Kouretes 2007

Department of Electronic and Computer Engineering
Technical University of Crete
Chania, 73100 Hellas (Greece)

I. Kouretes

According to Greek mythology, when Zeus was born on the mountain Ida in Crete, his mother Rhea was frightened that his father Cronus would hear him crying. Cronus made it practice to swallow and eat each of his children as soon as they were born to insure his safety. Rhea entrusted the guardianship of her son to the Daktyls (fingers) of Ida, also known as the Cretan Kouretes. These male vegetation daemons dressed in armor would “fight” outside the cave where Zeus was hidden, clashing spears and shields and keeping time to a rhythmic stamping of their feet. These sounds drowned out the infant Zeus’ cries and prevented his discovery by Cronus.

It is said that one of Kouretes, named Herakles (perhaps the earliest embodiment of the later hero Herculis), originated the Olympic Games by instigating a race among his four brothers. These brothers of Herakles (the thumb) were Paeoneus (the forefinger), Epimedes (the middle finger), Iasios (the ring finger), and Idas (the little finger). The five Kouretes brothers were working together like a hand. In the present days, Kouretes still fight together as one team, as one hand. Our AIBO robots (Paeoneus, Epimedes, Iasios, Idas) supported by an enthusiastic human team (Herakles) strive to become as successful as their ancestors in their own fight, the RoboCup Competition.

II. Team Information

The team Kouretes celebrates one year of existence. The team was formed in February 2006, soon after the arrival of the four AIBO robots at the Technical University of Crete. The team had its first exposure to the RoboCup event in 2006 in Bremen, Germany, where it participated to the technical challenges competitions. Even though it scored low, the team gained a tremendous amount of experience. Further information about the team, including pictures and movies from various events, may be found at www.intelligence.tuc.gr/kouretes.

The team of Kouretes 2007 includes eight members of the Department of Electronic and Computer Engineering.

1) Michail G. Lagoudakis, Assistant Professor (Team Leader) [Behavior and Strategy]
2) Petros Patelis, Graduate Student [Color Table, Localization]
3) Georgios Pierris, Undergraduate Student [Color Table, Localization]
4) Georgios Kontes, Undergraduate Student [Behavior and Strategy]
5) Souzana Vologi, Undergraduate Student [Landmark Recognition - Goals]
6) Andreas Panakos, Undergraduate Student [Landmark Recognition - Goals]
7) Alexandros Paraschos, Undergraduate Student [Landmark Recognition - Beacons]
8) Chrysavgi Kontogeorgou, Undergraduate Student [Landmark Recognition - Beacons]

III. Team Software

Last year our team used mainly the Universal Real-time Behavior Interface (URBI) combined with C++ to develop the robot software. This choice turned out to be quite limiting in the sense that everything had to be developed from scratch, from the simplest image processing to the basic walking gait. However, given the limited time and experience at the time, it was probably the only reasonable choice.

This year we decided to use existing RoboCup code as a starting point for our software, so that we could focus our development efforts on a higher level (coordination and strategies). We downloaded, studied, and tested code from various four-legged teams, including German Team, rUNSWift, UPennalizers, DutchAIBO Team, Cerberus, and SPQR-Legged. Our criteria for making a choice were not based on the past performance of these teams, but rather on the usability and simplicity of the code they offered. In order to minimize the necessary reverse engineering task on our side, we tried to avoid the complexity stemming from mixing various programming languages, such as C, C++, Java, xABSL, Perl, Matlab, in the same architecture, and we also preferred releases which were portable
to our machines. We finally decided to base our code on the SPQR-Legged 2006 code which is itself based on the German Team 2004 code. Despite the lack of documentation, the code of SPQR-Legged 2006 is written entirely in C++, is fully functional, uses techniques which we plan to extend, and is accompanied by useful debug tools.

We have invested a significant amount of time in understanding the base software architecture and we have already modified several modules to adapt to the league rules for 2007. We have also designed a new coordination method\(^1\) on top of the current planning mechanism. The following table shows the origin of the modules we currently use (the brackets indicate modules we plan to work on in the next few months).

