Small Robots League, Team *UQ-RoboRoos*, pages 77–80
http://www.ep.liu.se/ea/cis/1999/006/13/

# UQ RoboRoos: On-going Design of a Robot Soccer Team

## UQ-RoboRoos

*Gordon Wyeth, Brett Browning, Ashley Tews*

GW, BB, AT: The University of Queensland, Australia

**Abstract.** *The UQ RoboRoos have been developed to participate in the RoboCup robot soccer small size league. This paper describes the current level of implementation of the robots, including aspects of hardware design, as well as the software running on the robots and the controlling computer. Key features of the RoboRoos design include the agile and powerful mechanical frame, the robots' navigational techniques and a coordinating planner system based on potential field methods.*

### Overview

The UQ RoboRoos are a team of five field robots and one specialist goal keeper robot custom designed to play robot soccer in the small-size league. The field robots feature a high performance mechanical chassis. A novel goalkeeper design has also been developed and implementation is expected to be complete for the contest. Each robot is equipped with a custom Motorola 68332 based controller board. Preliminary communication boards have been developed, with plans for higher bandwidth boards well in hand. The vision system for the project consists of an overhead camera that has been mounted and interfaced to a high-speed frame grabber in a PC.

On the software side, all low-level routines for the robots have been successfully tested with a solid servo loop and reliable communications. The interfaces to this low-level code are duplicated in a kinematically realistic simulator. The simulator supports full game play of the robot team using identical source code to that on the real robot. A game coordinator, called MAPS, has also been developed to coordinate robots in attack and defence. The other major software achievement has been the development of the robot and ball identification and tracking system that currently tracks all objects at 25 fps.

The RoboRoos are a work in progress. The robots competed at RoboCup ´98 in Paris where they were runner-up. Their performance at RoboCup ´98 was limited by the communications system and the delay between frames in the tracking system which was running at 5 fps at that time. Improvements over the last eleven months provide a sound basis for the development of a world class team.

### Mechanical Design

The mechanical design of the field robots has been optimised for speed, acceleration and cornering ability, while maintaining as broad as possible a profile for contacting the ball. The wheels are arranged in a wheelchair arrangement, with Teflon skids at the front and rear of the robots providing the third point of contact. The goal-keeping robot has sacrificed cornering ability for raw acceleration capabilities. The four-wheel drive design ensures that all normal force is used to achieve traction. It is hoped that this high-speed manoeuvrability will enable the goalkeeper to block most attempts on goal. In addition, this robot has a mechanism for trapping and kicking the ball based on a rotating arm.

### Electronic Design

Each robot is powered by custom rechargeable NiCad packs that have the capacity to provide 20 minutes of match time under the most strenuous playing conditions. The batteries are regulated to provide isolated supplies for key electronic sub-systems. The motor drives are H-bridges driven by signals from the CPU. Low on-resistance MOSFETs act as switches for the H-bridge, with level shifting provided by gate drivers. The MOSFETs are complementary N and P channel devices. The system is rated for continuous operation under motor stall conditions, ensuring reliable motor control in all situations.

The robot is controlled by a Motorola 68332 microcontroller that carries a great number of useful peripherals and features for the robot. The microcontroller has 16 x 16 bit timer channels supporting PWM and quadrature capture for the motor drive and other timing functions for sensor capture. The CPU has access to 256 kb of SRAM and 256 kb of Flash memory on its 16 bit bus, providing adequate resources for programming. The microcontroller has a built-in asynchronous serial port that is used for radio communications and a high-speed serial port that interfaces to the sensors.

The robots receive a broadcast signal from the external PC for each frame processed (presently 25 fps). The PC broadcasts all player positions and heading vectors, as well as the ball position. In addition the PC sends strategic commands to the players indicating the best activity for each robot. The nature of these commands will be discussed in the software section. Suffice to say, the information transmitted is constrained by the 9600 bps bandwidth of the single broadcast channel.

### On-Board Software

Software on the robots runs in two threads a cognitive level thread and a schema level thread. The schema level thread is triggered every millisecond and deals with moment to moment navigation problems, such as achieving smooth acceleration or tracking a fast moving ball. The processes that run at this level might be called *behaviours* or *schemas.* The cognitive level thread selects the schemas that are appropriate at a given time, and enforces a hierarchy for arbitration. In addition, the cognitive level maintains planning resources such as paths to goal locations.

