KiRo – A Table Soccer Robot Ready for the Market

Thilo Weigel

This paper presents the autonomous table soccer robot KiRo. KiRo provides a competitive challenge for even advanced human players and is well suited as a toy or even as a training partner for professional players. Moreover, the table soccer game represents a demanding testbed for evaluating a multitude of techniques and approaches in the fields of robotics and artificial intelligence. KiRo has reached a technically mature level and will be commercially available by January 2005.

1 Introduction

For research in artificial intelligence and robotics it is common practice to first develop and test robotic systems in restricted domains. Domains like, for example, robot soccer [7] provide a simple yet challenging environment where a robot can be exposed in a controllable way to difficulties it may encounter in the real world.

However, such domains can be more than just an intermediate step towards the long term goal of having robots operating in everyday real world environments. Beside the fact that robots are a growing and popular category of toys, it is a research challenge in itself to overcome the prototype stage of academic robotic systems. For a marketable system, high demands regarding robustness, reliability and safety have to be met.

One appealing domain is the table soccer game[1]. On one hand, it is a popular pastime in bars and amusement arcades, and it is a sport in its own right, even with world championships[2]. On the other hand, it provides an interesting and challenging testbed for a number of research areas, such as sensor interpretation, real time control and autonomous systems.

In this paper we present the autonomous table soccer robot *KiRo*[3]. KiRo allows humans to play the table soccer game on a regular table against a machine. KiRo observes the playing field with a camera and controls the four rods of one team according to his observations and pre-defined tactics. In numerous test games KiRo showed to be a competitive challenge, even for advanced human players. KiRo's hard and software is fully developed and it is planned that KiRo will be commercially available by January 2005.

A prototype of KiRo was first presented in 2002 at conferences (RoboCup-Symposium '02 [10], IROS '02) and commercial fairs ("Hannover-Messe", "IMA"), as well as on TV ("Welt der Wunder") and in news magazines ("Der Stern"). As public feedback was very positive, the German company *Gauselmann AG*[4] licensed KiRo and brought KiRo's hardware to a stage robust enough for being marketed. We, in turn, adapted and further improved the software to make it sufficiently user friendly, robust and efficient.

Other sport games have been used as domains in robotic research as well. People in the *RoboCup* community use simplified variants of the soccer game for their research in fields such as mechatronics, robotics, artificial intelligence and machine learning [3, 6]. In less complex games, though, no simplifications are necessary, and a direct contest between humans and a machine is possible. Anderson demonstrated his research on dynamic sensing and stereo vision with a humanoid robot playing ping-pong [1]. Bentivegna *et al.* taught a humanoid robot how to play air hockey by letting it imitate a human opponent [2]. Also, Bishop *et al.* proposed this game as a testbed for research in intelligent robotic systems and built a robot arm specialized for air hockey playing [4]. Moore *et al.* used billiards for evaluating different variants of memory-based learning [9]. However, none of these systems has reached a level as technically mature as KiRo.

The rest of this paper is structured as follows. Section 2 describes the general table setup and rod control system. In Section 3 we present the vision system and the way in which a world model is obtained from the sensor data. Section 4 describes KiRo's basic actions and how they are selected. In Section 5 we outline the simulation system used for software development and evaluation. Section 6 shows statistics on KiRo's capabilities and reports results from games KiRo played against human opponents. We conclude in Section 7.

2 Rod Control

For the first KiRo prototype we developed specialized rod control units which were attached to the outside of a commercially available table soccer table. A color camera was mounted above the table for perceiving the players and the ball. An external PC connected to the camera and the control units [10].

However, in order to meet the industrial requirements regarding safety, robustness and product design, *Gauselmann AG* developed a completely new table purpose-built for table soccer games between humans and a machine. The table follows a common design with a playing field size of 1200

[1]Table soccer is also commonly known as "Bar Football" or "Foosball".

[2]See http://www.table-soccer.org/.

