Robotics and Autonomous Systems 61 (2013) 115–124

Contents lists available at SciVerse ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot

# Intelligent state changing applied to multi-robot systems

Tiago P. Nascimento [a,*], António Paulo Moreira [a], André G. Scolari Conceição [b], Andrea Bonarini [c]

[a] INESC TEC (formerly INESC Porto) and Faculty of Engineering, University of Porto, rua Dr. Roberto Frias, 4200-465 Porto, Portugal
[b] LaR - Robotics Lab, Department of Electrical Engineering, Polytechnic School, Federal University of Bahia (UFBA), Rua Aristides Novis, 02 Federação - Salvador - BA, Brazil
[c] Dipartimento di Elettronica e Informazione, Politecnico di Milano, Via Ponzio, 34/5, 20133, Milan, Italy

A R T I C L E  I N F O

*Article history:*
Received 10 April 2012
Received in revised form
7 October 2012
Accepted 15 October 2012
Available online 10 November 2012

*Keywords:*
Fuzzy automaton
Multi-robot systems
Mobile robots

A B S T R A C T

The target searching problem is a situation where a formation of multi-robot systems is set to search for a target and converge towards it when it is found. This problem lies in the fact that the target is initially absent and the formation must search for it in the environment. During the target search, false targets may appear dragging the formation towards it. Therefore, in order to avoid the formation following a false target, this paper presents a new methodology using the Takagi–Sugeno type fuzzy automaton (TS-TFA) in the area of formation control to solve the target searching problem. The TS fuzzy system is used to change the formation through the modifications in the states of the automaton. This change does not only switch the rules and therefore the state of each robot, but also the controllers and cost functions. This approach amplifies the versatility of the formation of mobile robots in the target searching problem. In this paper, the TS-TFA is presented and its implications in the formation are explained. Simulations and results with real robot are presented where it can be noticed that the formation is broken to maximize the perception range based on each robot's observation of a possible target. Finally this work is concluded in the last section.



## 1. Introduction

Fuzzy systems and automatons are often applied in robotics. Several research proposals use fuzzy systems to control, navigate or evaluate robot systems and the approach most frequently used is the Mamdani Fuzzy System [1–4]. Moreover, robotic soccer has become a good work-bench for testing artificial intelligence algorithms and it plays an important role in the progress of the intelligent control algorithm field [5–8]. In order to avoid deviating the formation from following a false target, this paper presents the use of a recently presented technique, the Takagi–Sugeno Type Fuzzy Automaton (TS-TFA) [9]. The TS-TFA selects the roles (behaviors) applied to the formation control of three 5 dpo soccer robots [10]. These robots, usually participants of RoboCup [11], are used here as a platform to demonstrate the performance of a formation during the search of a target when there is the commuting absence/presence of this target. The problem here lies in the fact that the target is initially absent. During the target search, false targets may appear dragging the formation towards it. Therefore, as a proposed solution for this problem, the fuzzy automaton contains three states which are sufficient in order to solve the problem in the proposed case of study. The fuzzy automaton does not only change the robots' role but it changes also the robots' cost function or controller (depending on the state change), while pursuing or searching for a target. A static leader approach was chosen for this work; nevertheless, the proposed contribution also works with a non-static leader approach. Simulations and experiments with real robots will also be presented in this paper.

The positioning of robots in a dynamic environment is also another issue regarding changes of roles in multi-robot systems [12,13]. In the work of Akiyama et al. [14], the authors proposed a novel agent positioning mechanism for the dynamic environments. They state that because the real-world problem is generally dynamic, suitable positions for each agent should be determined according to the current status of the environment. Therefore, the authors in [14] formalized this issue with a map from a focal point (like a ball position in a soccer field) to a desirable positioning of each player agent. Furthermore, they proposed a method to approximate this map using Delaunay Triangulation. The performance of the method was evaluated in RoboCup Soccer Simulation environment and compared to other function approximation methods such as the Normalized Gaussian Network.

Furthermore, Stone and Veloso [15] introduced periodic team synchronization (PTS) domains as time-critical environments in which agents act autonomously with low communication, but in which they can periodically synchronize in a full-communication setting. The two main contributions of this article were a flexible

* Corresponding author. Tel.: +351 915177752.
*E-mail address:* tpnascimento@gmail.com (T.P. Nascimento).

