Headless Chickens III

Paul Scerri, Johan Ydrén, Tobias Wiren, Mikael Lönneberg and Pelle Nilsson

Institute for Computer Science, Linköpings Universitet

1 Introduction

The development of the Headless Chickens III emphasized a high level team specification environment, called the Strategy Editor, that was intended for use by endusers, rather than computer programmers[2]. Using the strategy editor consisted of placing players on a image of the ground and indicating the direction(s) the player should kick and/or dribble when they get the ball. Different player formations and passing/dribbling patterns could be specified for different game situations. The designer could also specify the style of play for each of the players, e.g. defensive or inclined to shoot or dribble.

The Strategy Editor proved to be an effective tool primarily because a "player template" could be loaded into the editor. The template specified the different modes of play the players knew about, the different styles of play the player could play and the different actions the player could take. The mechanism allowed parallel development of the low level aspects of the players behavior (developed by agent experts) and high level strategies (developed by domain experts).

Specifications made with the Strategy Editor were "compiled" into separate behavior based agents. The core of the behavior based agents runtime engine had been previously developed for earlier versions of the Headless Chickens[1].

The Headless Chickens III finished equal 5th in the 1999 World Cup competition. They were involved in some of the more exciting games of the competition including the first ever World Cup overtime game which HCIII eventually won two to one. However HCIII were clearly inferior to the best teams losing seventeen nil to CMUnited99 and eleven nil to MagmaFreiburg.

2 Team Development

Team Leader: Paul Scerri
Team Members:

Paul Scerri
- Linköpings Universitet
- Sweden
- Graduate Student
- Attended Competition

Johan Y'dren
- Linköpings Universitet
- Sweden

M. Veloso, E. Pagello, and H. Kitano (Eds.): RoboCup-99, LNAI 1856, pp. 576–579, 2000.
Springer-Verlag Berlin Heidelberg 2000

- Masters Student
- Attended Competition

Tobias Wiren, Mikael Lönneberg and Pelle Nilsson
- Linköpings Universitet
- Sweden
- Undergraduate Students
- Attended Competition

Web page http://www.ida.liu.se/˜pausc/RC99/main.html

3 World Model

The agent architecture is split into two layers, one for skills and one for strategies. Although there is some basic information processing that is used at both layers, e.g. calculating the velocity of the ball, the way world information is presented to the different layers is quite different.

The skills layer uses world information almost directly from the sensors. Some low level calculations are done to ensure the players view of the world remains reasonably accurate between sensing cycles and after acting cycles. There is some very simple reasoning done so that objects that have been seen previously but are no longer in view are maintained in memory unless the player is looking at where the object was last seen.

The strategy layer uses world information only as abstracted fuzzy predicates. A separate Java class is associated with each fuzzy predicate. The class uses sensor information to assign a value between one and one hundred to predicates such as *the ball is close* or *defensive position*. The higher the value of the predicate the more the predicate seems to be true. Some of the predicates, for example *near ball*, have some "memory" so that the value of the predicate is still reasonable when sensor values are unable to determine its value, e.g. when the ball can be no longer seen the predicate *near ball* retains it's previous value.

4 Communication

The HCIII do not use communication between agents. It was not found to be necessary either from an individual player perspective or from a team perspective.

The reactive nature of the agents, i.e. a behavior based architecture, is well suited to having limited local information, hence there is little need for inter-agent communication about object locations.

From a coordination point of view it is the responsibility of the team designer at design time to ensure that players will be in appropriate positions at particular stages of a game to ensure that team behavior "emerges". The "emergent" team behavior does not require communication. The team designer also specifies the preferred directions for players to pass and dribble so communication is not required for that either.

5 Skills

The skills of HCIII are relatively simple. Every cycle one skill is called, perhaps with some parameters, and gets the chance to execute one action. A flag is passed to the skill indicating whether or not it was the skill that executed the previous action but it has no guarantees that it will get to execute the next action. Niether does the skill have any idea of the higher goal that the skill is part of achieving.

The algorithm for intercepting the ball looks for the closest position where the player can meet the ball. A loop calculates the expected position of the ball for subsequent cycles and for each cycle checks whether the player can reach that position within that time. There are special cases for when the ball is coming directly at the player or moving very slowly.

Because a skill can only execute one action at a time the dribble is also very simple. The agents action depends only on the position of the ball and the point that the agent should dribble to. The kick parameters are calculated by first working out where the agent wants the ball to be in two cycles then working out the power and direction to get the ball there.

The goalie is a special agent that attempts to maintain a position a certain distance along a line between the center of the goal and the ball. Any time the ball comes into the penalty box the goalie chases the ball. Once he has the ball he will wait (spinning around to watch as much of the field as possible) until finding a good player to pass to. If no good passing option is found within some time the ball is kicked hard towards the sideline.

6 Strategy

The strategy of the HCIII can vary greatly from game to game. The strategy of the team is defined in two parts, using two different graphical development systems, the individual strategy editor and the team strategy editor.

The individual strategy editor, as the name suggests, defines the strategies of a single player. In effect it defines a template of a player which will be instantiated for a particular team strategy. At a high level of abstraction the individual strategy determines the different "modes" of play that the player will react to. Example modes are *Before kick off*, *Deep defense* and *Transition to attack.* Within each of the modes the individual strategy determines the styles of play the agent has for that mode, for example *waiting* before kickoff or *crossing* from attack. Several different "styles" of behavior can be defined for one mode, for example a defensive style and an attacking style. Which particular style the player will have is determined in the team strategy editor. At a lower level of abstraction the individual strategy defines aspects of a player such as its preference for kicking with respect to dribbling, how long the player is willing to lose sight of the ball before searching for it and how keen a player is to attempt to intercept an opponent.

The team strategy editor allows an enduser to quickly instantiate the templates created in the individual strategy editor into a team configuration. In the strategy editor aspects such as where each player should be in each mode, the positions to kick to and the directions to dribble as well as the style of play for each of the players is specified.

The combination of the two strategy editors allows a great deal of flexibility. During the World Cup our development team would watch logfiles of prospective opponents and create specialized strategies for each team. A prime example was the opponent YowAI. It was realized that there were certain team tactics that would keep most of YowAI's team offside most of the time. The tactics were quickly specified in the team strategy editor without recourse to the individual behavior editor. Using the newly created tactics and having players prefer dribbling meant that HCIII had a large amount of ball possesion (bad stamina management meant that high ball possesion was not turned into a good score). A less successful strategy was against CMUnited99. It was clear that CMUnited99 were far better than HCIII and a loss was inevitable so a very aggressive strategy was employed in a (vain) attempt to be the first team to score against CMU in two years. Alas the aggressive strategy resulted only in making CMUnited's margin of victory more pronounced.

7 Conclusion

There are two teams planned for RoboCup2000 based on the HCIII. One of the teams will use the same team specification system and a similar agent architecture but port the agent runtime architecture to C++ (from Java). Porting the agent code is aimed at improving the efficency of the team. The other team planned for RoboCup2000 will also use the same team strategy editor but will use a slightly different, though still very reactive, agent architecture. The new agent architecture is aimed at providing facilities for more intelligent agent decision making, in particular the ability to simultaneously attend to multiple high level goals.

References

1. Paul Scerri, Silvia Coradeschi, and Anders Törne. A user oriented system for developing behavior based agents. In *Proceedings of RoboCup'98*, Paris, 1998.
2. Paul Scerri and Johan Ydrén. *RoboCup-99: Robot Soccer World Cup III*, chapter End User Specification of RoboCup Teams. Springer, 1999.