[3]*KiRo* stands for *Kicker-Roboter*, which is the German expression for "table soccer robot".

[4]See http://www.gauselmann.de.

mm x 680 mm and 11 players distributed among 4 rods for each team.

Figure 1 shows a picture of the table. It is designed so that the rods only extend outside of the table where the humans control their players. The rods for the human players

Figure 1: A picture of the table.

are simply hollow shafts which slide on an axle fixed inside the table.

In contrast, the rods controlled by KiRo consist of an outer and an inner hollow shaft with the playing figures attached to the outer one. Figure 2 shows how a rod is assembled. Both shafts intertwine such that turning the inner

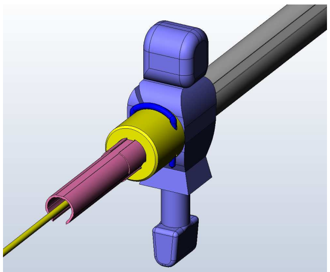

Figure 2: The structure of a rod. Along the longitudinal axis the shafts are cut open such that the end piece of the outer shaft can reach inside the inner shaft, where the circumferential wire is attached to it.

shaft causes the outer shaft to rotate as well. Inside both shafts a wire is attached to the outer shaft. Running around the entire table the wire can pull the outer shaft with the players in both directions along the rod.

Figure 3 shows the interior of the table. A motor attached to the inner shaft of a rod controls its rotation. A

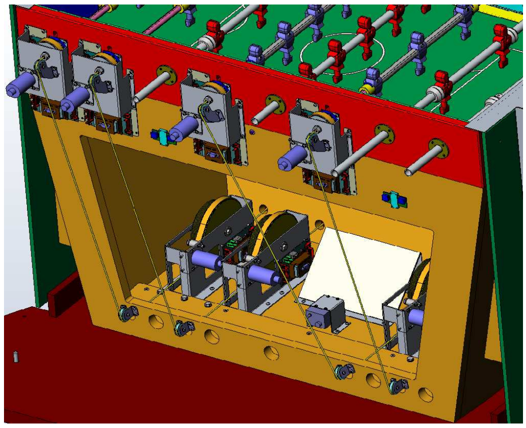

Figure 3: The table interior revealing the control units and the camera rack with a mirror. The wheels on the bottom wind up and move the circumferential wires. The translational motors are connected to the wheels on the bottom and the rotational motors are connected to the inner shafts of the rods.

second motor controls the translation of the playing figures by turning a wheel with the circumferential wire attached to it.

We use a total of eight *Faulhaber* servo motors and motor controllers – two for each rod. The controllers connect to the PC (*2.4 GHz, 512 MB RAM*) via a *Stallion Easy I/O* multi-port serial adapter.

3 Vision and World Modeling

With a color camera observing the playing field from above, the players and the ball can be recognized in a straightforward way by color classification [10]. However, such a system is usually quite vulnerable to changing lighting conditions and can be easily fooled by putting objects between the camera and the playing surface. Furthermore, the recognition of the ball is always potentially inaccurate when it is covered by a rod or playing figure.

In order to avoid these problems, the table body is now designed as a closed box with an infrared-sensitive b/w camera mounted on its bottom. As depicted in the lower right part of Figure 3, the camera observes the playing field via a mirror from underneath the table. The principal light sources are infrared LEDs mounted along the side walls directly above the playing surface. The LEDs are switched on and off by a camera trigger each time an image is taken. Even though the playing field appears green to the human player it is transparent for some of the light in the visible and infrared spectrum. The vision system is now only used for the purpose of perceiving the ball.

Figure 4 shows a typical camera image. The image is heavily distorted, and, due to the infrared LED illumination, the ball appears clearly visible as two “half-moons”.

