}$$

$t_0 = 0$ point of time of the last measured position

$$\Delta t_n = T_n - T_{n-1} \tag{5}$$

By this all coefficients Δt_n are negative. Furthermore the coefficients are not essentially equal, because they are depending on the time T_n when the n^{th} detected picture was taken.

$$\Delta t_1 \neq \Delta t_2 \neq \Delta t_3 \cdots \tag{6}$$

In the program the t_n values are stored in an array together with the position data. In each step the lines are shifted down with adding Δt_1 to the time values. The implemented array looks like the following:

$$\begin{bmatrix} 0 & x_0 & y_0 & \varphi_0 \\ t_1 & x_1 & y_1 & \varphi_1 \\ t_2 & x_2 & y_2 & \varphi_2 \\ \vdots & & & \\ t_n & x_n & y_n & \varphi_n \end{bmatrix} \tag{7}$$

The last measured x - value is x_0. The same value one step before is x_1. As mentioned before the time differences Δt_n between the steps are not essentially constant. The time of the last measured position is set to zero and the time of the steps before is relative to the time of the last measured value. Therefore the algorithm is independent from the global time. The same applies to the other coordinate values.

2.5 Control Module

The control module is responsible for generating tasks for the robots moving on the playground. It obtains its position information from the position module. The whole control mechanism is established by two parts. It contains beside the task generation part also the movement part (Figure 5). The higher the level of development the more partitions of this system are implemented on the robot.

The first part, which is generally spoken called strategy, is the decision-making part and is split up in the goal finding sub part and in the task generating sub part. A robot has first to generate a goal. This goal can never be the ball. The primary goal is either to defend or to shoot a goal. This primary goal is divided in sub goals. A sub goal is a position,

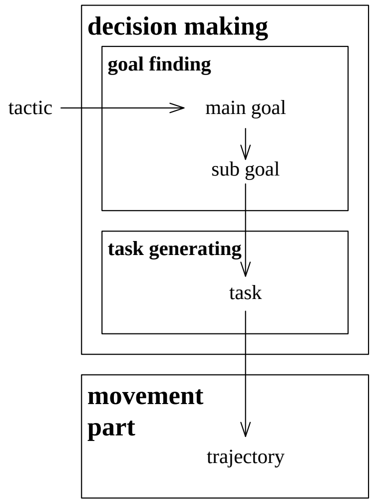

Figure 5: Controller

which the ball has to reach. If - then or fuzzy rules or neural networks are used for goal finding. This is the part of the program where the artificial intelligence is located. These sub goals are split up in the task generating sub part into tasks. A task always leads to movements of the robot. This is done in consideration of path finding methods and the basic movements of a robot. Path finding methods help the robot to maneuver through the opponent and the own robots in order to fulfill the given task. The basic movements of a robot are the abilities that a mini robot has and are located in the movement part. He is trained e.g. to shoot a ball in a certain manner, or to unscrew a ball out of the corner, etc. Therefore the basic movements provide information about the physical possibilities of the mini robot.

The second part of the controlling mechanism is the movement control part. For the implementation of a strategy it is absolutely necessary to have a number of functions available which enable a basic behavior of the robot. A superior strategy can wish merely for future whereabouts and activates of a robot. These have to be fulfilled by a subjacent level. Every complex task can be reduced to one function, which guides a robot towards or through a desired position with or without a desired orientation. This function has to include a controller and a mechanism of avoiding the wall. To fulfill the requirements of the second subgroup a great number of theoretical backgrounds is essential. There is required knowledge in mathematical models of a mini robot, kinematical and dynamical background as well as control engineering.

2.6 Robot

The mini robot for robot soccer is designed as a mobile platform, which can be used for any kind of multi agent system and consists of the following parts:

- Power supply
- Two DC - motors with digital encoder
- Single stage gear

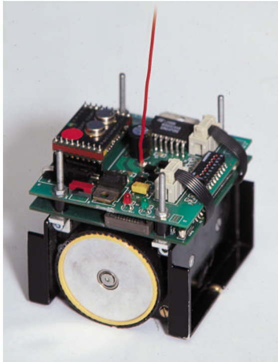

Figure 6: The Mini Robot

- Two wheels
- Micro controller for controlling the rotation of the DC - motors
- Power electronic
- Radio module to send tasks to the robot

2.6.1 Controller of the Robot

In order to win a robot soccer game it is important to have a soccer robot, which reacts very fast on new demand values and stays on its trajectory as accurate as possible. The basic ability of such a soccer robot is to work as a mobile platform. The functionality of such a mobile platform is to control the movement. Therefore it gets two demand values describing a trajectory from its superior control loop. These are are the translatoric velocity and the angular velocity. The robot's behavior is raised to the two DC - motors. Each motor has its encoder, which counts the rotations of the motor. Due to this feedback the motors are controlled. The manipulated variables are the voltages for the motors. The different voltages that are necessary to control the motor are the equivalent to a PWM signal. A PWM (Pulse Width Modulated) signal is a direct current with constant voltages, which is pulsed. The integral over the pulsed direct current signal with constant voltage is equivalent to a direct current signal with a specific current. The controller of this mobile platform, which is designed to react very fast on new demand values and to stay on its trajectory as accurate as possible, is implemented in the robot's micro controller. The sampling time for the controller is primarily based on the encoder resolution and the resolution of the demand value.

