

# Clockwork Orange: The Dutch RoboSoccer Team

Matthijs Spaan<sup>1</sup>, Marco Wiering<sup>2</sup>, Robert Bartelds<sup>3</sup>, Raymond Donkervoort<sup>1</sup>,  
Pieter Jonker<sup>3</sup>, and Frans Groen<sup>1</sup>

<sup>1</sup> Intelligent Autonomous Systems Group, University of Amsterdam, the Netherlands

<sup>2</sup> Intelligent Systems Group, Utrecht University, the Netherlands

<sup>3</sup> Pattern Recognition Group, Delft University of Technology, the Netherlands

## 1 Introduction

The Dutch RoboSoccer Team, called “Clockwork Orange”<sup>1</sup> is a Dutch research project in which 3 universities participate: the University of Amsterdam, the Delft University of Technology, and University Utrecht. In this research project we study multi agent systems in general and multi robot systems in particular, as we participate in the RoboCup middle-size league. Due to severe hardware problems Utrecht University could sadly not contribute to the team this year.

## 2 The robots

Our team consists of two types of robots: six Nomad Scout robots (used by Delft and Amsterdam) and one Pioneer 2 robot (used by Utrecht). The Nomad Scout (see figure 1(Left)) has odometry sensors and one wide angle colour camera for sensing and a pneumatic driven kick device as effector. Its 16 ultrasonic sensors and its tactile bumper ring are not used. The Pioneer 2 (see figure 1(Right)) uses a laser range finder, 16 ultrasonic sonar sensors, odometry sensors, and a camera for sensing. It uses the same pneumatic kick device as effector. For communication between robots we use a Breezecom wireless Ethernet system.

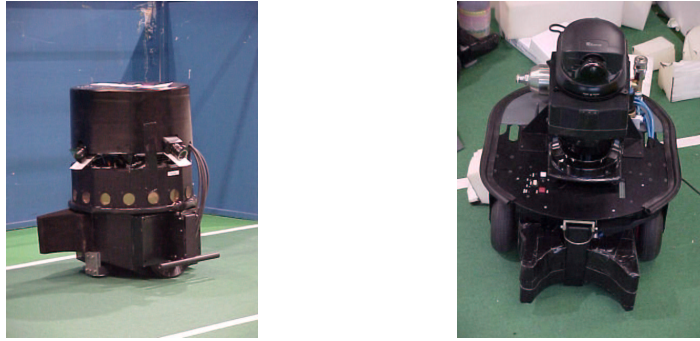
## 3 Vision

Our vision research is mostly directed at the problems of recognition and self-localisation. We can detect and track all objects typically found in a RoboCup environment[5]. This is done via color detection by circular arcs in an adapted HSI space.

We have implemented 2 different ways for self-localisation, a local and a global method (see [3] for details). Both are based on detecting the lines of the field and the goals. The global method first splits the screen into multiple regions of interest and does a rough Hough transform on each of these. Then the method

---

<sup>1</sup> Nickname of the human Dutch national soccer team of the seventies.



**Fig. 1.** Our robots. Left: a Nomad Scout robot. Right: the Pioneer 2 robot.

searches for the precise lines using linear edge detectors in the vicinity of the lines found by the Hough transform. These are matched to the world model giving an estimate of the position. Because of the symmetry of the field multiple candidates are found. We then use Multiple Hypothesis Tracking to follow all the candidates over time, updating them for our own movement. Since there are multiple candidates now we need a quick method to verify these and to correct for changes.

The local method uses a first order Taylor approximation of the displacement of the line, independent in  $x$ ,  $y$  and  $\phi$ , to calculate the displacement in robot coordinates. It uses the least square error to give an overall result on the match. We can repeat the method to see if it converges. Next we check all the candidates, verifying their heading and distance to the goals and the overall result given to us by the local method. The loop is repeated until one candidate remains, who is used to update our position. If no candidates remain we switch back to global self-localisation.

## 4 World model

To enable a distributed form of control, the world model is also distributed. Each of the robots in our team locally maintains a world model consisting of the robot's own position and the positions of the other objects in the game (i.e. other robots and the ball). To communicate the world model between the robots we need an absolute world model for which we need to know our own position. This position is based on odometry and vision-based self-localisation. Both have their advantages and disadvantages. Odometry is very accurate for small distances but is not suitable for longer periods of driving and completely loses its value when the robot collides. Vision self-localisation can give us quite a good position estimation at any time (assuming it sees enough lines) but is very expensive in time and resources and its positions arrive with considerable delay. The world model combines these two sensors to get an estimation of our position, taking into account the lag of the self-localisation [4].

The other objects are detected by the vision system as shapes, which the world model matches to the known objects which have the same shape and are closest to the observed shape. From this information we can form a relative model of the world, which we can easily transform knowing our own position, into an absolute one. This information is shared with the other robots using the wireless Ethernet, so each robot can not only keep track of objects it perceives itself but also of objects perceived only by its teammates.