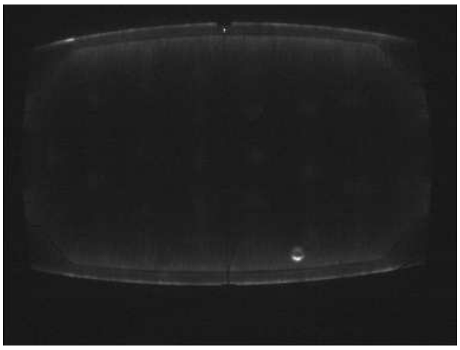

Figure 4: A typical camera image.

As the ball's size and brightness varies considerably over the field, simple thresholding wouldn't be enough for recognizing the ball. However, *image differencing* can be used to cope with the non-uniform lighting conditions.

The basis for this approach is a reference frame which is created initially by averaging over a series of camera images while the ball isn't inside the playing field. In order to detect the ball in subsequent camera images, the difference between the reference frame and the current camera image is calculated first. Then, at all possible ball locations, squares of the known ball size are examined and the number of “significant” pixels with a difference value greater than a certain threshold is counted. If this number exceeds a second threshold, the ball's position is estimated to be the center of gravity of the “significant” pixels. The actual values for the two thresholds depend on the position of the currently examined square. This way, the non-uniform contrast of the difference image and the varying size of the ball's image are taken into account. Efficiently implemented, the described method can detect the ball by examining each pixel only once.

Since the ball can not cover more than a certain maximum number of image pixels, the presence of too much noise can be detected easily. Additional objects on the playing surface usually raise the overall number of “significant” pixels clearly beyond the possible maximum number for the ball. In these situations (which only occur if one tries to fool the system), the ball detection is skipped rather than yielding an unreliable position estimate.

Before transforming from image coordinates to real world coordinates, we further improve the estimate of the ball's pixel coordinates by un-distorting the considerable radial distortion of the camera. For an image pixel given in polar coordinates (r, φ) (with respect to the image center) the radial image distortion can be modeled by a polynomial approximation as $r = r' + \alpha r'^3$, where r denotes the undistorted and r' denotes the distorted radius. We empirically determined α such that un-distorting a complete camera image leads to a visually reasonable “correct” image. Figure 5 shows an example of an undistorted camera image. Plotting the image requires calculating for each pixel (r, φ) the value of its distorted correspondent (r', φ'). This can be done using *Cardano's formula* [5].

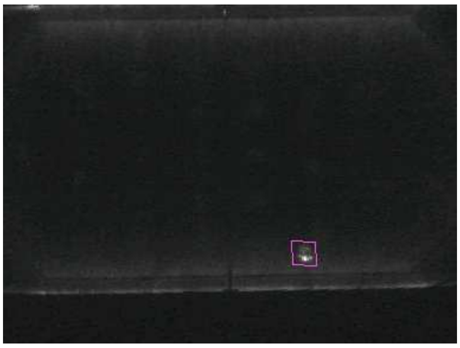

Figure 5: The result of un-distorting the camera image of Figure 4. The square displays the detected undistorted position of the ball.

The matrix for transforming from image coordinates to real world coordinates is obtained by means of two special calibration LEDs. They are mounted at known positions below the playing surface and can be reliably identified in the camera image when they are switched on while the field illumination is switched off.

By processing each half frame separately, we achieve an effective frame rate of 50 Hz with an image resolution of 384x288 pixels. Because odd and even frames result from different scan lines of the video signal, they are treated individually, and different reference frames and transformation matrices are maintained for them.

Since the camera can't perceive the playing figures, the position of the KiRo-controlled rods are obtained from the motor encoders which continuously send their updates every 70 ms. Currently, knowing the position of the opponent's rods is not necessary for KiRo's tactic. Of course, we aim at exploiting this information in the future and consequently plan to incorporate additional position sensors at a later stage.

