Robofoot ÉPM Team Description – RoboCup2005 MiddleSize League

Julien Beaudry, Sylvain Marleau, Pierre-Marc Fournier

julien.beaudry@polymtl.ca, sylvain.marleau@polymtl.ca, pierre-marc.fournier@polymtl.ca
Laboratoire de mécatronique, Département de génie électrique, École Polytechnique de Montréal, Montréal, Québec, Canada
http://www.polymtl.ca/robofoot

Abstract. This paper presents some of the technical elements of the Robofoot team of soccer-playing robots developed for the Middle-Size Robot League of the RoboCup 2005. Since there are some solutions that are common to many teams, only the most recent and interesting developments that distinguishes our multi-robot system from others are presented. Specifically, the dual-kicker robot platform is robust and could lead to interesting playing capabilities for a diff-drive platform. The perception localization algorithms include data filtering and sharing mechanisms for different types of objects perceived in the environment. In order to detect various obstacles on the field in real time and without consuming too much processing power, a fast and versatile obstacle detection mechanism is used. This mechanism works in conjunction with a simple but effective method for obstacle avoidance based on a reactive approach. To implement their cooperative soccer-playing algorithms, each robot of the team can use its own Hierarchical Decision Machine, an architecture concept developed recently. Along with these distributed decision machines can exist a Decision Supervisor which has the capability to help coordination and control the play when needed. Much of this work is recent and still in development so some interesting things to come are noted to conclude the team description.

Introduction

The next paragraphs present some of the most recent and interesting developments done by the team of researchers and students involved in the Robofoot project. The group's main objective is to develop a competitive team of soccer-playing robots for the RoboCup Middle-Size League. The project also involves a research program in multi-agent robotics systems (cooperative robotics, mechanics and electronics applied to mobile robotics, real-time computing, perception systems, low and high level control).

To develop a cooperative multi-robot system, a lot of different modules need to be developed and the next paragraphs won’t present all of them since many of these would be redundant with examples of experimental systems found in literature. The text is focusing on some important elements like the main characteristics of the robots platform, their perception system, their obstacle detection and avoidance mechanism, and finally the architecture they use to implement cooperative soccer-playing algorithms.

The team of robots, dual-kicker differential drive platform

The main guideline of Robofoot’s electromechanical platform is simplicity. Since Robofoot is a new team in the RoboCup competition, it has a lot of various elements to develop from very different fields like mechatronics, perception, multi-robot cooperation, etc. In order to obtain a multi-robot system that has all of the required components to play soccer autonomously and to comply with international competition rules, our first generation of robots has been built with modularity and simplicity in mind.

Our robots are using a differential-drive platform specifically designed to meet modularity and performance requirements (Beaudry 2001). It uses two 30W DC motors as driving power and two free wheels for stability. The center of mass of the robot is kept very low so that good accelerations can be reached. The weight of the platform itself is only 4.5kg, but the overall weight of the robots, including all electronics and pneumatic devices is near 20kg. This platform is powered with a 24V 7,2Ah battery pack.

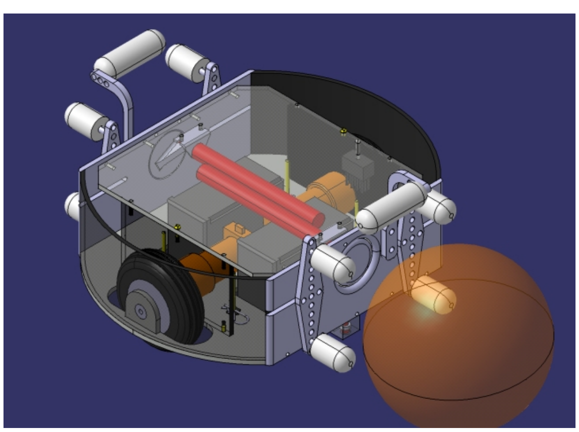
The maximum velocities of the players are 2.5m/s longitudinal speed and 17.5rad/s rotational speed. Longitudinal acceleration can reach up to 3.5m/s^2. The platform offers lateral and longitudinal symmetry, which allows the robot to complete the same operations, with appropriate programming, forward and backward. Our team is constituted of 6 of these robots which are shown in Figure 1. Depending on the configuration of the ball handling device, the number of players on the field will vary from 4 to 6, allowing replacement of at most 2 players in case of failures in the starting lineup.