<table>
<thead>
<tr>
<th>Module</th>
<th>SPQR-Legged 2006</th>
<th>Kouretes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ColorTableModule</td>
<td>SPQR2006</td>
<td>SPQR2006 + Kouretes2007</td>
</tr>
<tr>
<td>SensorDataProcessor</td>
<td>GT2004</td>
<td>GT2004</td>
</tr>
<tr>
<td>BallLocator</td>
<td>GT2004</td>
<td>GT2004</td>
</tr>
<tr>
<td>TeamBallLocator</td>
<td>GT2004</td>
<td>GT2004</td>
</tr>
<tr>
<td>ObstaclesLocator</td>
<td>GT2004</td>
<td>GT2004 + [Kouretes2007]</td>
</tr>
<tr>
<td>SelfLocator</td>
<td>SPQR2006</td>
<td>SPQR2006 + Kouretes2007</td>
</tr>
<tr>
<td>RobotStateDetector</td>
<td>GT2004</td>
<td>GT2004</td>
</tr>
<tr>
<td>CollisionDetector</td>
<td>GT2004</td>
<td>GT2004 + [Kouretes2007]</td>
</tr>
<tr>
<td>MotionControl</td>
<td>GT2004</td>
<td>GT2004</td>
</tr>
<tr>
<td>WalkingEngine</td>
<td>SPQR2006</td>
<td>SPQR2006</td>
</tr>
<tr>
<td>HeadControl</td>
<td>SPQR2006</td>
<td>SPQR2006 + [Kouretes2007]</td>
</tr>
<tr>
<td>LEDControl</td>
<td>GT2004</td>
<td>GT2004</td>
</tr>
<tr>
<td>GetupEngine</td>
<td>SPQR2006</td>
<td>SPQR2006</td>
</tr>
<tr>
<td>PlayersLocator</td>
<td>[Kouretes2007]</td>
<td></td>
</tr>
<tr>
<td>PassingEngine</td>
<td>[Kouretes2007]</td>
<td></td>
</tr>
<tr>
<td>CoordinationEngine</td>
<td>[Kouretes2007]</td>
<td></td>
</tr>
</tbody>
</table>

IV. TEAM RESEARCH

The team’s research interests stem from two fields of autonomous robotics: multi-agent coordination and learning. Therefore, the main focus of the team is primarily on team strategy.

A. Multi-Agent Coordination

The ability to coordinate within a team is a crucial factor to the success of the team. In years 2006 and 2007 the passing challenge in the four-legged league was specifically designed to promote research efforts in coordinated team play. The ability of successful passing, even though it could be seen as a single robot skill, underlies any form of team coordination in robot soccer. Given an accurate passing mechanism, team coordination in a soccer game extends to various team formations and strategies depending on the actual state of the game. Players can choose between different ways of positioning themselves on the field in order to move the ball faster, trick the opponents, and eventually score goals.

Team formations, tactics, and strategies is largely an unexplored area in the four-legged robocup research. In our first steps as a robocup team we decided to take a radical step in behavior control and implement robot soccer strategies which are based on human soccer strategies used in real soccer games. Taking under consideration that the ultimate goal of RoboCup is a game between robots and professional human football players, we believe that our current and ongoing work takes a step towards this goal.

Each team in a human soccer game consists of eleven players; one of them assumes the role of the goal keeper, whereas the others can freely move around the field. The most popular team formation is the so-called 4-4-2 formation which is shown in Figure 1 (a). FC Porto won the Champions League in 2004 (under coach Jose Mourinho) playing mostly with this formation.

In a four-legged robot soccer game each team consists of only four players, one of them being the goal keeper. In order to adapt the 4-4-2 system, we assume that the robot field corresponds to the center of an actual soccer field and we focus on the middle diamond formation. Since we lack a player in the middle we organize the players in a triangle formation. The actual positions are different for offence and defense as shown in Figure 1 (b), (c).

