Simulation League, Team *CMUnited99*, pages 7–11
http://www.ep.liu.se/ea/cis/1999/007/02/

The CMUnited-99 Simulator Team

CMUnited99

Peter Stone, Manuela Veloso, Patrick Riley

PS, MV, PR: Carnegie Mellon University

Abstract. *The CMUnited-98 simulator team became the 1998 RoboCup simulator league champion by winning all 8 of its games, outscoring opponents by a total of 66–0. CMUnited-99 builds upon the successful CMUnited-98 implementation, but also improves upon it in many ways. This paper gives an overview of the CMUnited-98 software and outlines our planned changes for CMUnited-99.*

1 Introduction

The CMUnited-98 simulator team became the 1998 RoboCup [1] simulator league champion by winning all 8 of its games, outscoring opponents by a total of 66–0. CMUnited-99 builds upon the successful CMUnited-98 implementation [4]

The main planned improvements in CMUnited-99 are

- parameter optimization for behavior mode and ball handling conditions;
- multi-step on-line planning;
- introduction of opponent modelling and prediction

2 Agent Architecture Overview

CMUnited-98 agents are capable of perception, cognition, and action. By perceiving the world, they build a model of its current state. Then, based on a complex set of behaviors, they choose an action appropriate for the current world state.

A driving factor in the design of the agent architecture is the fact that the simulator operates in 100ms cycles. The simulator accepts commands from clients throughout a cycle and then updates the world state all at once at the end of the cycle. Only one action command (dash, kick, or turn) is executed for a given client during a given cycle.

Since the simulator updates the world at the end of every cycle, it is advantageous to try to determine the state of the world at the end of the previous cycle when choosing an action for the current cycle. As such, the basic client loop during a given cycle t is as follows:

- Assume the client has consistent information about the state of the world at the end of cycle $t-2$ and has sent an action during cycle $t-1$.
- While the server is still in cycle $t-1$, upon receipt of a sensation (see, hear, or sense_body), store the new information in temporary structures. Do not update the current state.
- When the server enters cycle t (determined either by a running clock or by the receipt of a sensation with time stamp t), use all of the information available (temporary information from sensations and predicted effects of past actions) to **update the current state** to match the server's world state at the end of cycle $t-1$. Then **choose and send an action** to the server for cycle t.
- Repeat for cycle $t+1$.

Within the above framework, CMUnited-99 agents use the same approach to world modelling and low-level agent skills (predictive, locally optimal skills – PLOS) as did CMUnited-98 agents [4]. Agent skills include kicking, dribbling, ball interception, goaltending, defending, and clearing.

3 Coordination

Given all of the individual skills available to the CMUnited-98 clients, it becomes a significant challenge to coordinate the team so that the players are not all trying to do the same thing at the same time. Of course one and only one agent should execute the goaltending behavior. But it is not so clear how to determine when an agent should move towards the ball, when it should defend, when it should dribble, or clear, etc.

If all players act individually — constantly chase the ball and try to kick towards the opponent goal — they will all get tired, there will be nowhere to pass, and the opponents will have free reign over most of the field. The CMUnited-98 team uses several complex coordination mechanisms, including reactive behavior modes, pre-compiled multi-agent plans and strategies, a flexible teamwork structure, a novel anticipatory offensive positioning scheme, and a sophisticated communication paradigm as described in detail in [4].

In this section we outline our proposed improvements to coordination mechanisms within the CMUnited-99 team.

3.1 Parameter Optimization

A player's top-level behavior decision is its behavior mode. Implemented as a rule-based system, the behavior mode determines the abstract behavior that the player should execute. For example, there is a behavior mode for the set of states in which the agent can kick the ball. Then, the decision of what to do with the ball is made by way of a more involved decision mechanism. On each action cycle, the first thing a player does is re-evaluate its behavior mode.

The behavior modes are:

Goaltend: Only used by the goaltender.

Localize: Find own field location if it's unknown.

Face Ball: Find the ball and look at it.

Handle Ball: Used when the ball is kickable.

Active Offense: Go to the ball as quickly as possible. Used when no teammate could get there more quickly.

Auxiliary Offense: Get open for a pass. Used when a nearby teammate has the ball.

Passive Offense: Move to a position likely to be useful offensively in the future.

Active Defense: Go to the ball even though another teammate is already going. Used in the defensive end of the field.

Auxiliary Defense: Mark an opponent.

Passive Defense: Track an opponent or go to a position likely to be useful defensively in the future.