Figure 6 shows a 2D visualization of the world model constructed from the sensor information. The world model

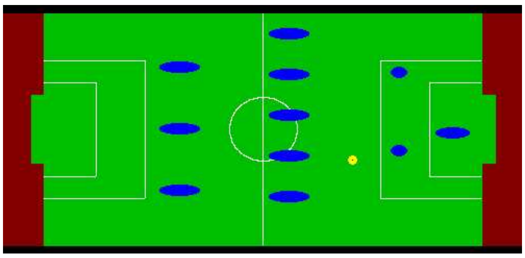

Figure 6: The 2D visualization of the world model. The ellipses indicate the position and angle of the playing figures.

calculates additional information like the ball's velocity and heading, and also detects if the ball is most probably locked between a playing figure and the playing surface. To do so, a rod's rotational target position, its progress in rotation and the current ball position are evaluated.

The world model incorporates new sensor data using a standard *Kalman filter* [8]. While the filter reliably predicts ball reflections at the walls, the error between predicted and measured position usually exceeds a threshold when the ball bounces off a playing figure. In these cases the Kalman filter is reinitialized with the measured position, and the ball's movement vector is estimated based on the heuristic assumption that it just bounced off the rod it was about to reach. Figure 7 illustrates how the ball's moving direction is estimated using its measured position and the position where it was expected to hit the nearby rod.

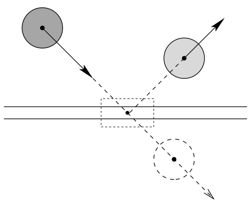

Figure 7: The ball (dark grey) is predicted to pass the rod. Since the predicted position (white) and the measured position (light grey) deviate considerably, the ball is assumed to be at the measured position with the movement vector estimated accordingly.

4 Action Selection and Action Control

Currently, KiRo uses a simple decision-tree for action selection. It considers the current game situation and chooses among the following actions:

- *DefaultAction*: Move and turn the rod to a default position.
- *KickBall*: Rotate the rod by 90° in order to kick the ball forward.
- *MoveKickBall*: Move the rod with the ball locked against the floor sideways and, after a short delay, turn the rod by 360° in order to kick the ball forward[5]
- *BlockBall*: Move the rod so that a figure intercepts the ball.
- *ClearBall*: Move to the same position as *BlockBall* but turn the rod to let the ball pass from behind.
- *BlockAtPos*: Move the rod so that a figure prevents the ball from passing at a specific position.
- *ClearAtPos*: Move the rod to a specific position and turn the rod in order to let the ball pass from behind.

Figure 8 shows the decision tree used to select an action for a rod. Only a few predicates are used and the actions are selected in a straightforward way. All the rods are treated in the same way, except that the positions for *BlockAtPos* and *BlockBall* are calculated individually.

Every figure on a player's rod can move within a certain range. Usually the ranges of two figures overlap, and consequently there can be situations where two figures may be a candidate for moving to a specific target position. In these

[5]Please note, that this shot is used by many human players and is in accordance with official foosball rules because the ball is hit before a full revolution of the rod is completed.

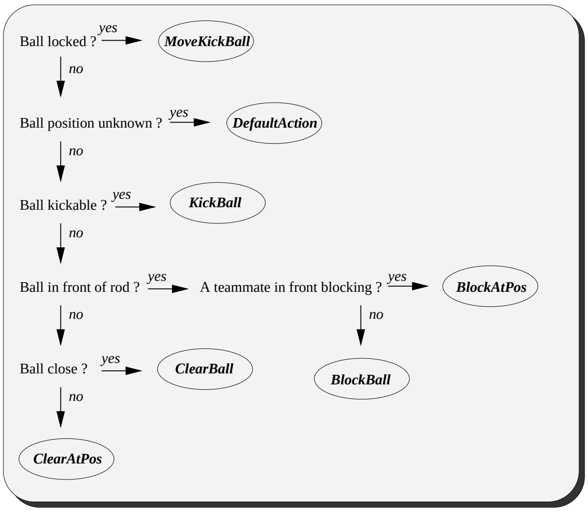

Figure 8: The decision tree for selecting among the basic actions.

cases KiRo selects the playing figure with the best trade-off between the distance from it to the target position and the distance from its range limits to the target position. To avoid oscillation, e.g. when blocking the moving ball, KiRo additionally prefers the playing figure that was already selected in the last cycle.