\(^1\)G. Kontes and M. Lagoudakis. *Coordinated Team Play in the Four-Legged RoboCup League*. Forthcoming RoboCup Symposium paper.
In our work we consider four different tactics for applying the aforementioned formation: two for offence (Counter Attack and Passing Attack) and two defense variation (Pressing Defense and Passive Defense). The decision to switch between tactics is made dynamically during the game.

In real soccer, when a defending team wins the ball, it strives to pass the ball from the Defenders to the Midfielders and from the Midfielders to the Attackers. Meanwhile, the Attackers and the Midfielders try to spread along the wings of the field, in order to “open” the opponent’s defense and score more easily. In Robocup, we chose two ways of implementing this tactic. In the Counter Attack, the robot which wins the ball becomes the Attacker, the robot that is further ahead becomes the Midfielder and the remaining robot becomes the Defender. Alternatively, according to the Passing Attack, the robot which wins the ball passes the ball to the robot further ahead which becomes the Attacker. The robot located further back becomes the Defender and the remaining robot becomes the Midfielder. In both cases, the tactic switches from Defense to Offence.

In real soccer, when the attacking team loses the ball all the teammates, except the attackers, try to get themselves behind the ball. Everyone, including the attackers, is pressing for a mistake in the opponent team. Under some circumstances, even the attackers move behind the ball line when the team’s tactics is strongly defensive. In Robocup, we implement the switch from Offence to Defence in two different ways depending on the current state of the game. If the Attacker loses the ball near the opponent’s goal the tactic switches to Pressing Defense. If the Attacker loses the ball around the middle of the field or near the team’s goal, the tactic switches to Passive Defense.

For any given tactic, each player has a role in the field which can change dynamically during the game depending on the current player and ball locations. In our work, we consider three roles: Attacker (A), Defender (D), and Midfielder (M). Each role has a different behavior depending on the tactic followed at the time as follows.

**Counter Attack**
- **Attacker (A)** The Attacker dribbles with the ball towards the opponent’s goal straight from the place he won the ball. When he reaches near the opponent’s goal area he either shoots directly to score (if possible) or passes to the Midfielder who waits on the opposite side of the field at the corner of the penalty area. If such a pass occurs the Attacker and the Midfielder switch roles.
- **Midfielder (M)** The Midfielder supports the Attacker, positions himself at the opposite side of the field, in particular at the corner of the penalty area, and waits for a pass from the Attacker to shoot. If he receives a successful pass, he becomes the Attacker.
- **Defender (D)** The Defender stays always at the center of the field, just past the center circle. In the extreme case where the ball bounces back to him, he shoots towards the opponent’s goal.

**Passing Attack**
- **Attacker (A)** The Attacker waits for a pass from the Midfielder who has the ball and is located further back. As soon as the pass is successfully received, he advances forward and acts as in the Counter Attack above.
- **Midfielder (M)** The Midfielder dribbles with the ball towards the opponent’s goal from the place he won the ball, looking for opportunities to pass the ball to the Attacker. Once the pass is made, he advances and positions himself at the opposite side of the Attacker as above.
• **Defender(D)** The Defender stays near the center of the field as above.

Similarly, we define the roles in the Defense tactics. We implemented the roles of the players using Petri Nets Plans\(^2\), expanding the implementation of the SPQR-Legged team. We have used several basic actions to create the complex rational behaviors required by each role. The Petri Nets describing the roles for the Attacker and the Midfielder in the Counter Attack tactic are shown in Figure 2.

The tactic to be played at any moment is selected dynamically depending on the current situation of the game. Coordination between the robots take place through message broadcasting according to the following rules.

- Any robot winning the ball signals the team to choose an offensive tactic.
- Any robot loosing the ball signals the team to choose a defending tactic.
- The robots use the same protocol for selecting a tactic.
- Deadlock is avoided by initiating tactic switching from a single signal.
- The default tactic is Passive Defence.