0921-8890/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.robot.2012.10.011

team agent structure and a method for inter-agent communication in domains with unreliable, single-channel, low-bandwidth communication. In the said paper, homogeneous agents can flexibly switch roles within formations, and agents can change formations dynamically, according to pre-defined triggers which are evaluated at run-time. This flexibility increases the performance of the overall team. Our teamwork structure further includes pre-planning for frequent situations. Second, the novel communication method is designed for use during the low-communication periods in PTS domains. Finally, they fully implemented both the flexible teamwork structure and the communication method in the domain of simulated robotic soccer, and conducted controlled empirical experiments to verify their effectiveness.

Regarding the dynamic change of roles, the work of Reis et al. [16], presents an interesting approach. In their paper, they proposed an approach for coordinating a team of homogeneous agents based on a flexible common Team Strategy as well as on the concepts of Situation Based Strategic Positioning and Dynamic Positioning and Role Exchange. The authors also introduced an Agent Architecture including a specific high-level decision module capable of implementing this strategy. Their proposal was based on the formalization of the idea of what a team strategy is for competing with an opponent team having opposite goals. Agent's reactivity was also introduced for appropriate response to the dynamics of the current situation. However, in their approach this was done in a way that preserves team coherence instead of permitting uncoordinated agent behavior.

Changing roles in a multi-robot formation system requires a high level of navigation and control. Role assignment, according to the robot's features, is a crucial step in the coordination of multiple heterogeneous robots. The research of [17] presents a strategy for picking heterogeneous players and forming a soccer team in the RoboCup simulation environment, which is a multi-robot coordination research platform. Using fuzzy evaluation and fuzzy inference, the authors identify the most suitable role in a soccer team for a given heterogeneous robot. In the games, the team that possessed this strategy as its bases showed a significantly better performance when compared to a team based on the previous hand-tuned solution.

In [18], the authors focus on the Middle Size Soccer Robot league (MSL) as well as on new hierarchical hybrid fuzzy methods for decision making and action selection of a robot. In article [18], the behavior of an agent was introduced, implemented and classified in two layers, the *Low Level Behavior* and the *High Level Behavior*. In phase one, the robot's situation is checked in order for a decision to be make on how to perform the required behavior. In the second phase, the team strategy, team formation, robot's role and the robot's positioning system are introduced. A fuzzy logic approach is adopted to provide the player with the best position to move based on the information given by the current state.

In all previously mentioned approaches it is assumed that the ball (target) is in the group observation range, where at least one robot is seeing the target. This situation does not occur in the target searching problem. To solve this problem, the TS-TFA was proposed and implemented. To describe the TS-TFA implemented here, this article is organized in the following order: The next section presents the state machine that represents the robot's possible behavior. Section 3 presents the TS-Type Fuzzy Automaton. The problem is formulated in the following section and in Section 5 the results obtained both in simulation and with real robots are presented. Finally, the conclusion is presented in the last section.

## 2. The state machine

The TS-TFA combines fuzzy sets and automaton theory. In one of the most popular applications of the automaton theory in robotics, a state machine is used to define a high level control (or navigation). This navigation system is based on the robot's change in behavior (e.g. changing from a standstill state to a moving state). In this paper, the multi-robot system performs a search and track task of a target (ball). In this task, the ball can be within the robot's sensor range or not. Therefore, the multi-robot system changes its behavior from three states that can be seen in the state machine shown in Fig. 1.

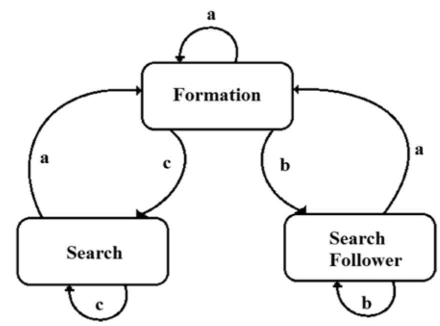


**Fig. 1.** Robot state machine.

To better understand the TS-FSA implications, the state machine that governs the robots' behavior has to be explained first. Here, a Deterministic Finite Automaton composed of three states governs the behavior of the robots in the target searching problem.