The detailed conditions and effects of each behavior mode were determined heuristically by hand in the CMUnited-98 implementation. We intend to tune them in a more principled way in our CMUnited-99 implementation, for example by treating them as an empirical parameter optimization problem.

Similarly, as described in [4], the ball-handling decision procedure that is used by an agent in possession of the ball was a heuristic rule-based system. The heuristics were hand-coded with a variety of adjustable parameters, which we intend to optimize.

These optimizations will be done with off-line training. An automated coach will run the a player or players through repeated trials. The coach will reset the ball and players and record relevant data from each trial. This data will be used to identify the tradeoffs in the parameters.

3.2 Multi-Step Plans

At the core of the CMUnited-98 coordination mechanism is what we call the Locker-Room Agreement [3]. Based on the premise that agents can periodically meet in safe, full-communication environments, the locker-room agreement specifies how they should act when in low-communication, time-critical, adversarial environments.

The locker-room agreement includes specifications of the flexible teamwork structure and the inter-agent communication paradigm [4]. A good example of the use of the locker-room agreement is CMUnited-98's ability to execute pre-compiled multi-agent plans after dead-ball situations. While it is often difficult to clear the ball from the defensive zone after goal kicks, CMUnited-98 players move to pre-specified locations and execute a series of passes that successfully move the ball out of their half of the field. Such “set plays” exist in the locker-room agreement for all dead-ball situations.

In CMUnited-99, we plan on adding the ability to execute multi-step plans on-line, as opposed to just after dead-ball situations. CMUnited-98 agents think in terms of what is the best 1-step pass to make, or how to move in support of the player that currently has possession of the ball. One of our primary research goals in creating CMUnited-99 is to add the ability to think at least 2 or 3 passes in the future: both when considering where to pass, and when considering where to move to receive a future pass. Ideally, we hope to demonstrate that agents can successfully play “keepaway” via short-term pass and movement planning.

4 Opponent Modelling

In CMUnited-98, individual decisions are often affected by the adversaries' behavior, but strategy decisions at the level of the entire team of the cooperating agents are for the most part fixed.

Ideally, agents should effectively adjust their group behavior in response to adversary actions. A first step in this endeavor is to be able to effectively classify adversary behavior into a group that determines what strategy change is appropriate. Our work in this area is built upon [2].

First we need to define relevant features of teams to separate them into behavior classes. Then, using these features, we will define possible classes of opponent behaviors. During a game, we need to calculate how well the current opponent fits into each behavior class using the information gathered about that opponent while playing. This updating process is one of the primary areas of research that will be undertaken.

The classes themselves can be built up from observations of opponents. We intend to use the logfiles from RoboCup-98 to help build up such classes of behaviors.

The coach agent introduced at RoboCup-99 offers the opportunity to gather global opponent information for classification. We intend to use the on-line coach to do the classification of the opponent. The coach then will communicate that classification to each of the players. Each agent will then change strategy in a predefined manner based upon the coach's classification.

5 Conclusion

The success of CMUnited-98 at RoboCup-98 was due to several technical innovations ranging from predictive locally optimal skills (PLOS) to strategic positioning using attraction and repulsion (SPAR). We plan to improve the CMUnited-99 agents by introducing more principled methods for determining agent behavior modes and ball handling options; multi-step on-line planning for offensive passing and positioning; and opponent modelling and prediction for improving team strategy.

References

[1] Hiroaki Kitano, Yasuo Kuniyoshi, Itsuki Noda, Minoru Asada, Hitoshi Matsubara, and Eiichi Osawa. RoboCup: A challenge problem for AI. *AI Magazine*, 18(1):73–85, Spring 1997.

[2] Patrick Riley. Classifying adversarial behaviors in a dynamic, inaccesible, multi-agent environment. undergraduate senior thesis at Carnegie Mellon University (Accessible from http://www.andrew.cmu.edu/˜priley/papers/thesis.ps.gz), 1999.

[3] Peter Stone and Manuela Veloso. Task decomposition and dynamic role assignment for real-time strategic teamwork. In J. P. Müller, M. P. Singh, and A. S. Rao, editors, *Intelligent Agents V — Proceedings of the Fifth International Workshop on Agent Theories, Architectures, and Languages (ATAL-98)*, volume 1555 of *Lecture Notes in Artificial Intelligence*, pages 293–308. Springer-Verlag, Heidelberg, 1999.

[4] Peter Stone, Manuela Veloso, and Patrick Riley. The CMUnited-98 champion simulator team. In Minoru Asada and Hiroaki Kitano, editors, *RoboCup-98: Robot Soccer World Cup II*. Springer Verlag, Berlin, 1999.