A small portion of the selection algorithm is given as pseudocode below.

```
if (incoming_signal == ball_won) {
    if (ball_position == OppField) {
        tactic = Counter Attack;
        switch(signal_sender) {
            case Attacker: break;
            case Midfielder: { Midfielder <- Attacker; Attacker <- Midfielder; break; }
            case Defender: break;
        } else {
            tactic = Passing Attack;
            switch(signal_sender) {
                case Attacker: { Midfielder <- Attacker; Attacker <- Midfielder; break; }
                case Midfielder: break;
                case Defender: { Defender <- Midfielder; Midfielder <- Defender; break; }
            }
        }
    } ...
```

Given that the above protocol is not centralized, it is robust to penalizations and/or hardware failures. Our approach, in order to work flawlessly, must be supported by an accurate localization module and an accurate passing module (only for the offending tactics). In addition, the players must have the ability to identify the opponents and the teammates at any time and communicate constantly through the wireless network.

A recent trend in multi-agent coordination borrows methods and techniques from economics (auctions, trading) and applies them to agent problems. We plan to replace our simple role allocation method with an auction-based method to handle the complexity of additional formations and tactics. Our team leader was a member of a research group which proposed a framework for automatic derivation of bidding rules and proved the first theoretical results for auction-based agent coordination\(^3\). Although these auctions have been tested in simulated domains, they have not been applied to real robots. The team Kouretes is the vehicle for bringing these methods to the real-world.

The most significant advantage of our approach is the fact that it is derived from real soccer theory which has been developed and tested over several years in the real soccer fields. Such approaches have been mostly used in the Simulation League, but not in the Four-Legged League. It is also an important step towards the ultimate goal of Robocup, the game between humans and robots.

\section*{B. Multi-Agent Learning}

Reinforcement learning has been used widely in many robotic applications mostly in a single-agent form. The multi-agent versions have not been adopted widely due to difficulties associated with efficiency and scaling to realistic domains. Recent research work by our team leader and collaborators has led to extensions of the Least-Squares Policy Iteration (LSPI) algorithm to collaborative multi-agent learning (where many agents learn to collaborate as a team)\(^4\) and competitive multi-agent learning (where two teams learn to compete against each other, but collaborate within the team)\(^5\).

The scaling properties of these algorithms through exploitation of domain knowledge make them attractive for the RoboCup domain. We plan to use reinforcement learning in order to let the robots learn the cost function needed in an auction-based method for team coordination. We are currently working on modeling the learning problem with the appropriate state space and reward function. Factorization of the representation can be done on the basis of the proximity between players during a game. Again, Kouretes is our venue for adapting these algorithms and testing their potential in a difficult task with real-time constraints.

\section*{C. Localization}

Given the new rules for the 2007 league, our localization method relies on using the only stable landmarks in the field (two beacons and two goals). The absence of the pink color makes it harder to distinguish beacons from goals and the structure of the new goals requires the recognition of a colored frame rather than a filled colored rectangle. Our method, based on the Particle Filter localizer of SPQR-Legged 2006, is not yet reliable enough, especially in areas outside the diamond formed by the four landmarks. We are working along two directions to improve it. On one hand, we are focusing on improving landmark recognition by examining collectively the proportions of colored and non colored parts in the scanlines. On the other hand, we are considering using the field lines as additional landmarks to improve accuracy.

\section*{V. Travel Support}

Our team is partially supported by a Marie Curie European research grant. Additionally, we are currently negotiating with the institutional research office and various external sponsors to secure additional funding, sufficient for our participation to RoboCup 2007 in Atlanta in case of qualification.

\section*{VI. Extra Material}

A collection of video clips is located at \url{http://www.intelligence.tuc.gr/kouretes/ROBOCUP2007}

\section*{VII. Acknowledgements}

We would like to thank Fr. Demetrios Alexandrakis and the parish of Panagia in Kounoupidiana for providing valuable laboratory space to the team when it was most needed.