Therefore, consider the State Machine $SM = (Q, \Sigma, \Phi, Q_0, F)$ which can be seen in Fig. 1, where $Q$ is the set of all states, $\Sigma$ is the input alphabet, $\Phi$ is the state-transition function that correlates the alphabet with the transition between states, $Q_0$ is the set of initial states and $F$ is the set of final states. The following assumptions are made:

1. $Q := \{\text{Formation}, \text{Search}, \text{Search Follower}\}$
2. $\Sigma := \{a, b, c\}$
3. $\Phi := Qx\Sigma \rightarrow Q$
4. $Q_0 := Q$
5. $F \subseteq Q$.

Given $Z$ as the evaluation of the situation generated by the fuzzy system, and $L$ as the subscript identifying a robot $n$ (where $1 \leq n \leq N$ and $N$ is the total number of robots in formation) chosen to be the leader, the values for $\Sigma$ are:

$$\begin{cases} a, & \text{If } Z = 1 \\ b, & \text{If } Z = 0;\ n, \forall\, n \neq L \\ c, & \text{If } Z = 0;\ n, \forall\, n = L. \end{cases} \tag{1}$$

It is also known that $Q_0 = Q$ and $F = Q$. The *Formation* state is when the robot sees the target and starts moving towards it for improved observation and tracking. The *Search* state is when the leader robot ($n = L$) cannot see the ball and starts the procedure to search for a target, moving around the soccer field. The last state is the *Search Follower* applied to any robot where $n \neq L$ that cannot see the target. When the *Search Follower* is applied, the robot starts following robot $L$. In the *Formation* state, the robot uses the nonlinear model predictive controller (NMPC) [19]. This controller minimizes the cost function that dictates the robot's behavior and penalizes it if the formation does not perform as expected. When the *Formation* state is activated, the robot visualizes the target as a leader and starts moving towards it. The final position of the formation (in a three robot case) should put all of the robots around the ball, forming a 120° angle between them when the ball is not moving. In the *Search Follower* state, the robot also uses an NMPC but with a different cost function that considers the leader robot and not the target, thus it moves towards the leader robot. Finally, in the *Search* state, the robot changes to a normal reactive controller [20] using the global path planner $A^*$ [21] to move around the field.

The aim of this work is to use the resulting Output State value to change the state of the robot. The robot formation possesses three states, as previously seen. These three states are enough to solve the target searching problem where the change between absence/presence of a target becomes an issue. This is one of the main differences between this approach and the one in [9], where the authors use the Output State as a linear function instead of a value. In this paper the TS-fuzzy is simplified and instead of having an output fuzzy function, it has an output state. Therefore, if the resulting output state value is equal to 1, the resulting state is *Formation*(F), whereas, if the value is equal to 0 its resulting state is either *Search*(S) or *Search Follower*(SF) depending on the value of $Z$ from robot $n$, where $n$ is the robot's number. If the robot's number is the chosen one to be the leader ($n = L$) then the final state will be *Search*(S). If $n \neq L$, then the final state will be *Search Follower*(SF). This interaction can be seen in the look-up table below. The look-up table is a concise representation of the TS-TFA's behavior with respect to the state transitions defined through the dominant crisp states.

## 3. The Takagi–Sugeno type fuzzy automaton

Tracking the status of an event-driven, large control system is a difficult problem. These systems often encounter unexpected roadblocks in an uncertain environment. The use of a fuzzy automaton offers an effective approximation method to model continuous and discrete signals in a single theoretical framework using the combination of two techniques: the Automaton Theory and Takagi–Sugeno Fuzzy Systems. A Max–Min automaton can successfully model a cluster of relevant states when a decision must be made on the next state of a goal path at a supervisory level. However, to provide analytical proof for stability and other key properties of a fuzzy controller, a Takagi–Sugeno (TS) model is preferred. Finally, in [9] a TS-type fuzzy automaton is introduced.

The fuzzy automaton can remain in different fuzzy states simultaneously, to a certain degree in each. These degrees are defined by a state membership function. For each fuzzy state there is just one dominant (crisp) state for which the state membership is 1 (full membership). Each fuzzy state is associated with a linguistic label for inference. For each fuzzy state there is a fuzzy set that has a state membership degree greater than 0 in that fuzzy state [9].