The exact target position for blocking the ball depends on the position, direction and velocity of the ball. If the ball is moving very fast, the position where the ball is anticipated to pass the rod is taken. However, if the ball is moving very slowly, the target position is set in the way of the expected kicking direction of the opponent. For this, it is assumed that the playing figures of an offensive rod always aim at the goal, while the other rods usually try to kick the ball straight forward. In order to achieve smooth transitions between the anticipated and the expected position, the target positions for intermediate ball velocities are calculated by interpolating between the two positions.

KiRo considers a ball to be *kickable* when it is within a certain range around the playing figure in question. Depending on the width of this range KiRo sometimes hits the ball closer to one corner of the player and consequently makes a diagonal rather than a straight shot. While this somewhat "randomized behavior" is intended for most of the time, the left and right wing attacker purposely try to shoot the ball at an angle towards the opponent goal when the ball approaches from ahead at moderate velocities. In this case the shoot range is selected more precisely.

To avoid unnecessary "hectic" behavior, the velocity at which a rod should move is determined by considering how urgent it is to reach a given target position. If the ball is far away and moving only slowly, there is no need for a fast reaction. But the closer and the faster the ball gets, the higher is the speed assigned to the rods.

A simple way to adjust KiRo's performance in blocking and shooting the ball is to multiply values in the range $[0; 1]$ to the previously determined rotational and translational target velocities. This provides a simple way for defining different playing levels which can be changed anytime during a game.

The currently available basic actions and the manner of selecting them result in a very "agile" and effective behavior. The well organized team line up and the reliable ball blocking skill make it quite difficult for an opponent to bring the ball forward. In turn, the fact that the ball is kicked forward as soon as possible makes it hard for opponents to react fast enough to block it.

5 Simulation

In order to aid the development and evaluation of software components we developed a very realistic 3D-simulator for the table soccer game. The simulation is based upon the freely available *ODE* library[6] which is particularly suited for simulating articulated rigid body dynamics. Figure 9 shows a screenshot of the *Open GL*-based 3D-visualization of the simulator.

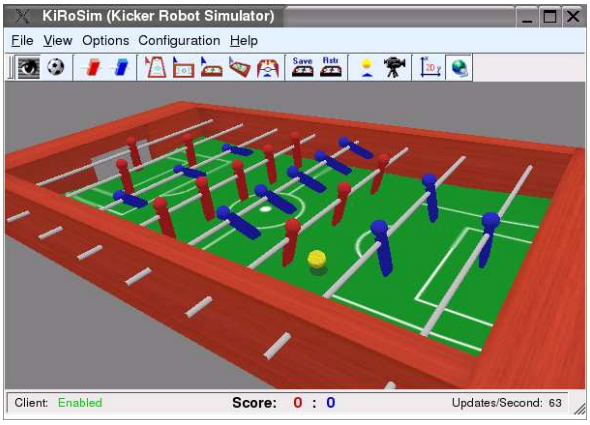

Figure 9: The 3D visualization of the table soccer simulation.

KiRo communicates with the simulation server via TCP/IP. It receives the same type of position data and calculates the same motor commands no matter if it is connected to the simulator or to the real table. In fact, this can be changed with a mouse click at runtime. Two clients can connect to the simulator such that different skill parameters and playing strategies can be evaluated in games KiRo vs. KiRo.

6 Results

In order to demonstrate the velocities at which KiRo reacts, blocks and kicks, we evaluated a series of log files from regular games and examined KiRo's behavior in an experimental standard situation.