2.6.2 Acceleration Sensors

In robot soccer it is essential to have a fast accelerating robot. Arising the acceleration increases the risk of slipping wheels. Usually the prevention of slipping wheels is done via a starting ramp inside the controller. However this method is heavily dependent from the playground conditions. Another possibility to prevent slipping wheels is to use an adaptive controller or to adapt the starting ramp. In order to realize this adaptation additional sensor information are required. The encoder signals are telling only the rotation of the motor. In the case the robot is slipping the encoder signals are unable to measure the real movement of the robot. Acceleration sensors find a remedy. By implementing two acceleration sensors with two axes and putting them over the wheels, or in two opposite corners the whole movement of the robot can be measured. The difference between the encoder information and the information of the acceleration sensors mirrors the slip. The idea is to control the power of the DC - motors in a way that the robot accelerates always at the limit. With an adaptive controller the parameters of the implemented controller are modified. Detailed information on this topic is given by [6].

3 Real Time Aspects

Robot Soccer is a time critical application. As a result of the fast speed of the robot, which is up to 2.5m/s, a short dead time between the picture shot and the reacting of the robot is indispensable. This is indicated by the fact, that at the maximum speed of the robot with a current dead time of about 50ms it leads to an inaccuracy of the last picture of about 100mm. Figure 7 shows the time flow.

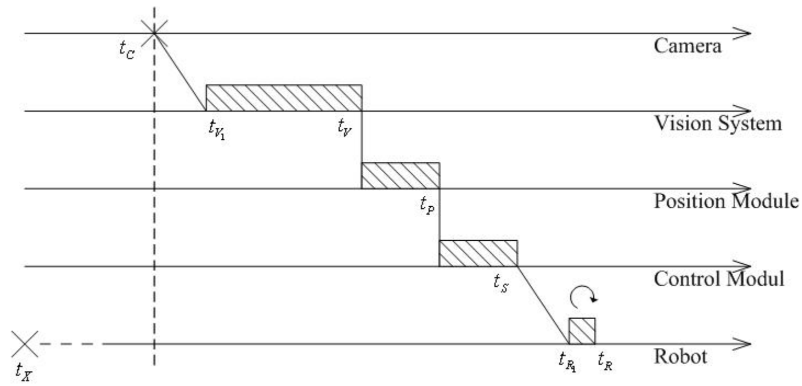

Figure 7: Time Flow

At the time stamp t_C the picture is shot. For transmitting the picture from the camera the time $(t_{V_1} - t_C)$ is required. The calculation of the positions last till t_V. For processing the predicted positions for the time stamp t_R the time $(t_P - t_V)$ is needed in the position module. The strategy needs till t_S. Afterwards time is required for transmitting data to the robots. At the robot the calculation time for the output values of the controller is $(t_R - t_{R_1})$. t_R is the time when the robot can react first on a new command.

In order to shorten this dead time the following strategies can be applied. By using a faster computer the calculation time needed for the vision system, the position estimation and the strategy can be limited. Using a camera providing more pictures makes available more detected positions for the prediction algorithm. Furthermore a faster communication between the host computer and the robot shorts the dead time too. But never the less there is still dead time remaining the prediction algorithm has to compensate.

Another method to reduce the inaccuracy of the calculated actual position of the robot requires an advanced robot. If the robot is able to measure its last movements as well as having the ability to move to a target position, it is possible to calculate for the robot its position at the time stamp t_{R_1} by knowing the position at the time stamp t_C. However, it remains the prediction for the interval $(t_R - t_{R_1})$. In order to decrease the inaccuracy of the vision system the following improvement can be done. The prediction and estimation module located in the position module calculates by the knowledge of a number of previous positions the position for a time stamp t_X, whereas $t_X < t_C$. This position is provided to the robots instead of the position for the time t_C. These methods improve the movement of the robot, only. However it is the most critical part in the system. For the strategy the accuracy of the robot positions is not as important as for the movement part.

4 Future Works

For the future work is planned to omit the host computer, by moving step by step of the system to the robot. This requires an on board vision system as well as further sensors and a powerful processor board for calculating the higher level tasks. In order to realize this, it is planned to develop a robot made up by several independent modules networked by a special bus system.

References

[1] H. Kitano: *RoboCup97, Robot Soccer World Cup 1* Springer-Verlag Wien New York (1998) ISBN 3-540-64473-3

[2] H.R. Everet: *Theory and Applications* Sigma Press (1995) ISBN 1-56881-048-2

[3] J.L. Jones and A.M. Flynn: *Mobile Roboter* Addison-Wesley (1996) ISBN 3-89319-855-5

[4] G. Schmidt: *Regelungs- und Steuertechnik 1* Skriptum TU München (2000)

[5] E. Oliveira, K. Fischer and O. Stepankova: *Multi-agent systems: which research for which application* Robotics and Autonomous Systems Elsevier Science B.V.

[6] B. Putz: *Antriebsregelung eines mobilen Kleinroboters* April 2002 Diploma work Unpublished

[7] G: Novak: *Multi Agent Sytems - Robot Soccer* April 2002 PhD theses Unpublished