Small Robots League, Team *Rogi2Team*, pages 64–71
http://www.ep.liu.se/ea/cis/1999/006/11/

Rogi Team Description

Rogi2Team

Josep Lluis De la Rosa, Rafael Garcia, Bianca Innocenti, Israel Munoz, Albert Figueras, Josep Antoni Ramon

JD, RG, BI, IM, AF, JR: Girona University

Abstract. *This paper describes the main features of the new Rogi Team and some research applied, focused on dynamics of physical agents. It explains the vision system, the control system and the robots, so that the research on dynamical physical agents could be performed. It presents part of the research done in physical agents, especially consensus of properly physical decisions among physical agents, and an example applied to passing.*

1 Introduction

The Rogi Team started in 1996 as the result of a doctorate course in multiagent systems. In 1997 and 1998 it took part at the international workshops held in Japan and Paris. The main goal of the team has been always the implementation and experimentation in dynamical physical agents and autonomous systems. This year a step further towards the platform to develop this type of research is done.

1.1 New Features in the 1999 Generation

A new team for RoboCup-99 has evolved from past generations, has solved many important problems existing in previous versions, and is now more focused on dealing with dynamical physical agents. Here, dynamical is understood from the viewpoint of automatic control and means dynamics of robots. This generation is designed to let further study the impact of dynamics of the agent's bodies in the co-operative world.

The robots have been improved and its structure has changed in order to have a better-fit dynamical behaviour for control. Now we have good transfer functions. The new vision system is able to process up to 50 frames/sec, locating ten robots and a ball, with a dedicated hardware result of our research. A rational physical agent's approach is operative for robots co-operation, for instance, applied in passing actions. This is also result of our research.

2 Team's Description

The system that implements micro-robot soccer is made up of three parts: robots, vision system and control system. The vision and control systems are implemented in two different computers. The control system is also called the host system. The control system and the vision system are connected by means of a LAN, using TCP/IP protocol. This fact allows remote users to perform tests over the micro-robots platform and lets co-operative research [4]. The vision system provides data to the control system that analyses the data and takes decisions. The decision is split up into individual tasks for each robot and is sent to them using a FM emitter at the host computer.

2.1 Vision System's Description

A specific hardware has been designed to perform the vision tasks, merging specific components for image processing (video converters, analogue filters, etc.) with multiple purpose programmable devices (FPGAs). A real time image-processing tool is obtained, which can be reconfigured to implement different algorithms. The goal is to have a fast tracking system by colour segmentation that permits clear dynamic control of the robots, up to 50 images per second.

The vision system is based on an overhead camera looking down at the playing field, providing a global view to track the robots on the field. According to RoboCup F-180 League Rules, each robot has to be marked using a yellow or blue Ping-Pong ball mounted at the centre of their top surface. In order to provide angle orientation, additional colour markings are al-located to the top of the robots. As shown in the following figure, purple and olive-green colour patches have been added as the orientation patches in a basic configuration for our team robots. However, the robots may incorporate additional colour patches to be distinguished from each other.

To locate the robots and the ball, the first step consists in their segmentation from the scene. The discrimi-natory properties of two colour attributes, hue and satura-tion, are used so as to segment the objects. In this way, a pixel labelling of the image is obtained, assigning a different label to pixels belonging to different colour textures. Moreover, a more robust behaviour under non-uniform lighting of the scenario is achieved, thanks to the stability of hue and saturation under variations on the intensity of the illuminant [2].

Since the size of the objects to track is rather small (5-6 pixels of diameter) mathematical morphology is applied in a 33 neighbourhood of the processed pixel. Erosion and dilation operations are performed at video rate by using the tracking processor. In this way, particles smaller than the tracked patches are removed from the image. The next step consists in the classification of the remaining blobs. The presented relates those blobs that may belong to the same object using the knowledge of the robot-patches geometry. Then, the position and orientation of the 5 robots is computed. Given that all the team robots use the same colours as orientation patches, it is difficult to distinguish between them. The vision system overcomes this problem through a two-step approach: Firstly, the robots are codified by means of additional color markings, having special shapes which are difficult to detect at video rate. This identification requires expensive

