The UQ RoboRoos Small-Size League Team Description for RoboCup'98

Gordon Wyeth, Brett Browning and Ashley Tews

Department of Computer Science and Electrical Engineering
University of Queensland, Australia
{wyeth, browning, tews}@csee.uq.edu.au

Abstract. The UQ RoboRoos have been developed to participate in the RoboCup '98 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 of RoboCup '98. The project was initiated at the University of Queensland to provide a highly motivating environment for the development of techniques for robot navigation and multi-agent cooperation. The project also provides a unique educational opportunity for undergraduate engineers to work in a team on a complex mechatronic project.

The team of eight undergraduates, two postgraduates and one academic has been developing the robots for twelve weeks. In that time, the team has made remarkable progress. A high performance mechanical chassis for the field robots has been designed, and five copies constructed. A novel goalkeeper design has also been developed, and partial implementation completed 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. Local sensor boards have been designed, but haven't been integrated with the software at this point in time. The playing field has been constructed to RoboCup specifications, with appropriate lighting. 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 routines to those on the real robot. The virtual physics of the simulator realistically represents the physics of the real field, including such issues as communications bandwidth and delay. Code developed on the simulator can be directly ported from the simulator to the robots. Code for chasing and kicking the ball has been developed in this manner, incorporating collision avoidance and directed passing. A game

M. Asada and H. Kitano (Eds.): RoboCup-98, LNAI 1604, pp. 428-433, 1999.

coordinator 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 5 fps. Plans to significantly improve the frame rate are well in hand.

The RoboRoos are a work in progress. Their performance at RoboCup '98 is limited by the current communications system and the delay between frames in the tracking system. The current implementation provides 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. An isometric view of the chassis with electronics removed is given in Figure 1. 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 soft rubber tyres provide a high coefficient of friction, minimising slip at the tyre contact points. Table 1 outlines the mechanical abilities of the robot.

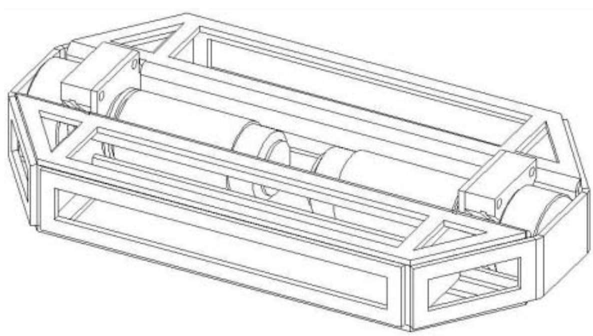

Figure. 1. Mechanical design of the RoboRoos field robots.

	Continuous	Peak
Acceleration (ms^{-2})	1.5	3.6
Speed (ms^{-1})	2	3.4
Time over half field (s)	1.33	0.86
Time for about face (s)	0.6	0.4
Current per motor (A)	0.4	1

Table 1. Mechanical capabilities of the RoboRoos field robots.

The goal-keeping robot has sacrificed cornering ability for raw acceleration capabilities. The four-wheel drive design ensures that all available normal force is used to achieve traction enabling the robot to reach a theoretical acceleration of $8ms^{-1}$. It is hoped that this high-speed manoeuvrability will enable the goalkeeper to block most attempts on goal. In addition, this robot will have a mechanism for trapping and kicking the ball based on a rotating arm. The robot, which is illustrated in Figure 2, is yet to be completed, but is still an effective goalkeeper in its current partial form.

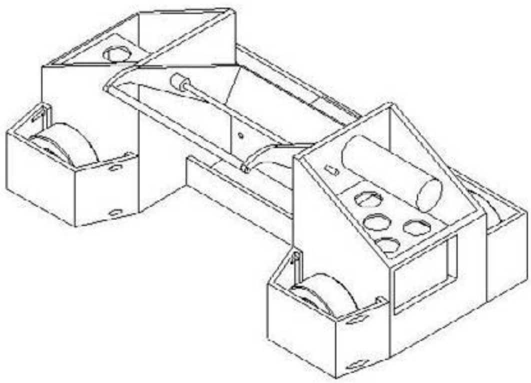

Figure. 2. Chassis design for the goalkeeper robot.

Electronic Design

The electronics are housed on two boards. The main board has the CPU and associated memory, as well as the power electronics for supply of the subsystems and the control of the motors. The RF communication electronics are housed on a separate shielded board. Provision has been made for sensor boards to be mounted to face the front and back of the robot.

Power

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 system is rated for continuous operation under motor stall conditions, ensuring reliable motor control in all situations.

CPU

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.

Communications

The robots receive a broadcast signal from the external PC for each frame processed (presently 5 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 4800 baud bandwidth of the single broadcast channel. The arrival of new RF modules will soon improve the bandwidth, but supplier delays have prevented the modules being available for RoboCup '98.

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.

