CMDragons 2013 Team Description

Joydeep Biswas, Juan Pablo Mendoza, Danny Zhu, Phillip A. Etling, Steven Klee, Benjamin Choi, Michael Licitra, and Manuela Veloso

Carnegie Mellon University
{joydeepb,jpmendoza,dannyz,mmv}@cs.cmu.edu
{petling,sdklee,bengyenc,mlicitra}@andrew.cmu.edu

Abstract. In this paper we present an overview of CMDragons 2013, Carnegie Mellon's entry for the RoboCup Small Size League. Our team builds upon the research and success of RoboCup entries in previous years.

1 Introduction

Our RoboCup Small Size League entry, CMDragons 2013, builds upon the ongoing research used to create the previous CMDragons teams (1997-2003,2006-2010) and CMRoboDragons joint team (2004, 2005). Our team entry consists of six omni-directional robots controlled by an offboard computer. Sensing is provided by two overhead mounted cameras linked to the offboard computer. The software then sends driving commands to the individual robots. This paper describes the robot hardware and the offboard control software required to implement a robot soccer team.

2 System Overview

Our team consists of eleven homogeneous robot agents, with six being used in a game at any point in time. In Figure 1, an example robot is shown with and without a protective plastic cover. The hardware is mostly the same as used in RoboCup 2006-2010. We believe that our hardware is still highly competitive and allows our team to perform close to optimal within the tolerances of the rules. We focus most of our efforts on improving the software to fully utilize the robots' capabilities instead.

2.1 Robot Hardware

Each robot is omni-directional, with four custom-built wheels driven by 30 watt brushless motors, each featuring a reflective quadrature encoder. The kicker is a large diameter custom wound solenoid attached directly to a kicking plate. It is capable of propelling the ball at speeds up to $15m/s$, and is fully variable so that controlled passes can also be carried out. The CMDragons robot also has a chip-kicking device, implemented by a custom-made flat solenoid located

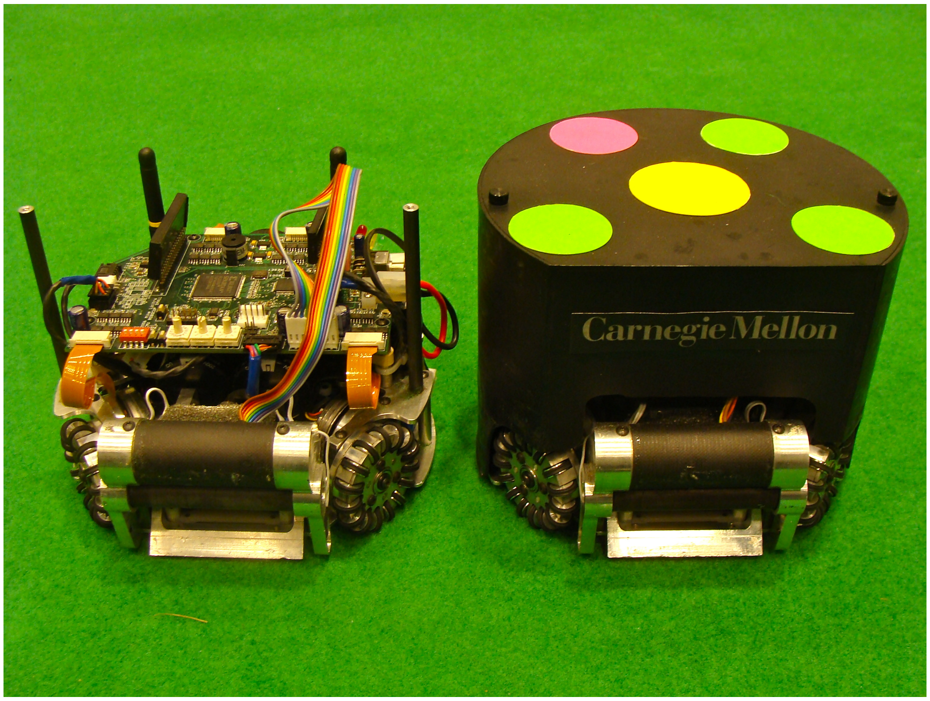

Fig. 1. A CMDragons robot shown with and without protective cover.

under the main kicker, which strikes an angled wedge visible at the front bottom of the robot. It is capable of propelling the ball up to $4.5m$ before it hits the ground. Both kickers are driven by a bank of three capacitors charged to $200V$. Ball catching and handling is performed by a motorized rubber-coated dribbling bar which is mounted on an hinged damper for improved pass reception. A more detailed description of the robot's design and electronics can be found in [1].

Our robot is designed for full rules compliance at all times. The robot fits within the maximum dimensions specified in the official rules, with a maximum diameter of 178mm and a height of 143mm. The dribbler holds up to 19% of the ball when receiving a pass, and somewhat less when the ball is at rest or during normal dribbling. The chip kicking device has a very short travel distance, and at no point in its travel can it overlap more than 20% of the ball due to the location of the dribbling bar. While technically able to perform kicks of up to $15m/s$, the main kicker has been hard-coded to never exceed kick-speeds of $8m/s$ for full rule compliance.