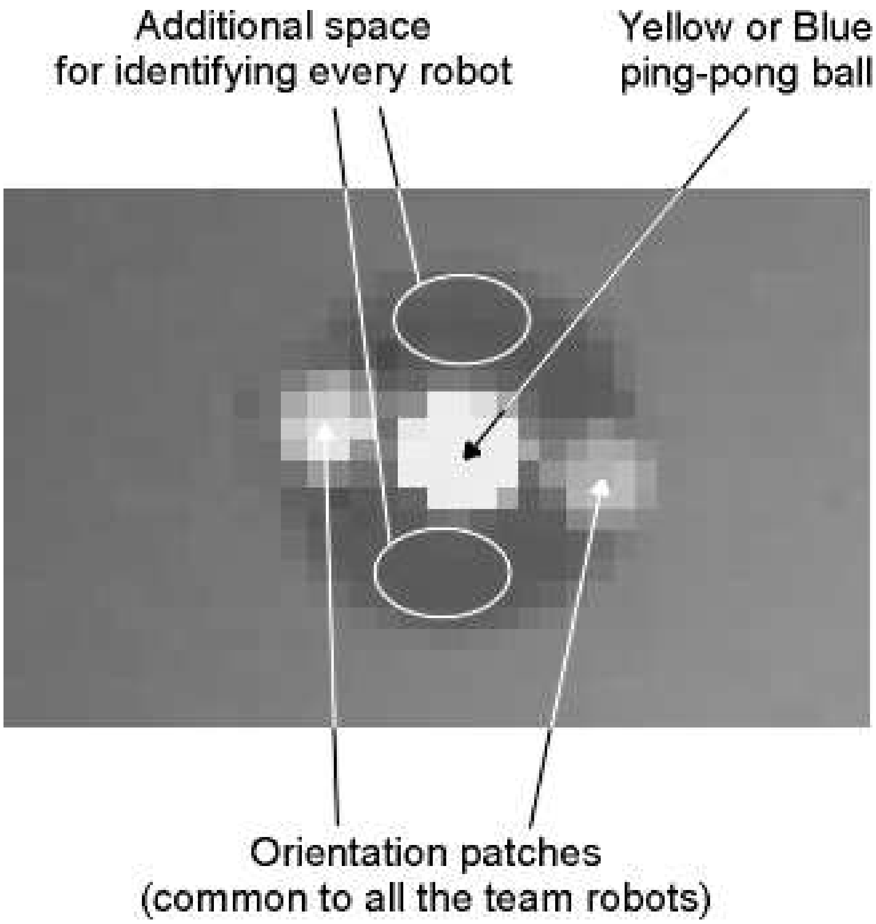

Figure 1: Fig. 1. Top view of the robot, showing the team-coloured Ping-Pong ball and the orientation patches.

computation, which is only performed in the first image of the tracking sequence by the identification process. Therefore, the set-up image takes a long time to be computed, but the robots can be identified in an automatic way. The position and orientation of every robot in the first image is then stored. Secondly, in the following images of the sequence, only the position and orientation of the robots will be computed, regardless of which is robot 1, or robot 2, etc. Thus, an unordered set of positions and orientations is provided by the tracking hardware. A data association process solves the temporal matching problem at the highest abstraction level. However, the identification process is executed periodically to check that the data association has kept track properly of the robots' locations.

As a last (but not least) step, the measured locations of the objects are filtered in a sequence of im-ages by means of an Extended Kalman Filter. The low-level image processing operations are performed in hardware, while identification, data association, post-filtering and prediction are implemented in software. The cycle time of the algorithm is 20 ms, being limited by the PAL video standard.

2.2 Robots' Description

The robots have on board 8 bits microprocessors 80C552 from Philips and RAM/EPROM memories of 32kBytes. The robots receive data from the host computer by means of a FM receiver. The FM receiver is prepared to work with two frequencies 418/433 MHz in half-duplex communication.

The information sent by the host computer is converted to RS-232C protocol. The two motors have digital magnetic encoders with 265 counts per turn. They need 9V to work and consume 1.5W at a nominal speed of 12.300 rpm. 9 batteries of 1.2 V supply the energy. It is compensate to have clear dynamics, which is non-linear but linear piece-wise.