Fig. 1. : The Robofoot team is a group of 6 mobile robots, 4 to 6 of them will be used in a MiddleSize League match-up.

The device developed for handling and kicking the ball is mainly constituted of a passive handling device and a pneumatic kicker. The more interesting feature of this device is that it is compact enough to fit two devices in the robot, one in the front and one in the back of each robot. With the mechanical symmetry on the longitudinal direction along with the benefits of the omnidirectional vision, the robots are able to dribble and kick the ball forward and backward.

The handling devices are made of 5 damping rolls on each side allowing the robots to dribble the ball forward and backward. Each roll has been precisely located to give sufficient motion resistance for dribbling and turning while not stopping the ball from rolling on the ground. Adjustments on the geometry and roll materials are easily possible to adapt the devices to various field surfaces.

To implement the dual kickers, a pneumatic system has been designed in order to obtain high kickoff speeds and low electrical consumption. A 3000psi cylinder is used to feed a system that works in the 0-150psi pressure range. The pressure of the kicks can be manually adjusted and the valve opening period can be controlled by the robots behavior. This way the kicking power can be precisely adjusted during game play to achieve various goal kicks and passes. The kicking device's autonomy is more than 400 kicks with a filled cylinder.



a) CAD of the device in CATIA

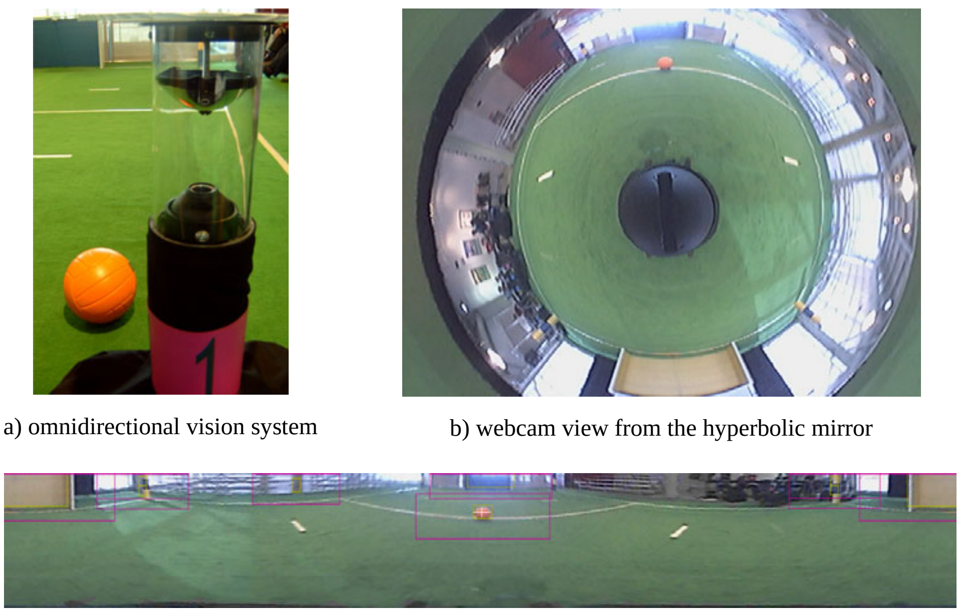
b) Device kicking the ball during tests

Fig. 2. : Dual ball handling/kicking device of the Robofoot soccer playing robots.

Perception and localization using Kalman filtering

Measurements and vision system

The robots perception and localization are based on two measurement systems: odometry (measurement of wheels rotation) and an omnidirectional vision system. The measures returned by the vision system are respectively the horizontal angle between the object and the front of the robot, and the vertical angle between horizon and the object. These angles are function of the object's position and the image focus center's position. Then, the entire vision system is based on a general image representation which allows to use conventional planar camera as well as omnidirectional camera. As the main vision device, the robots use a high-quality webcam with a hyperbolic mirror to achieve low-cost and efficient omnidirectional vision. The camera's image is converted into a panoramic image to simplify further treatments and to facilitate the human interpretation. The conversion is very fast since it uses a predefined look-up table translating pixels from omnidirectional image to pixels in panoramic image. Since the objects of the RoboCup's soccer field use predefined colors, HSI color space is used to achieve precise and robust color or black and white segmentations. Then, object's recognition is based on the estimation of the object's size and position in the image compared to the segmentation results. The next figure shows the physical aspects of the vision system and its image representations.



a) omnidirectional vision system

b) webcam view from the hyperbolic mirror

c) panoramic image used for object detection

Fig. 3. : Omnidirectional vision system, panoramic image transformation.