2.2 Software

The software architecture for our offboard control system is shown in Figure 2. It follows the same overall structure as has been used in previous years, outlined

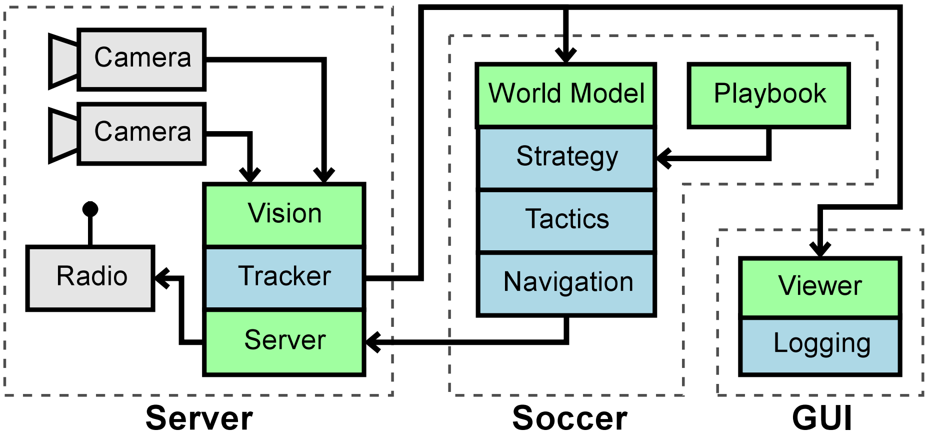

Fig. 2. The general architecture of the CMDragons offboard control software.

in [2, 1]. The major organizational components of the system are a server program which performs vision and manages communication with the robots, and two client programs which connect to the server via UDP sockets. The first client is a soccer program, which implements the soccer playing strategy and robot navigation and control, and the second client is a graphical interface program for monitoring and controlling the system.

The server program consists of vision input, tracker, radio, and a multi-client server. The vision input is supplied via ethernet from the RoboCup SSL shared vision system SSL-Vision [3]. Some of the integration details are described in section 3 of this paper. Tracking is achieved using a probabilistic method based on Extended Kalman-Bucy filters to obtain filtered estimates of ball and robot positions. Additionally, the filters provide velocity estimates for all tracked objects. Further details on tracking are provided in [4]. Final commands are communicated by the server program using a RS232 radio link.

The soccer program is based on the STP framework [4]. A world model interprets the incoming tracking state to extract useful high level features (such as ball possession information), and act as a running database of the last several seconds of overall state history. This allows the remainder of the soccer system to access current state, and query recent past state as well as predictions of future state through the Kalman filter. The highest level of our soccer behavior system is a strategy layer that selects among a set of plays [5, 6]. Below this we use a tree of tactics to implement the various roles (attacker, goalie, defender), which in turn build on sub-tactics known as skills [4]. One primitive skill used by almost all behaviors is the navigation module, which uses the RRT-based ERRT randomized path planner [7–9] combined with a dynamics-aware safety method to ensure safe navigation when desired [10]. It is an extension of the Dynamic Window method [11, 12]. The robot motion control uses trapezoidal velocity profiles (bang-bang acceleration) as described in [13, 4].

Additionally, our system features a detailed physics-based simulator based on rigid-body dynamics as described in [2]. Furthermore, this year we have enhanced the simulator to better serve as a testing platform for our algorithms: First, we have made an effort to match the physical parameters of our simulator to the real world. More importantly, we have implemented an automated referee agent that enforces the RoboCup Small Size Rules during simulation. Implementation of these two points has allowed us to more thoroughly test our algorithms: thanks to the automation and fidelity of the simulator, hundreds of unsupervised iterations of our algorithms can be run, after which we can look at the statistics and interesting cases to find areas for improvement.

3 Vision Hardware and Software

CMDragons 2013 operates using SSL-Vision as its vision system [3]. In our lab, we use two Firewire 800 cameras (AVT Stingray F-46C) which provide a 780×580 progressive video stream at 60Hz. SSL-Vision is released as open source and is therefore available to all teams. In order to use SSL-Vision, the “Vision” component in Figure 2 represents a network client that receives packets from the SSL-Vision system. These packets will contain the locations and orientations of all the robots, as well as the location of the ball. However, data fusion of the two cameras and motion tracking will continue to be performed within our system, as SSL-Vision does not currently support such functionality.

SSL-Vision has the capability to report multiple balls on the field. We use this feature to selectively track balls on the field, which has proven to be especially useful while sharing the field with other teams during testing and setup time during RoboCup.