For this purpose we shot the ball several times from the own midfield rod towards the left corner of KiRo's goal. The goalkeeper was initially placed at the right corner and started to block the ball as soon as it had passed the middle field line. Figure 10 exemplifies in successive world model screenshots (taken every 20 ms) how the goalkeeper successfully blocked the ball. It took the ball approximately 170 ms to travel 525

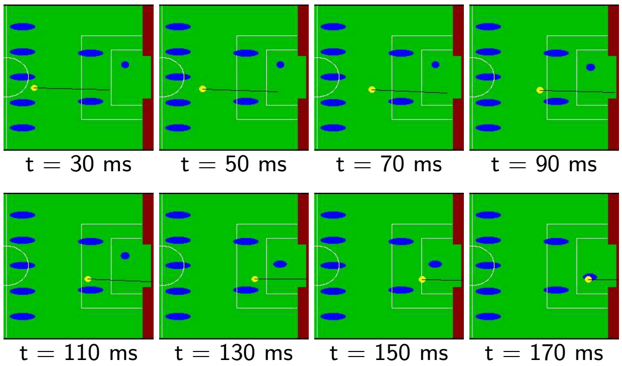

Figure 10: Screenshots of the world model taken every 20 ms, starting 30 ms after the ball passed the middle field line. The black line indicates the ball's direction and velocity.

mm from the middle field line to the goalkeeper's rod. Thus, the ball moved at about 3 m/s on average. This already represents a strong kick – even though experienced human players are able to fire off the ball at 6 m/s and faster.

In order to assess KiRo's reaction time, we examined a couple of shots and averaged the times when KiRo started to move. On average this happened after approximately 80 ms. Yet, considering that the player positions are updated at only 15 Hz, this is only a rough estimate. In Figure 10, for example, the goalkeeper started to move at t = 90 ms.

The goalkeeper moved 125 mm from its starting position to its blocking position. Assuming that it was idle for almost half of the elapsed 170 ms, its average blocking speed can be estimated as 1.4 m/s.

In further tests we moved the defending rod over larger distances and found that a rod can reach an average speed of up to 2 m/s. When kicking, KiRo is able to turn a rod with up to 3000 deg/s. During a game this enables KiRo to shoot the ball with up to 3 m/s.

For evaluating KiRo's general playing performance we conducted a series of test games at the University of Freiburg. Altogether 61 staff members, students and visitors played a total of 61 matches. Each match was played until one team reached 6 goals. Always two players were playing as a team, and each team configuration played only once against KiRo. Every human player played on average 2 games.

For all the games, we assessed the team's playing standard by observing the players' skills and their strategical behavior. We distinguished between the following categories:

- A *beginner* has hardly ever played and has neither special skills nor a strategical understanding for the game.
- An *amateur* plays once in a while, has fundamental skills and a strategical understanding of the game.
- An *advanced player* is playing regularly and consciously improving his skills and strategy while playing.
- A *professional* is explicitly practicing special shots or tricks, plays with an elaborate strategy and competes at tournaments.

[6] See http://opende.sourceforge.net/ode.html.

Of course, only a rough and subjective assessment of the teams' playing standard was possible.

Table 1 shows the results of the games. KiRo won about 85% of all the games and showed an improved performance compared to its first prototype [10]. The very few goals shot by beginners show that KiRo is clearly superior to these players. Against amateur players KiRo was less predominant but nevertheless won 90% of these matches. In contrast to the first prototype [10], KiRo now proves to be a real challenge, even for advanced players. The results in this category indicate that KiRo plays roughly on the level of advanced human players.

	Games won : lost	Goals shot : received
Beginners	12 : 0	72 : 8
Amateurs	29 : 3	177 : 83
Advanced Players	11 : 6	93 : 70
Professionals	0 : 0	0 : 0
Total	**52 : 9**	**342 : 161**

Table 1: Game results.

A longer trial test was carried out in a public amusement arcade. Visitors were able to choose among 6 different speed levels and payed 1 Euro for a game up to ten goals. During 18 days a total of 510 games were played. KiRo won 354 games (69.4% of all games) and scored 4522 goals while it received 2753 goals. On average, 28.3 games with an average duration of 6.06 minutes were played every day.