Kalman filtering

Each type of measurement has advantages and inconvenients. Odometry measurements determine precisely the robot displacements in a given time interval. Their acquisition does not require long treatments and can be done at a relatively high rate. However, integration of odometry measurements cumulates errors which make them impossible to use alone for a long period of time. The vision system measures angles between the front of the robot and landmark. These measures are absolutes but they are least precise and require a relatively long and fastidious treatment. Furthermore, the number of landmarks detected by the vision system changes constantly. Given all these considerations, a Kalman filter (Welch, Bishop 2004) is used to estimate the robot's position and to minimize the error by combining the two types of measurement. This allows precise robot localization and reduces considerably the impact of the image treatment's time delay. Another Kalman filter is used to estimate the ball's position and trajectory. This second filter is based only on vision system's measurements. It considers predictions of ball's acceleration which can be function of the robots positions around the ball.

Data sharing

Our data sharing algorithm which allows robots to share the ball's position and their position is innovative and very efficient (Marleau 2005). Data are always shared with an absolute time reference corresponding to the time of their last update and an estimation of their precision (Bierman 1977). The absolute time reference allows robots to predict how data changed since they were sent and to reevaluate the data's precision. Then, data are combined proportionally to their estimated precisions. For example, this algorithm allows robots to consider more a recent ball's position sent by a robot close of it then an older ball's position sent by a robot far of it. Prediction and correction are done by two other Kalman filters, one on the data sharing server and the other on each robot.

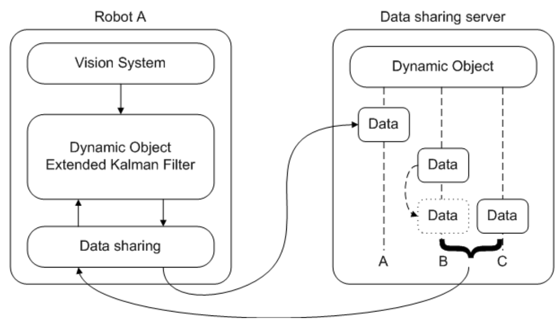


Fig. 4. Data sharing mechanism within the robots distributed perception.

Obstacle detection and avoidance

Fast and versatile obstacle detection mechanism

Apart from the ball and the landmarks, other objects on the field pose collision risks to the robots and therefore must be taken into consideration when choosing movement paths. These are most notably other robots on the field, but may also be in a more general approach, the goals, humans or tools. Such objects are however not practical to track with the main vision system because their shapes and colors are not clearly defined, their movement may be unpredictable, and the trajectory of each one of them need not be tracked. Therefore, we developed a fast algorithm capable, by analysis of the panoramic image, of finding the regions of the field that contain potential obstacles, and returning them in terms of angular and distance ranges to the caller. Unlike the main vision system, the obstacle detection algorithm only identifies regions containing obstacles and it makes no attempt to detect the nature of each obstacle, allowing for significant savings in computing time.

The algorithm is applied several times per second, and the data obtained is used to plot a safe path to the destination of the robot, bypassing obstacles if necessary. The following is an overview of the algorithm used.

First, the region of the field where the robot will pass within a certain time frame is determined. Then, this region is mapped to a rectangle on the panoramic image, which is divided into a number of smaller rectangles. By testing the colors of the rectangles, each one is marked as containing an obstacle or not. If its hue, saturation and intensity components fall within a certain range, a rectangle is marked as being a non-obstacle. Such ranges are predefined for each color considered “safe”, most notably green for the terrain, and white for the terrain markings. Finally, contiguous regions are assembled into a list of angular and distance ranges, and regions smaller than a certain threshold are ignored.

Short computing times are essential for this algorithm to allow frequent updates of potential obstacles for safe navigation in a dynamic environment. In order to diminish the computing time, at the cost of some precision, the size of the rectangles within the region to analyze can be increased, which effectively lessens the resolution of the analysis. In order to assign a single HSI value to each rectangle (which is composed of many pixels), we use the color value of a pixel near the center of the rectangle. While this method may introduce more errors than methods involving average calculations, we have found that it gives satisfying results, while demanding low computing times.

