CMPack'03: Robust Modeling and Coherent Teamwork

Manuela Veloso, Douglas Vail, Scott Lenser, Sonia Chernova,
James Bruce, Maayan Roth, Juan Fasola

School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, USA

1 Introduction

The CMPack'03 team follows our past teams CMPack'02, CMPack'01, CMPack'00, CMTrio'99, and CMTrio'98 [8, 2, 11, 10]. Our research this year focuses on refining the systems used in last year's RoboCup competition and adding new components that are made possible by the increases in the processing power of the robots. We refined our world model by adding more robust modeling of the ball and opponents. This information feeds into decisions in the behavior system to allow for more intelligent action selection. We also made improvements to our shared world model, which is built up through communication between the robots [5]. These improvements took the form of better uncertainty estimates for our incoming sensor data. In addition to more accurate models, the team's coordination architecture was revisited [9]. Improvements were made by the addition of a token passing mechanism to enforce mutual exclusion in role assignment. Clearer boundaries between roles allowed us to create a more coherent team. To aid in the creation of effective behaviors, we formalized the nested state machine architecture that was used the previous year. Through this formalization we added the necessary infrastructure to trace execution and detect loops in our state machines. The increase in the CPU speed of the robots allowed us to move back to Sensor Resetting Localization (SRL), a particle filter based localization approach [3]. We chose to use this approach because it may be extended to use a richer set of landmarks, such as goal lines and walls, than our previous, Gaussian based approach, which only used field markers. Finally, we used Visual Sonar, a model that returns range and object identity information in a fashion analogous to sonar, for improved obstacle avoidance and localization [4]. As with SRL, this technique was enabled by the faster processors in the robots.

2 Vision

The basic foundation of our vision system is the same as our entry in last year's RoboCup. We use our low level vision library CMVision [1] to do all of the low level processing of color segmentation and region identification. We aim to further increase the robustness of our color segmentation techniques by using local information in the image; in the past, we simply used the pixel value itself to perform the color segmentation. High level vision is responsible for identifying objects of interest from the colored regions returned by the low level vision. This information is then passed to the world model.

In addition to high level information about individual objects, we introduce the use of Visual Sonar to provide general obstacle information to behaviors[4]. Visual Sonar builds a radial map of the distances to objects in the vicinity of the robot. The name Visual Sonar comes from the similarity of the resulting model to the information provided by sonar. We build the radial map by tracing scan lines through the camera image to look for colors associated with particular objects. Scan lines from the base of the robot's neck are created at regular intervals (we used $5°$). The objects found along these scan lines will form a radial map of the objects around the robot with more resolution devoted to nearby objects by the polar nature of the map. This provides a spatial model of nearby obstacles that allows the robot to reason about the free space around it.

3 Localization

Our localization system, Sensor Resetting Localization [3], uses a probabilistic model of the world to estimate the robots location on the field. The robots location is represented as a probability density over the possible positions and orientations, hereafter locations, of the robots. Since the probability density is in general a very complex function, we approximate the probability density by a set of sample points. The samples are chosen such that if $x\%$ of the samples are expected to be found in a particular area then the probability that the robot is in that area is $x\%$. Each sample point represents a particular location on the field at which the robot might be situated. Localization is the process of updating this probability density. To make the computation tractable, we make the Markov assumption that the robots future location depends only on its present location, the motions executed, and the sensor readings. Updates are done in such a way that the expected density of sample points in a region is proportional to the probability of the robot's location being within that region.

4 The World Model

The world model tracks the position of objects on the field when they were out of view. The positions of objects and their associated uncertainties are represented with two dimensional Gaussians. Sensor and motion updates are used to update the means and standard deviations (uncertainties) of these estimates. In addition to providing object positions, the world model also provides the robot with the velocity of the ball and how close the path of the ball will pass to the robot. This information is calculated by applying linear regression to a history of the ball position.

In addition to tracking information from their own sensors, each robot also incorporated information shared by its teammates. Twice a second, each robot broadcasts its own position and its position estimate for the ball. Teammates then incorporate these broadcasts into their own world models. This enables robots to track the ball even when it is not directly visible to them [5].

5 Behaviors

The CMPack'03 behavior system draws heavily from our 2002 team. Each behavior is represented as a finite state machine. State machines may themselves contain encapsulated state machines. Transitions between states are represented programmatically and may be arbitrarily complex. By formalizing the allowed state transitions and recording them as they occur, we are able to trace the execution of our behaviors, which aids debugging, and detect loops in our state transitions.