The transitions between fuzzy states are based on the transitions defined between their dominant crisp states. There is an underlying Boolean finite state machine that implements the fuzzy automaton. The states of this Boolean automaton are the dominant crisp states of the fuzzy automaton. In the TS-TFA model, the Takagi–Sugeno version of the IF THEN rules are adopted [9]. Therefore,

$$R: \text{ If } x_1 \text{ is } A_1, \ldots, x_k \text{ is } A_k \text{ then } Z = g_{Sk} \tag{2}$$

and $g$ is a linear function that was defined in this work to assume only two values

$$g_{Sk} = 0 \quad \text{or} \quad g_{Sk} = 1. \tag{3}$$

In each crisp state $S_k$ the final output $Z_{Sk}$ is calculated according to

$$[Z_{Sk} = Z^i_{Sk}] = (A^i_1(x^0_1) \wedge \cdots \wedge A^i_k(x^0_k)) \wedge [R^i] \tag{4}$$

$$Z_{Sk} = \frac{\sum [Z_{Sk} = Z^i_{Sk}] \times Z^i_{Sk}}{\sum [Z_{Sk} = Z^i_{Sk}]} \tag{5}$$

where $[\cdot]$ means the truth value of proposition $\cdot$, the $\wedge$ operator is implemented as min and $A(x^0)$ stands for the grade of membership of $x^0$ in fuzzy set $A$. For simplicity, $[R^i] = 1$ is assumed. Furthermore, $\times$ means a scalar multiplication. The notion of the composite output $Z^*$ is introduced to reflect the contribution of the output values devised from the Takagi–Sugeno linguistic models that are attached to crisp states to the final output in a fuzzy state. Let the TS-TFA be in a fuzzy state $S_{Fk}$, then

$$Z^*_k = \frac{\beta^k_1 Z_{S1} + \cdots + \beta^k_p Z_{Sp}}{\sum_i \beta^k_i} \quad (i = 1, \ldots, p). \tag{6}$$

It is clear from Eq. (6) that only crisp states that have greater than 0 degrees of membership in fuzzy state $S_{Fk}$ contribute to the final output. Therefore, a TS-TFA automaton with $p$ states is defined as follows [9].

$$S_{Fk} : S_k, g_{Sk} \tag{7}$$

$$R_S = TS(X_F, Z_S) \tag{8}$$

$$G = \begin{bmatrix} \beta^1_1 & \cdots & \beta^1_p \\ \vdots & \cdots & \vdots \\ \beta^p_1 & \cdots & \beta^p_p \end{bmatrix} \tag{9}$$

$$Z^* = TS(X^0_F, R_S, G) \tag{10}$$

$$X_T \text{ is TRUE if } X_A \geq X_{AT} \tag{11}$$

$$X_B = B(X_F) \tag{12}$$

$$Y_B = f_y(X_B, W_B, X_T, y_B) \tag{13}$$

$$U_B = f_u(X_B, W_B, X_T, y_B) \tag{14}$$

where $S_{Fk}$ stands for fuzzy state $k$, $S_k$ represents crisp state $k$ and $g_{Sk}$ is the state membership function associated with $S_k(k = 1, \ldots, p)$. $\beta^k_1, \ldots, \beta^k_p$ stand for the degrees of state membership function $g_{Sk}$. $G$ stands for the matrix of state membership functions. The computational algorithm for Eq. (10) is given by Eqs. (4)–(6). Finally, $X^0_F$ stands for inputs values evaluated by the member function of the fuzzy sets ($A$). The variables $W_B$ and $X_A$ stand for two-valued (Boolean) and analog inputs with associated $X_{AT}$ threshold values, respectively. A threshold comparator module compares the value of each analog signal with its associated threshold value to set the corresponding $X_T$ signal as true or false. $U_B$ stands for two-valued (Boolean) outputs. $X_B$, $Y_B$ and $y_B$ stand for two-valued Boolean inputs, next states and the present states of the state variables, respectively.