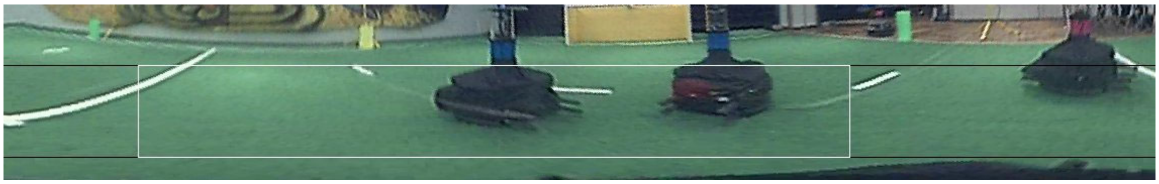


Fig. 5. : Obstacle detection area is mapped to a rectangle on the panoramic image.

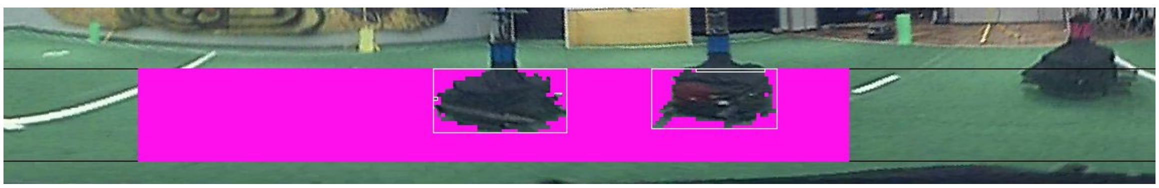


Fig. 6. : Example of results obtained with obstacle detection mechanism. Obstacles are bounded with white rectangles and safe regions are marked with magenta color.

Reactive obstacle avoidance in a dynamic environment

Many different approaches are possible to allow the robots to navigate on the field while avoiding the numerous obstacles on it, the other robots which are also moving on the field. Since our robots mainly use target selection and attainment to implement their soccer-playing algorithms, some solutions, based on path planning with near optimal solutions come naturally in mind. This type of methods, which we can call predictive ones, for example the Generalized Voronoi Graph method (Latombe 1991), are very precise and appropriate for static environments or very slow dynamic environments.

Since the environment on the field of a robotic soccer match-up is highly dynamic, it is very difficult to predict the position of different players over time, which yields to inefficient navigation when using predictive obstacle avoidance methods. Moreover, in team sports like soccer, one of the key advantages of good players are speed and responsiveness. We must then obtain a method that offers good navigation speeds with quick reactions to moving obstacles. This can be achieved by using reactive obstacle avoidance methods, which are generally computationally faster than predictive ones.

We have studied various approaches and tested two of them. One being a potential fields based method (Beaudry, Mailhot 2003). This method had the advantage of being completely defined by a mathematical model but experiments shown that it was very difficult to adjust all the parameters to avoid

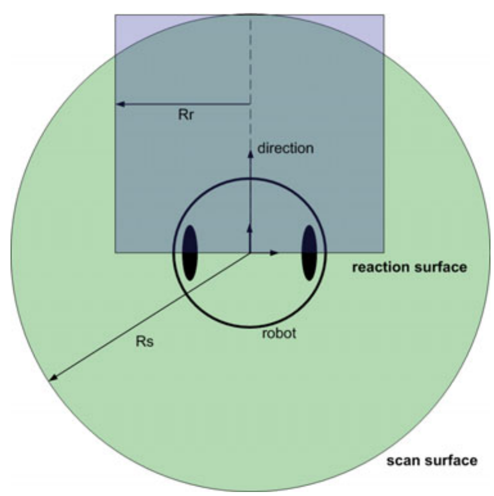


Fig. 7. : Scan and reaction surfaces for the obstacle detection and avoidance method.

collisions in every possible situations. But the most important conclusion about this method is that it is more appropriate for omnidirectional robots. Robots with non-holonomous constraints like the diff-drive platform are most of the time unable to apply movement forces in the resultant vector's direction which greatly compromises avoidance effectiveness.