2.3 Control System

The control system contains the strategy and the team decision making. The control system, using the positions of the robots and the ball provided by the vision system, has to determine which is the best decision to score a goal or to prevent the opponent team from scoring a goal. The system has been implemented using the industrial Lab-Windows. There are some advantages,

A very easy graphic interface to compute and to work with. Ease to compute using C language. High level commands that make easier to compute some vital tasks (using COM to send data to robots, compute the sample time, etc....).

On the other hand some drawbacks have come up using it,

Some difficulties to work in real time. The sampling rate can be as low as 0.1 seconds and the operative system is not able to send data at this rate. The lack of object oriented language to better approach to agents' implementations.

Hereafter a new control system is developed using Agent Oriented Programming (AOP). However, a C++ Object Oriented Programming (OOP) language is used since it is a convenient current and practical first approach to AOP.

3 Research Challenge: taking dynamics into account.

The research challenge in this paper stressed is about dynamics of physical agents. The research of the proper vision system will be demonstrated in RoboCup Workshop but not explained in detail in this paper. Explicit reasoning on dynamics of the physical body of agents will prevent from undesirable situations. Also, knowing that a controller modifies (controls) dynamics of the physical body of agents, here agents are proposed to be aware (introspection) of the set of controllers their physical body has. Control engineers need of tools for developing this type of agents and their controllers, as stated in [3].

AGENT0 [6] is used as an agent language. In this language an agent is described as an entity whose state consists of mental components such us beliefs, capabilities, choices and commitments. In our point of view, the capabilities are precisely the ones that let us represent the dynamics of the system. We believe that some of the capabilities are associated to the control of the system and we proposed them as a way for the agent to be aware of what it can or cannot do. An extension of the agent concept from physical agent [1] to dynamical physical agent is the following: The physical knowledge of the physical agent is obtained from dynamics of the physical body, which is represented by a further declarative control level and a further declarative supervision level [5]. We assert that this knowledge has to be declared by means of capacities.

As a first approach, our research is applied to ball passing between two robots. This experiment is simplified as follows: two robots have crossing trajectories that are two trajectories with a common point that is the meeting point of one robot and the ball. We consider that there are no obstacles in the trajectories. The robots have several controllers to move forward in a one-dimensional linear movement. We know the transfer function of the robots and the ball.

Let us show this new knowledge through the following example that shows the utility of inter-agent negotiation in terms of dynamic behaviour and consists of improving braking decisions. Steps will be as follows:

1 Model and simulate dynamics of two autonomous mobile robots and the ball. 2 Implement position and speed controllers for passing. A specification is to reach the setpoints with precision and stability. 3 Inspect untargeted situations: not enough impulse for the ball. 4 Implement some negotiation procedures by means of dynamics represented in capacities inspired from Agent0, and inspect the results.

There are passes that are not physically feasible since there aren't controllers to execute them. For instance, to do a shot pass could be an extremely difficult task if there is no slow speed controller, and the same happens at any speed setpoint required where no controller exists.

The undesirable situations could be:

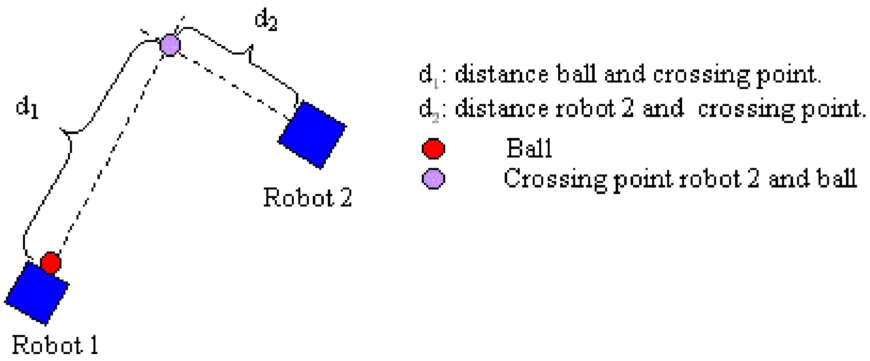

Figure 2:

(1) the robot 2 have slow dynamics to be in the crossing point at the same time that the ball (2) the robot 1 does not give the necessary impulse to get ball to the crossing point in the convenient time. This impulse has to be calculated according to the dynamics of the ball.