4 Adaptation to More Robots

Since our last participation in 2010, the rules of the Small Size Robot League have changed to incorporate a sixth robot per team. This required us to restructure our play system, which was based on a five-robot-per-team model. Making this change forced us to think about the scalability of the traditional approach: while writing plays that assign static roles to six robots is doable, this approach does not scale well to teams of, e.g., eleven robots. With that many players, roles need to switch frequently and fluidly, which is why we have modified our play system to assign roles dynamically: During regular gameplay, instead of specifying a role for each robot, the robots automously choose their role based on a utility function that determines where they are most needed at a particular time. This restructuring has lifted some burden out of defining plays for our robots, and it has also made the robots more robust to situations unforseen to the programmers, relying on a general utility function as opposed to a static role assignment.

5 Experience from Logs

A great portion of our efforts this year has been devoted to carefully analyzing logged games from previous tournaments. Our software has the capability of replaying logs, including the ability to visualize the physical state of the world as well as the state of our AI at any moment during the game. This has allowed us to focus our efforts on correcting mistakes and acting more intelligently in situations that actually made a difference multiple times in past games. Furthermore, our log player is designed such that we can implement and visualize passive components of our intelligence on recorded games. This means that, while we cannot change the course of the game, we can run passive algorithms, such as pass evaluation functions, and refine them to act as we would have wanted in the recorded games.

6 Conclusion

Competition	Result
US Open 2003	1st
RoboCup 2003	4th
RoboCup 2004	4th [1]
RoboCup 2005	4th [1]
US Open 2006	1st
RoboCup 2006	1st
China Open 2006	1st
RoboCup 2007	1st
US Open 2008	1st
RoboCup 2008	2nd
RoboCup 2009	Eliminated during quarter-final
RoboCup 2010	2nd

Table 1. Results of RoboCup small-size competitions for CMDragons from 2003-2010

This paper gave a brief overview of CMDragons 2013, covering both the robot hardware and the software architecture of the offboard control system. The hardware has built on the collective experience of our team and continues to advance in ability. The software uses our proven system architecture with continued improvements to the individual modules. The CMDragons software system has been used in four national and eight international RoboCup competitions, and the competition results since 2003 are listed in Table 1. We believe that the RoboCup Small Size League is and will continue to be an excellent domain to drive research on high-performance real-time autonomous robotics.

[1] Provided software component as part of a joint team with Aichi Prefectural University, called CMRoboDragons

References

1. Bruce, J., Zickler, S., Licitra, M., Veloso, M.: CMDragons 2007 Team Description. Technical report, Tech Report CMU-CS-07-173, Carnegie Mellon University, School of Computer Science (2007)
2. Zickler, S., Vail, D., Levi, G., Wasserman, P., Bruce, J., Licitra, M., Veloso, M.: CMDragons 2008 Team Description. In: Proceedings of RoboCup 2008
3. Zickler, S., Laue, T., Birbach, O., Wongphati, M., Veloso, M.: SSL-vision: The shared vision system for the RoboCup Small Size League. RoboCup 2009: Robot Soccer World Cup XIII (2010) 425–436
4. Browning, B., Bruce, J.R., Bowling, M., Veloso, M.: STP: Skills tactics and plans for multi-robot control in adversarial environments. In: Journal of System and Control Engineering. (2005)
5. Bowling, M., Browning, B., Veloso, M.: Plays as effective multiagent plans enabling opponent-adaptive play selection. In: Proceedings of International Conference on Automated Planning and Scheduling (ICAPS'04). (2004)
6. Bruce, J.R., Bowling, M., Browning, B., Veloso, M.: Multi-robot team response to a multi-robot opponent team. In: Proceedings of the IEEE International Conference on Robotics and Automation, Taiwan (May 2003)
7. Bruce, J.R., Veloso, M.: Real-time randomized path planning for robot navigation. In: Proceedings of the IEEE Conference on Intelligent Robots and Systems. (2002)
8. LaValle, S.M., James J. Kuffner, J.: Randomized kinodynamic planning. In: International Journal of Robotics Research, Vol. 20, No. 5. (May 2001) 378–400
9. James J. Kuffner, J., LaValle, S.M.: RRT-Connect: An efficient approach to single-query path planning. In: Proceedings of the IEEE International Conference on Robotics and Automation. (2000)
10. Bruce, J.R., Veloso, M.: Safe multi-robot navigation within dynamics constraints. Proceedings of the IEEE **94** (July 2006) 1398–1411
11. Fox, D., Burgard, W., Thrun, S.: The dynamic window approach to collision avoidance. IEEE Robotics and Automation Magazine **4** (March 1997)
12. Brock, O., Khatib, O.: High-speed navigation using the global dynamic window approach. In: Proceedings of the IEEE International Conference on Robotics and Automation. (1999)
13. Bruce, J.R.: Real-Time Motion Planning and Safe Navigation in Dynamic Multi-Robot Environments. PhD thesis, Carnegie Mellon University (Dec 2006)