Another possible approach is by using simple movement commands simply based on the perceived obstacles in a certain range around the robot. This method, along with the visual detection method, is very similar to the Visual Sonar concept developed by Lenser and Veloso (2003). With the benefits of the omnidirectional vision of the robots the circular radar map is obtained only via visual sensing, no need to memorize previously detected obstacles. This algorithm is quite simple to program, computationally fast and has shown very good results in highly dynamic environments. It is schematically explained in the figures 7 and 8. First, the robot determines the path to follow to reach the initial target. Then, it will look at the results of the visual scan to see if there are obstacles potentially disturbing its navigation. Since desired target is frequently reevaluated and obstacles are moving, it is inefficient to scan and react to obstacles too far away. For this purpose, parameters R_s and R_r can dynamically adjust the dimensions of the scan and reaction areas, based on environment dynamics (robot and obstacles speeds, accelerations and sizes). If there are robots in the reaction area, the robot will find the shortest turn allowing obstacle-free navigation and then determine an intermediate target. It will move toward an intermediate target until it evaluates that obstacle-free navigation toward the initial target is possible. Consequently, it should get to the desired target if it is attainable (i.e. not surrounded by obstacles).

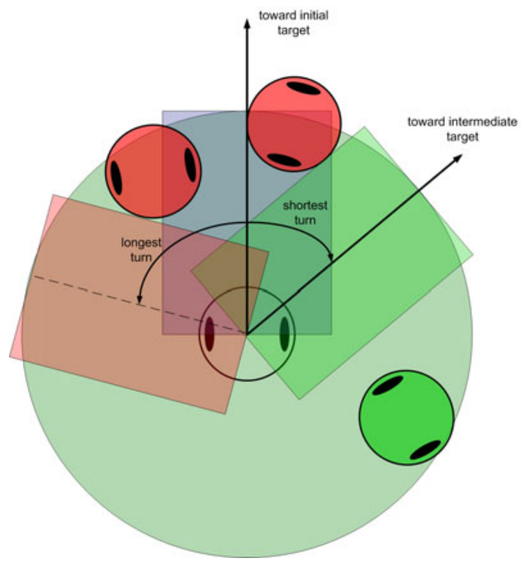


Fig. 8.: Robot target reevaluation with obstacles in its reaction area.

Multi-robot cooperation, hierarchical decision machine with supervised decisions

The control software of our robots has been initially developed for the purpose of one single robot having to complete a predefined sequence of operations (Beaudry 2001). Although this software has been developed with modularity and interesting evolution possibilities, no attention to multi-robot cooperation was then given. The highest level control scheme was based on finite state machines (Reference) and until recently this scheme was also used with the soccer-playing robots.

The need for a more flexible and powerful approach arisen with the need for cooperation and coordination mechanisms between the players. Many different approaches and architectures have been studied before defining the precise guidelines for our development. R.C. Arkin's *Behavior-Based Robotics* (Arkin 1998) and some exhaustive researches on multi-robot architectures (e.g. Singh, Thayer 2001) provide a great understanding of various possible approaches and the main concerns to deal with. Some architectures like CAMPOUT (Pirjanian and al. 2000), Alliance (Parker 1998) and XABSL (Lötzsch 2003) show very interesting features that are well suited for the type of multi-robot system invoked: a multi-robot system with distributed sensing, cognition and control evolving in a structured, highly dynamic environment showing adversity. Given all this background the guidelines for the architecture development can be identified:

- The architecture must be well-suited for a team of cooperative soccer-playing robots, but developed in a more general approach.

- Hierarchical structure: arbitrary deep hierarchy with formal structural elements allowing ease of definition and reconfiguration of decisional structures.
- No need for communication: each robot must be able to coordinate its actions with others without the need for communication.
- Benefits of communication: given some situations yielding to coordination conflicts, communication can address this problem to resolve conflicts with the help of a decision supervisor.
- Versatility in implementation: decision mechanisms are dictated by the application and there must be no constraint to the development of such mechanisms.

The architecture resulting, which is called Hierarchical Decision Machines (HDM), is currently being completed (Beaudry 2005) and Robofoot's soccer-playing robots serve as its main test bed. Simply put, each machine must be constituted of an arbitrary number of levels of decision nodes, leading to behavior nodes which then activate specific actions to implement desired behavior. This architecture can be considered using an hybrid approach where deliberation and reaction can be present in every decision taken, but generally deliberation leads the way to reaction as we descend the hierarchy levels. The numeric and graphical characteristics of the concept are very interesting for decision supervision over the whole decision hierarchy, i.e. from main decision to behaviors. Easy definition and reconfiguration of machines can also result from these characteristics.