The robots playing offense are assigned three, separate roles. These roles are a *primary attacker*, which approaches the ball and attempts to move it upfield; an *offensive supporter*, which moves up the field from the primary attacker and positions itself to recover the ball if the primary attacker misses its shot on goal; and a *defensive supporter*, which positions itself down the field from the primary attacker to recover the ball if the other team captures it.

The role of primary attacker is assigned by passing a token between the robots to ensure mutual exclusion; two robots approaching the ball at the same time tend to become entangled. The primary attacker divides the other two robots between the two supporting roles when it first receives the token. Robots in supporting roles request the token from the primary attacker when they detect that they are better positioned than it to approach the ball. The primary attacker may choose to pass the token or refuse to hand it off.

Once they have been assigned roles, robots in supporting roles use a potential field to position themselves. This field encodes both obstacle information and heuristic information, such as robots should avoid blocking shots on their opponents' goal. The primary attacker moves directly to the ball and attempts to score or pass it to a teammate while avoiding opponents using Visual Sonar.

6 Motion

The motion system is given requests by the behavior system for high level motions to perform, such as walking in a particular direction, looking for the ball with the head, or kicking using a particular type of kick. The motion system for CMPack'03 is very similar to the design used in last year's competition.

The walking system implements a generic walk engine that can encode crawl, trot, or pace gaits, with optional bouncing or swaying during the walk cycle to increase stability of a dynamic walk. The walk parameters were encoded as a 49-parameter structure. Each leg has 11 parameters; the neutral kinematic position (3D point), lifting and set down velocities (3D vectors), and a lift time and set down time during the walk cycle. The global parameters were the z-height of the body during the walk, the angle of the body (pitch), hop and sway amplitudes, and the walk period in milliseconds. Using this interface, we developed a high performance walk with a maximum walking speed of 200 mm/sec forward or backward, 200 mm/sec sideways, or 2.3 rad/sec turning. Turning and translating have different optimal parameters, so we implemented a system for this year that allows us to switch parameters during a walk cycle while guaranteeing continuity. Additional motions, such as kicks can be requested from a library of motion scripts stored in files.

References

1. J. Bruce, T. Balch, and M. Veloso. CMVision (http://www.cs.cmu.edu/ jbruce/cmvision/).
2. S. Lenser, J. Bruce, and M. Veloso. CMPack: A complete software system for autonomous legged soccer robots. In *Autonomous Agents*, 2001.
3. S. Lenser and M. Veloso. Sensor resetting localization for poorly modelled mobile robots. In *Proceedings of ICRA-2000*, 2000.
4. Scott Lenser and Manuela Veloso. Much ado about Visual Sonar. In *Proceedings of IROS'03*, 2003. under submission.
5. Maayan Roth, Douglas Vail, and Manuela Veloso. A world model for multi-robot teams with communication. In *Proceedings of IROS'03*, 2003. under submission.
6. Ashley W. Stroupe and Tucker Balch. Collaborative probabilistic constraint-based landmark localization, 2002.
7. Ashley W. Stroupe, Kevin Sikorski, and Tucker Balch. Constraint-based landmark localization. In *RoboCup-2002: The Sixth RoboCup Competitions and Conferences*, 2002.
8. William Uther, Scott Lenser, James Bruce, Martin Hock, and Manuela Veloso. CM-Pack'01: Fast legged robot walking, robust localization, and team behaviors. In A. Birk, S. Coradeschi, and S. Tadokoro, editors, *RoboCup-2001: The Fifth RoboCup Competitions and Conferences*. Springer Verlag, Berlin, 2002, forthcoming.
9. Douglas Vail and Manuela Veloso. Dynamic multi-robot coordination. In Alan C. Schultz, Lynne E. Parker, and Frank E. Schneider, editors, *Multi-Robot Systems: From Swarms to Intelligent Automata*, volume 2, pages 87–98, 2003.
10. M. Veloso and W. Uther. The CMTrio-98 Sony legged robot team. In M. Asada and H. Kitano, editors, *RoboCup-98: Robot Soccer World Cup II*, pages 491–497. Springer Verlag, Berlin, 1999.
11. M. Veloso, E. Winner, S. Lenser, J. Bruce, and T. Balch. Vision-servoed localization and behavior-based planning for an autonomous quadruped legged robot. In *Proceeding of AIPS-00*, 2000.