Schemas

The primary schemas are *traverse* and *align*, which provide translational and rotational motion respectively. Each schema has a length parameter associated with it that allows the schema to terminate safely in the event of loss of communications. For example, if the robot is facing towards the top of the field and wishes to face the goal, the *align* schema will be instantiated with -90° length parameter. The robot will then turn, accelerating and decelerating in a fashion suitable for a 90° turn. The length of the turn is determined by encoder feedback. Should new information come through during the turn, the length parameter may be modified or the schema switched out in a smooth transition. In the event of no new information the turn will terminate smoothly. The schemas also have variable acceleration and velocity parameters. This means, for example, that the traverse operation can be easily modified for an aggressive kick operation by increasing the acceleration and allowed velocity.

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. A delay between event and receipt of that event on the robot is inherent in the use of the overhead vision system. The effects of this delay are minimised by updating beyond the delay using the information from the path integrator.

Cognitive Level

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 decisions made by the cognitive level are tactical commands based on a grid representation of the field. The field is divided into 90mm grid units, which roughly fits the field in a 30 x 16 grid rectangle. The choice of grid size corresponds with one-half of the maximum robot dimensions. As data arrives from the communications system the obstacles are added to the grid to represent the current playing conditions. Having established the obstacles, a flood fill operation is performed to determine the path the robot will take to move to its goal. This operation is carried out by flooding the grid with incrementing values from the current goal.

The robot then uses these numbers, termed *weights*, to decide the best motion from the current position. Lower values of weight indicate a point closer to the goal location.

To evaluate the best current action for the robot, the cognitive level observes the weights currently surrounding the robot and the current direction of the robot. If the direction that the robot is facing is favourable, the *traverse* schema is made dominant. If a directional change is required the *align* schema provides the necessary directional correction. This method provides goal-oriented activity with reliable obstacle avoidance. In addition, the robot is able to get to intermediate locations in the case of a blocked path to the goal, and is able to align with target locations for kick operations.

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 performs all vision processing as well as generating the multi-agent strategy for the team. The primary task of the PC is to accurately locate, identify and track each object on the field. This software must be robust to lighting and positional changes. Software reports a position for each object as well as heading information for our robots at a minimum of 5 Hz.

The image is segmented by colour. As the ball and the robots all have distinct markings this is the obvious way to perform initial segmentation. Colour classification is performed by observing the distance between the RGB vector of each pixel from a specified prototype vector. If the distance is less than a specified threshold then the pixel is classified as belonging to a particular segment. The system is readily adapted by observing RGB values of the desired segmentations for given lighting conditions and adjusting the prototype vectors and thresholds. The ease with which the system can be modified allows the system to be easily adapted to new teams, fields or lighting conditions.

A template match performs the identification of the objects. As the image is segmented, the segmentation routine will call the template matching code when an interesting pixel is found. The template is passed over the region surrounding the seed pixel, searching for the location that maximises the template match. If the match level exceeds a predefined threshold the object is identified and its location reported. Once the objects are identified, the coloured balls on top of the robots are paired to ensure the presence of the robot. This information is also used to determine the heading of the robot.

Planner

The planner provides the strategic commands to each of the robots based on a current assessment of the situation. There are two commands that the planner sends to the robots: GOTO and KICK. Each of these commands carries position parameters that indicate where the robots should “go to” or “kick to”. The coordinates used in the planner are the same 30 x 16 grid used by the cognitive level, but the manner in which this grid is used is somewhat different. This highest level of planning is based on a combination of potential fields generated from field position and the position of the robots (shown in Figure 3) and the algorithm described below.

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, *P(i,j)*. Friendly robots form a potential well, while opposing robots form a potential lump. For the robot with the ball at *(x,y)*, each point *B(i,j)* is then evaluated by Equation 1.

$$B(i,j) \square P(i,j) \square \max(i,j \ldots x,y) \square \mathrm{length}(i,j \ldots x,y) \qquad (1)$$

P(i,j) represents the potential field calculated as above. max() returns the maximum potential field value between *(i,j)* and *(x,y)*. length() gives the Manhattan distance from *(i,j)* and *(x,y)*. If a robot is in a good position to kick the ball, it is issued with a KICK command with the parameters *(i,j)* based on the minimum *B(i,j)*. If no robot is in a suitable position, the planner selects the nearest to ball and directs it to go to a suitable position using the GOTO command.

6	6	6	5	5	4	3	3
9	8	7	5	4	4	2	0
9	8	7	5	4	2	2	0
9	8	7	5	3	1	0	-1
9	8	7	5	3	1	-1	-1
6	6	6	5	4	2	1	2

Figure. 3. Potential field formed by the base field with a robot from each team superimposed.

If the planner deems that the opposition has the ball or is likely to assume control of it, the planner 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 a potential field calculated in similar fashion to Equation 1.

If the planner 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. The coordinates to which each robot goes are determined in similar fashion to Equation 1. 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.