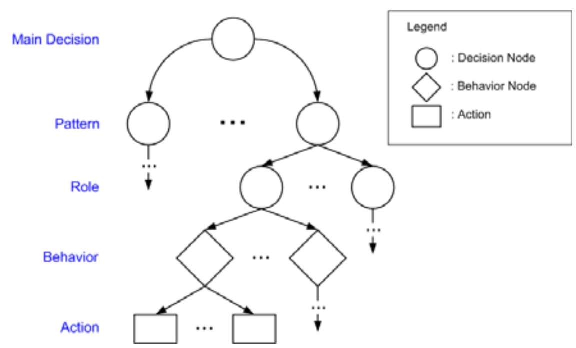


Fig. 9. : HDM structure for soccer-playing robots

The HDM developed for the team of robots is mainly constituted of 4 hierarchical levels which are shown in figure 9: Main Decision, Pattern, Role, Behavior, Action. The Main Decision consist of selecting a pattern based on the current game situation. These patterns are most of the time pre-established formations that are dynamically selected, like the concept of locker-room agreement (Stone 2000). The number of patterns is not limited and reinforcement learning is particularly well suited for this type of decision mechanism. Metrics are easy to established when good offensive/defensive performance are desired. The aside figures show examples of offensive and defensive patterns with 3 players. Each pattern is defined by a number of specific roles equal to the number of players in the team. Dynamic role assignment is done in a distributed fashion to ensure proper function of the team without communication. A certain role is then given an appropriate set of behaviors to allow it to complete its task. For example, if we take the leader role of the offensive triangle formation, role *TO1*, it can choose between 5 different behaviors. The selected behavior will finally set the action parameters appropriately. The soccer-playing robots have access to only 2 types of actions, moving on the field and deploying their kickers (*Move* and *Kick*). This structure is

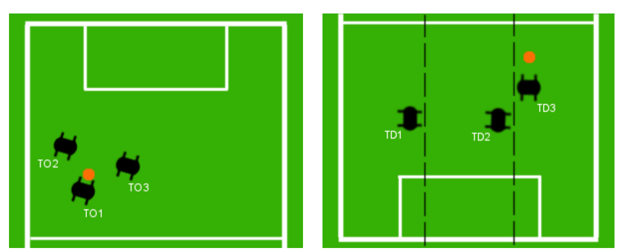


Fig. 10. : Examples of offensive and defensive patterns of the players HDM.

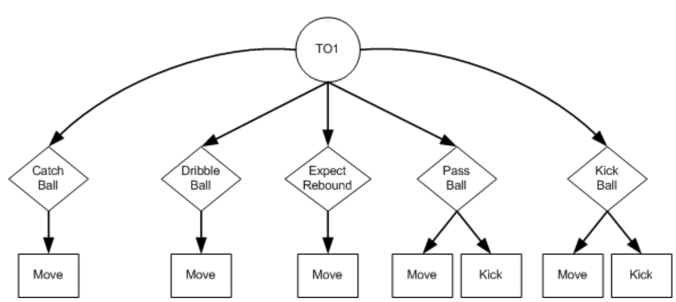


Fig. 11. : Behaviors and Actions for an offensive role.

shown in figure 11. The whole HDM developed for the team is made of an impressive number of nodes, but one interesting characteristic is that many types of roles and behaviors are used multiple times in the machine with different parameters. Reconfiguration of the machine, for example to add more players on the team, is done very easily.

To ensure appropriate coordination, it is necessary that every robot of the team uses the exact same machine. Team coordination is then ensured if every robot chooses the same pattern and if they all select a different role. Decision functions have been implemented to achieve such coordination. Nevertheless, because of the local errors associated with distributed perception, some coordination conflicts can be hardly avoided. In these situations, the decision supervision capabilities that come with the HDMs are very helpful. With the benefits of the data sharing mechanism explained before, a decision supervisor is able to detect such situations that can lead to coordination conflicts. In this case, the supervisor is able to force the appropriate decisions for the conflicting robots. Moreover, with appropriate patterns defined, the supervisor can control the game play by selecting some predefined patterns to respond to various game situations (kickoffs, corner kicks, etc.).