## 5 Team skills

The advantage of a absolute shared world model is that team coordination is greatly simplified. Sharing the same world model enables the team members to reason not only about their own actions, but they can also predict the next action a team member will take. Communication on the team coordination level is used to improve robustness and is almost a necessity since the team is heterogeneous both in hardware and software. The Utrecht Pioneer 2 robot uses a subsumption architecture while the Amsterdam/Delft Nomad Scouts operate on a hybrid architecture. Our team uses a global team behavior (a strategy) from which each robot derives its individual role.

Our team skills module determines the current team behavior using a simple finite state machine. Each player has its role defined by the team behavior. There are about half a dozen different roles available, all of which have an attacking, defending or intercepting purpose. Utility functions based on potential fields [7, 6] are used to distribute the roles among the team members. This technique is similar to the ones used by other participants in the middle-size league [2, 9]. Each robot calculates its own scores for the roles useful at the moment, communicates them and then calculates the scores of its team mates. When no scores from other team mates are received the locally calculated scores can be used. This introduces a level of redundancy which can improve the team performance. When everybody knows the scores for the roles of each team member the roles can be easily distributed.

The exact specification of an role is not (and can not be) the same on both architectures. The following describes the Amsterdam/Delft approach (see [4] for details) which is similar to the one Tambe [8] described for use in the simulation league. A Markov decision process is used to search through state space of a robot's future states. First a number of possible actions and their parameters with respect to the current situation and individual behavior are generated. For each of these actions the probability of success is calculated. The impact on the world of an action and the expected concurrent actions of team mates and enemy players is evaluated. Modeling team mate actions is relatively easily compared to modeling your opponent with reasonable accuracy. The evaluation criteria both include a role evaluation (using potential fields) as well as concepts like ball possession. Now the action with the maximum expected utility can be chosen and the process can be repeated to look further ahead.

## 6 Behavior based control

The Pioneer 2 robot of Utrecht University uses an extended version of the subsumption architecture [1] in which particular behaviors such as *Get\_Ball*, *Dribble*, and *Score* compete for controlling the robot. All behaviors can react in correspondence to the world model or to direct sensor data provided by the camera and laser range finder. In future work, we extend the system by integrating learning as part of the behavior-based control architecture of the robot. This is done by providing the robot with different implementations of the same behavior. These behaviors are tested during online play against an opponent. If a particular behavior is successful against an opponent team, the behavior gets strengthened, and is therefore used more often. This allows us to learn a sequence of behaviors which has the highest expected probability of success against an opponent.

## 7 Conclusion

We were able to maintain a distributed world model using vision-based self-localisation. This information was used to create high level team coordination. Clockwork Orange reached the quarter finals of RoboCup 2001 to be defeated by the world champion. In total seven games were played, resulting in three victories, one draw and three losses.

## References

1. R.A. Brooks. Achieving artificial intelligence through building robots. *MIT AI Lab Memo 899*, 1986.
2. C. Castelpietra, L. Iocchi, M. Piaggio, A. Scalzo, and A. Sgorbissa. Communication and coordination among heterogeneous mid-size players: ART99. In *Proceedings of the 4th International Workshop on RoboCup*, pages 149–158, 2000.
3. F. de Jong, J. Caarls, R. Bartelds, and P. P. Jonker. A two-tiered approach to self-localization. In *RoboCup 2001*. Springer-Verlag, 2002.
4. F.C.A. Groen, J. Roodhart, M. Spaan, R. Donkervoort, and N. Vlassis. A distributed world model for robot soccer that supports the development of team skills. In *Proceedings of the 13th Belgian-Dutch Conference on Artificial Intelligence (BNAIC'01)*, 2001.
5. P. P. Jonker, J. Caarls, and W. Bokhove. Fast and accurate robot vision for vision-based motion. In *RoboCup 2000: Robot Soccer World Cup IV*, pages 72–81, Melbourne, Australia, 2000.
6. O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. *International Journal of Robotics Research*, 5(1):90–98, 1986.
7. J. Latombe. *Robot Motion Planning*. Kluwer Academic Publishers, 1991.
8. M. Tambe and W. Zhang. Towards flexible teamwork in persistent teams: Extended report. *Journal of Autonomous Agents and Multi-Agent Systems*, 3(2):159–183, 2000.
9. T. Weigel, W. Auerbach, M. Dietl, B. Dümler, J. Gutmann, K. Marko, K. Müller, B. Nebel, B. Szerbakowski, and M. Thiel. CS Freiburg: Doing the right thing in a group. In *RoboCup 2000: Robot Soccer World Cup IV*. Springer-Verlag, 2001.