The primary schemas are *smooth curve*, *traverse* and *align*, which provide smooth curving, translational and rotational motion respectively. In addition, the schema level keeps track of the motion of the robot to provide a current estimate of position. This process of path integration provides the robot with an accurate position and heading. The effects of the delay between event and receipt of that event on the robot are minimised by updating beyond the delay using the information from the path integrator.

The cognitive level spends most of its time working out how to get to goal locations that are specified in communications received from the planning system. The system works by attempting to turn the robot towards the region of open space closest in direction to the goal location. The cognitive level also determines the maximum speed that the robot should travel at in order not to crash.

**Vision System**

The vision system receives input from an overhead colour CCD camera with a variable length lens. The camera interfaces to a high-speed machine vision frame grabber with a PCI interface. The grabber resides in a 233 MHz Pentium II PC that runs all vision processing as well as generating the multi-agent strategy for the team.

The base colour of the field is removed from the image by analysis of the colour histogram. To separate the field background from the image, histogram analysis is performed on the previous frame. Pixels with RGB values outside the histogram cluster that represents the field are tagged as candidates for robot matches. Features such as lines and noise are removed by successive thinning and dilation operations. The remaining pixel groups are analysed with respect to shape and size features as matches for robots. Successful matches are searched for the ball marker that indicates the team and the centroid. Robots from our team are then analysed by a moment calculation to derive the heading of the robot. Once the heading is derived, the markings on the robot case are checked to verify that we have indeed found one of our robots, and to identify which robot we have found.

The system also reports locations of the opposition robots, the locations of any robot collisions and the location of the ball. If the system is unsure about any feature that it has attempted to identify, it will report the error. The systems on-board the robots will then work on their own estimates of position.

**Multi Agent Planning System (MAPS)**

The planner, called MAPS, provides the strategic commands to each of the robots based on a current assessment of the situation. There are two prime commands that MAPS sends to the robots: GOTO and KICK. Other MAPS commands are used for set plays such as kick-offs and penalties. The prime commands carry position parameters that indicate where the robots should "go to" or "kick to". The coordinates used in MAPS are a 30 x 16 grid. MAPS is based on a combination of potential fields generated from field position and the position of the robots. The first aim of the system is to find a good location for the ball that might reasonably be achieved with a single kick from its current location. The evaluation begins with the base potential field that contains values that indicate the desirability of the ball at each location for an empty field. A lower number indicates a more desirable location, forming a potential well to which a ball might roll. The robot potential fields are then superimposed on the base field to create a useful potential map. Friendly robots form a potential well, while opposing robots form a potential lump. Further features such as the distance to the ball, or whether the cell is in a clear path to the robot are also used to create the potential map.

If a robot is in a good position to kick the ball, it is issued with a KICK command with the parameters based on the minimum location in the map. If no robot is in a suitable position, MAPS selects the nearest to ball and directs it to go to a suitable position using the GOTO command.

If MAPS deems that the opposition has the ball or is likely to assume control of it, MAPS will send defensive GOTO commands to all the other robots that are not going for the ball. The parameters for the GOTO command are designed to occupy the space between each opposition robot and the goal. This corresponds to maximum values in the potential map. If MAPS deems that we have control of the ball, it will issue GOTO commands to each of our field players not involved with manipulating the ball. As in human soccer, each robot is assigned its position on the field such as left wing or centre. Playing positions are implemented with a field position potential field that encourages the robot to move within its specific area of responsibility. The RoboRoos currently play with a defender, a centre and two wings.

**Summary**

The UQ RoboRoos robots provide a basis for ongoing research into navigation in dynamic environments and emergent multi-agent cooperation. Further investigation into these two areas combined with improvements in communication bandwidth and vision processing can only lead to improved robot performance, nearing the mechanical capabilities of the robots. Designing and building the robots has been a highly motivating task for the many undergraduate students involved. The multi-disciplinary nature of the project creates unique project based experiences for the students. Robot soccer will continue to play a role in undergraduate education as well as furthering robotic research.