Both robots have to agree the type of control to apply for the pass and the moment to execute it based on the knowledge they have of their dynamics. Our proposal is to ensure passing between the robots by proper physical decisions.

Let us consider the following situation of passing:

do the pass, robot 1 will inject an impulsion (kick) to the ball and robot 2 will catch it in the crossing point. The ball has to arrive there at the same time that robot 2, so the amplitude of the impulse will depend on d1 and on the way the ball responds to this impulse (its transfer function). It also will depend on dynamical movement of robot 2 to arrive to the meeting point, this means that it depends on d2 and on the transfer function of robot 2. Furthermore, both robots must agree on the time. The way to do it is modifying the parameters of the controllers. Once agreed, the robot 1 must kick the ball and robot 2 must be in the crossing point at the same time that the ball.

Our aim is to demonstrate that most of the time undesirable situations can be prevented if the dynamics of the systems is considered when deciding. Figure 2 shows a trial of the ball to an impulse input, and figure 3 the distance run by robot 2 and the ball, their crossing point and the agreed time.

4 Results

Partial preliminary results of passing are available in the ftp:

- http://eia.udg.es/ peplluis/backpass.avi
- http://eia.udg.es/ peplluis/forwardleftpass.avi
- http://eia.udg.es/ peplluis/forwardrightpass.avi
- http://eia.udg.es/ peplluis/cornerpass.avi

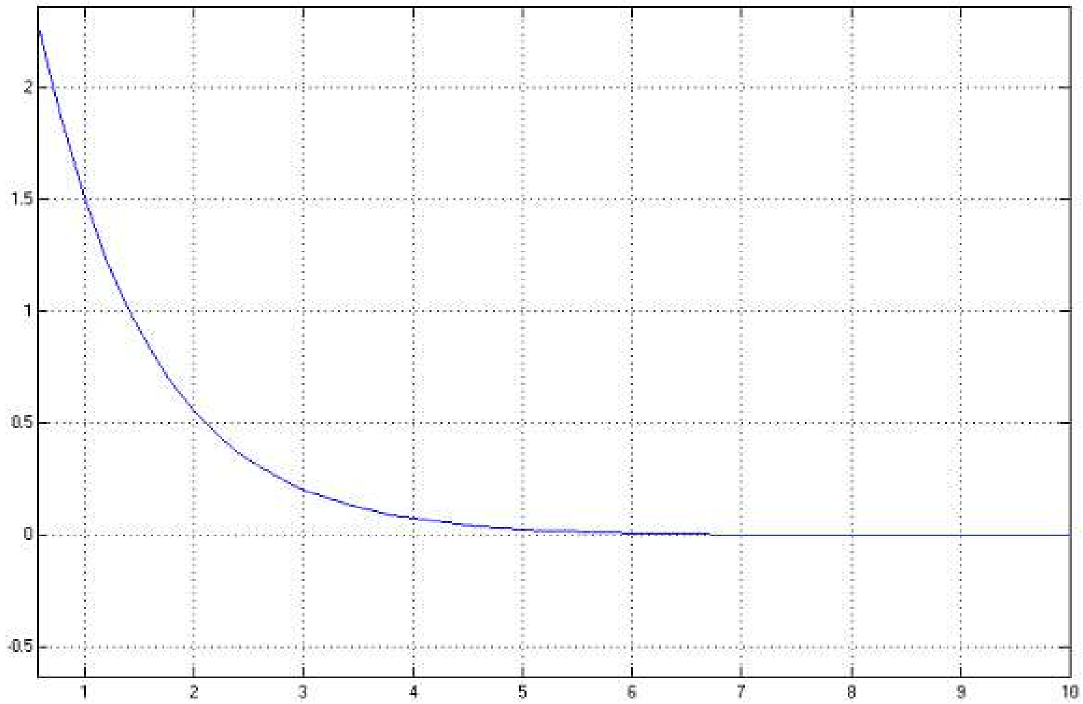

Figure 3:

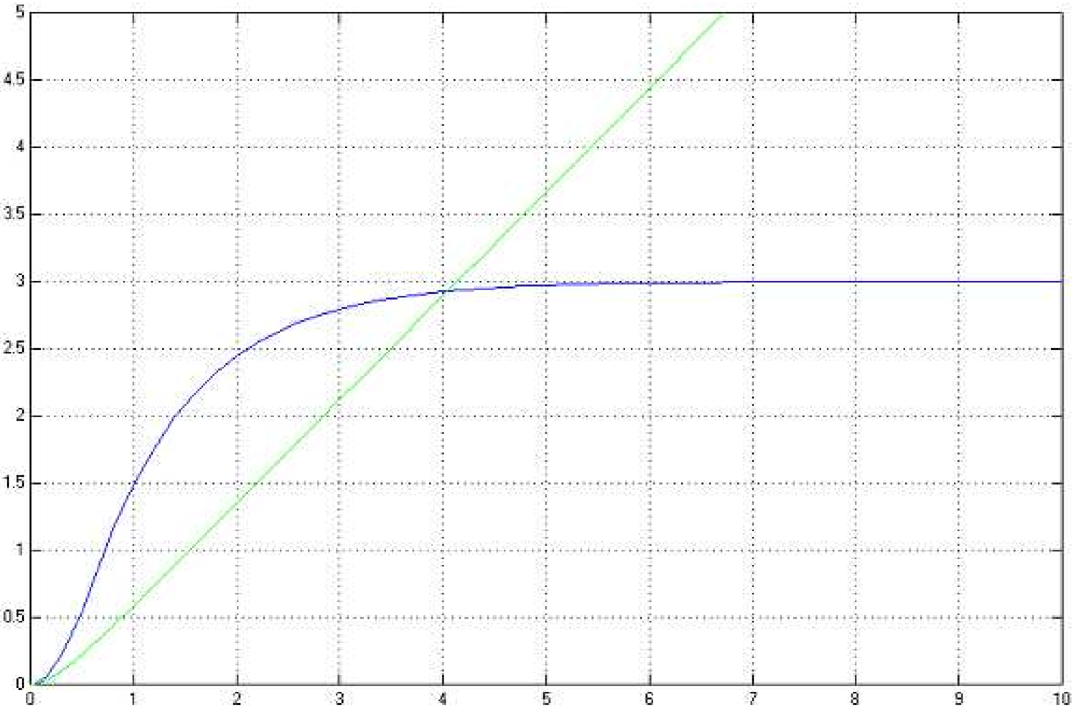

Figure 4:

The way the system is playing in the current state is shown in

- http://rogiteam.udg.es/descrobots.html#sequencies,

and also consult the following

- http://eia.udg.es/ peplluis/shortgame.avi
- http://eia.udg.es/ peplluis/longgame.avi

5 Bibliography

[1] Asada M., Kuniyoshi Y., et al. "The RoboCup Physical Agent Challenge", First RoboCup Workshop in the XV IJCAI-97, pp.51-56, 1997.

[2] Celenk M., "A color clustering technique for image segmentation", Computer Vision, Graphics and Image Processing, vol. 52, pp. 145-170, 1990.

[3] de la Rosa J. Ll., et al., "Soccer Team based on Agent-Oriented Programming", Robotics and Autonomous Systems. Ed. Elsevier, Vol 21, pp.167-176, October 1997.

[4] Johnson J., de la Rosa J. Ll., and Kim J.H., "Benchmark Tests in the Science of Robot Football" Proceedings IEEE of Mirosot-98, Paris 1998. R. J. Stonier (ed). Univ. Central Queensland.

[5] Mller J., "The Design of Intelligent Agents: a layered approach", Lecture Notes in Artificial Intelligence, Vol. 1177, Ed. Springer Verlag, 1996.

[6] Shoham Y. "Agent-oriented programming", Artificial Intelligence, vol. 60 (1), pp.51-92, 1993 & Technical Report STAN-CS-1335-90, Computer Science Department, Stanford University, Stanford, CA, 1990.