Future work

Many of the elements presented herein are still in active development stages so further results are expected in the months to come. For example, the sensor fusion for the whole team of robots is expected to gain in performance when the filter parameters variations according to confidence levels of each robot will be precisely adjusted. It is also planned to review every single behavior of the Hierarchical Decision Machine developed in order to get the best performance out of our team of robots. Moreover, the Decision Supervisor currently used is at a very early stage of development. In order to improve accuracy of dynamic role assignment and responsiveness of the system in highly dynamic environments, the supervision mechanism will be improved. It should be able to easily detect conflicting situations and show appropriate supervision. It is currently able to do so, but the detection mechanisms are very simple. Much improvement is expected with the use of traditional AI techniques. The concept of decision machine developed also show great opportunities for graphical tools development and scalability. Since its representation is simple and numerical, it is planned to study the possibilities of implementations in various scales of computing architectures.

Much effort is also given to the development of our new robot platform. Since the omnidirectional drive platform is quite superior to the differential drive one in terms of ball handling capabilities, our next generation of robots will be based on an omni-drive platform currently at its first prototype stage (Akiki 2004). It was first planned to have it ready for RoboCup 2005, but our diff-drive platform has been sufficiently improved to give it a chance to show its capabilities. It will also let the developers more time to optimize the performance of our new platform. Competition experiences will also surely lead to interesting income for further developments.

References

Beaudry J.. **Projet SpinoS: conception et contrôle d’un robot à vitesses différentielles**. PFE report, Département de génie électrique, École Polytechnique de Montréal. 2001.

Marleau S.. **Système de perception et localisation pour une équipe de robots mobiles**. M.Sc.A. report, Département de génie électrique, École Polytechnique de Montréal. 2005. (in preparation)

Welch G., Bishop G. **An Introduction to the Kalman Filter**. Department of Computer Science, University of North Carolina at Chapel Hill. 2004.

Bierman, G. J. **Factorization Methods for Discrete Sequential Estimation**. Academic Press. 1977.

Latombe J.-C.. **Robot Motion Planning**, Kluwer Academic Publishers. 1991.

Beaudry J., Mailhot P.-Y.. **Navigation d'un robot mobile dans un environnement dynamique**. ELE6207 report, Département de génie électrique, École Polytechnique de Montréal. 2003.

Lenser S., Veloso M.. **Visual Sonar: Fast Obstacle Avoidance Using Monocular Vision**. In Proceedings of IROS'03, Las Vegas, 2003.

Arkin, R.C.. **Behavior-Based Robotics**. The MIT Press, Cambridge. 1998.

Singh S., Thayer S.. **ARMS (Autonomous Robots for Military Systems): A Survey of Collaborative Robotics Core Technologies and Their Military Applications**. Robotics Institute, Carnegie Mellon University. 2001.

Pirjanian P., Huntsberger T.L., Trebi-Ollennu A., Aghazarian H., Das H., Joshi S., Schenker P.S.. **CAMPOUT: A control architecture for multi-robot planetary outposts.** In Proceedings of the SPIE Symposium on Sensor Fusion and Decentralized Control in Robotic Systems III, Vol. 4196, Boston, MA, Nov. 2000.

Parker L. E.. **ALLIANCE: An Architecture for Fault Tolerant Multi-Robot Cooperation**. IEEE Transactions on Robotics and Automation. 1998.

Lötzsch M., J. Bach J., Burkhard H.-D., Jüngel M.. **Designing Agent Behavior with the Extensible Agent Behavior Specification Language XABSL**. In 7th International Workshop on RoboCup 2003 (Robot World Cup Soccer Games and Conferences), Lecture Notes in Artificial Intelligence, Padova, Italy. 2003.

Beaudry J.. **Machine décisionnelle pour systèmes multi-robots à perception distribuée**. M.Sc.A. report, Département de génie électrique, École Polytechnique de Montréal. 2005. (in preparation)

Stone P.. **Layered Learning in Multiagent Systems**. The MIT Press, Cambridge. 2000.

Chainowicz L., M.F.M. Campos M.F.M., Kumar V.. **Dynamic Role Assignment for Cooperative Robots**. Proceedings of the 2002 IEEE International Conference on Robotics & Automation. 2002.

Akiki, M.. **Développement d'un premier prototype de robot omnidirectionnel**. PFE report, Département de génie électrique, École Polytechnique de Montréal. 2004.