7 Conclusion

In this paper we presented the autonomous table soccer robot KiRo. Currently, KiRo plays with a simple but effective strategy and is able to win against even advanced human players.

It appears that people enjoy playing against KiRo and are especially ambitious to win against a machine. KiRo's advanced playing level makes it attractive as an entertaining pastime, as well as a more serious training partner for sport players.

KiRo is technically fully developed and ready to be marketed. The hard and software is robust enough to be operated alone by human end-users. Only a simple human-machine interface is necessary to adjust KiRo's playing level and to start the game after a coin is inserted.

KiRo represents a successful example of how the prototype of an academic robotic system has been further developed into a market ready product. To the best of our knowledge, it represents the first instance of a robotic system that plays a well known, unmodified physical sport game on a competitive level.

Nevertheless, there is still room for improving KiRo and taking advantage of it as a robust basis for research into robotics and artificial intelligence. In particular, it seems feasible to incorporate new skills, like stopping or passing the ball, and to devise a more elaborate action selection mechanism that applies these skills in favor of a more deliberate behavior. KiRo could also pre-learn certain skills using the simulator and then refine them on the real table. Additionally, statistics of the opponent's playing style could be obtained and used to adapt KiRo's tactics and level of play – either to exploit the opponent's weaknesses or to keep the game interesting for unexperienced human players.

Acknowledgments

This work has been carried out in cooperation with the German company *Gauselmann AG*, which developed specialized hardware and will market KiRo in the future. We would like to thank all the staff who took part in bringing KiRo to this technically mature level. The authors would also like to thank Dapeng Zhang for his contribution in the development of the table soccer simulator.

References

[1] R.L. Anderson. Dynamic Sensing in a Ping-Pong Playing Robot. *IEEE Transactions on Robotics and Automation*, pages 723–739, 1989.

[2] D.C. Bentivegna and C.G. Atkeson. Learning From Observations Using Primitives. In *Proc. Int. Conf. on Robotics and Automation (ICRA)*, pages 1988–1993, Seoul, Korea, 2001.

[3] A. Birk, S. Coradeschi, and S. Tadokoro, editors. *RoboCup-2001: Robot Soccer World Cup V.* Lecture Notes in Artificial Intelligence. Springer-Verlag, Berlin, Heidelberg, New York, 2002.

[4] B. Bishop, P. Shirkey, and M. Spong. An Experimental Testbed for Intelligent Control. In *American Control Conference*, pages 1328–1332, Seattle, WA, 1995.

[5] G. Cardano. *Ars Magna, Translated by T.R. Witmer as "The Great Art or The Rules of Algebra"*. MIT Press, 1968.

[6] G.A. Kaminka, P.U. Lima, and R. Rojas, editors. *RoboCup-2002: Robot Soccer World Cup VI.* Lecture Notes in Artificial Intelligence. Springer-Verlag, Berlin, Heidelberg, New York, 2003.

[7] H. Kitano, M. Asada, Y. Kuniyoshi, I. Noda, E. Osawa, and H. Matsubara. RoboCup: A Challenge Problem for AI. *AI Magazine*, 18(1):73–85, 1997.

[8] P. S. Maybeck. The Kalman filter: An introduction to concepts. In I.J. Cox and G.T. Wilfong, editors, *Autonomous Robot Vehicles*, pages 194–204. Springer-Verlag, 1990.

[9] Andrew W. Moore, Daniel J. Hill, and Michael P. Johnson. *An Empirical Investigation of Brute Force to choose Features, Smoothers and Function Approximators*, volume III: Selecting Good Models, pages 361–379. MIT Press, 1995.

[10] T. Weigel and B. Nebel. KiRo – An Autonomous Table Soccer Player. In *Proc. Int. RoboCup Symposium '02*, pages 119 – 127. Springer-Verlag, Fukuoka, Japan, 2002.