The UT Austin Villa 2005 Simulator Coach

Gregory Kuhlmann and Peter Stone

Department of Computer Sciences
The University of Texas at Austin
Austin, Texas 78712-1188
`{kuhlmann,pstone}@cs.utexas.edu`
`http://www.cs.utexas.edu/~{kuhlmann,pstone}`

Abstract. Due to the large changes in the structure of the RoboCup Coach Competition, the UT Austin Villa 2005 simulated online soccer coach employs a very different strategy from the one used in previous years. At the same time, much of the opponent modeling infrastructure used in our previous entries remains the same. Our research focus continues to be placed on online and offline machine learning. In this paper, we review the opponent modeling techniques that our coach uses and discuss this year's innovations.

1 Background

The UT Austin Villa[1] 2005 coach infrastructure is based on our previous year's entry, which was based on our 2003 entry [1], and operates as follows. Prior to a match, the coach examines the provided logfiles of games played by the team to be modeled. The coach collects data about players on both of the teams. In particular, it is able to identify high-level events, such as passes and shots on goal, from the low-level positional data. The following section describes how this is done.

1.1 Play by Play

From the coach's perspective, a game proceeds as a sequence of possessions. A possession change occurs whenever a new player gets the ball, there is a goal, or the play mode changes to one of several dead ball modes. When a possession change occurs, the coach analyzes the previous possession and attempts to characterize it.

Each possession consists of two parts. The first part, the *hold interval*, starts when the ball owner gains possession and persists until the last time the ball is within kickable range. The final cycle in this interval is called the *last kickable* time. The second part, the *kick interval* begins in the cycle following the last kickable time and ends at the next possession change.

If the ball moves a significant distance during the hold interval then the ball owner's action sequence is classified as a **dribble**. If the ball remains stationary, but the interval lasts several cycles, then the sequence is declared a **hold**. Otherwise, the player is said to have not performed any action at all.

[1] `http:/www.cs.utexas.edu/~AustinVilla`

If at the end of the kick interval, the ball is in the goal, then the ball owner obviously shot and scored a **goal**. If a teammate, gains possession, then the ball owner is said to have made a successful **pass**. Although, the next possessor may not have been the ball owner's intended receiver, we found that it was a safe assumption to make. If the play mode at the end of the interval is a dead ball mode, then the ball owner is said to have caused a **foul**. The coach does not distinguish between different types of fouls.

The most difficult case to interpret is a turnover. The coach considers four possibilities. If the ball is still a short distance away from the ball owner at the time of the turnover, then the the sequence is classified as a **steal** by the opponent. If the ball was headed for the goal and originated from a position reasonably close to the goal, then the kick is declared a **missed shot**. If the ball was headed for a teammate within a reasonable distance of the ball owner, the kick is assumed to be an **intercepted pass**. If the kick cannot be classified as any the above categories, then it is called a **clear**.

2 Learning

In the past, the events collected by the game-analysis module were used to build opponent models for the purpose of advice. Models of the fixed opponent were used to predict their actions and hopefully thwart them. Models of the opposing teams that played well were used to directly create advice that our players could mimick. Formational advice was also learned from the opposing players. Both offensive and defensive advice were created using decision trees as the agent models.

3 Innovations

This year, we plan to again use decision trees to model the agents behavior based on the the game analysis input. However, because the rules of this year do not focus on exploiting opponent flaws, but just detecting them, the models will be used for their descriptive rather than predictive power. Models learned offline from the flaw pattern log files will be labeled with the corresponing flaw name.

During game play, models will be built in the same way from the online game analysis. When the models match those labeled with the flaw from the offline phase with sufficient confidence, the flaw pattern will be reported. We will experiment with several different distance measures for decision trees to find one that works well in this scenario.

In addition, we will be exploring several different methods for how to give advice to our own players in order to discover the patterns of the opposing team most efficiently. Currently, we plan to do this by mimicking the other team represented in the log files to try to trigger similar behavior in the opposing team.

Acknowledgments

This research is supported in part by NSF CAREER award IIS-0237699.

References

1. Gregory Kuhlmann, Peter Stone, and Justin Lallinger. The champion UT Austin Villa 2003 simulator online coach team. In Daniel Polani, Brett Browning, Andrea Bonarini, and Kazuo Yoshida, editors, *RoboCup-2003: Robot Soccer World Cup VII*. Springer Verlag, Berlin, 2004.