## 4. Problem formulation

In order to avoid deviating the formation from following a false target, we present the use of the TS-TFA technique to select roles applied to formation control of a multi-robot system composed of three 5 dpo soccer robots [10]. The fuzzy automaton will not only choose the robot's role, it will also select the robot's cost function or controller (depending on the state change), while pursuing or searching for a target. In the 5 dpo robot soccer team, the architecture of each robot is composed of a Coach (a state machine that changes the robot's behavior) inside each computer that runs parallel with the Dec (controller) and the HAL (vision system). These systems are explained in more detail in [10]. The state machine used that governs the robot's behavior and a brief explanation of the TS-TFA theory have been presented to outline the proposed solution. In this work, Intelligent State Changing is applied in the Formation Control of Multi-Robot Systems to solve the target searching problem. To accomplish this task, a TS-TFA was applied using the state machine previously explained in this paper. This state machine is governed by a set of rules extracted from the TS fuzzy system. The switch between states performed by the fuzzy rules assumes that each state in the state machine is a possible scenario that the robot would be in each instant of time. In

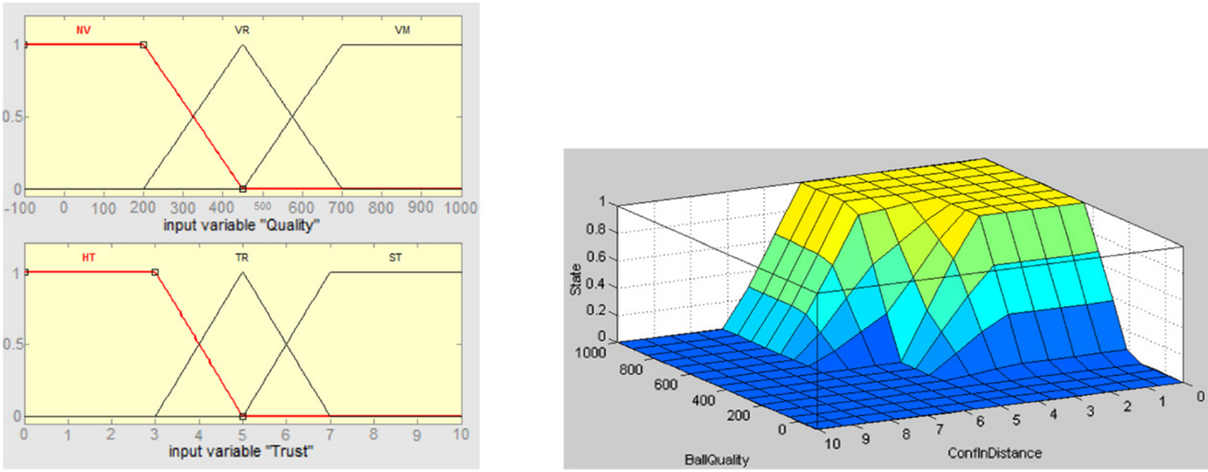


**Fig. 2.** Membership functions and surface.

turn, each scenario is the result of a fuzzy rule. Finally, each fuzzy rule performs a weighted calculation based on the measurements from the robots distance to the target and the robot's quality of the target perception measurements. In this TS model the implications (TS fuzzy rules) are as follows:

$$
\begin{aligned}
&R_1: \text{ If } x_1 \text{ is NV and } x_2 \text{ is HT then } Z = 0\\
&R_2: \text{ If } x_1 \text{ is NV and } x_2 \text{ is TR then } Z = 0\\
&R_3: \text{ If } x_1 \text{ is NV and } x_2 \text{ is ST then } Z = 0\\
&R_4: \text{ If } x_1 \text{ is VR and } x_2 \text{ is HT then } Z = 1\\
&R_5: \text{ If } x_1 \text{ is VR and } x_2 \text{ is TR then } Z = 0\\
&R_6: \text{ If } x_1 \text{ is VR and } x_2 \text{ is ST then } Z = 0\\
&R_7: \text{ If } x_1 \text{ is VM and } x_2 \text{ is HT then } Z = 1\\
&R_8: \text{ If } x_1 \text{ is VM and } x_2 \text{ is TR then } Z = 1\\
&R_9: \text{ If } x_1 \text{ is VM and } x_2 \text{ is ST then } Z = 0
\end{aligned}
\tag{15}
$$

where $x_1$ is the fuzzy input that represents the quality of the robot camera's visual on the target. A camera perception quality decay range that stretches from $-100$ to 1000 was created for the 5 dpo robot soccer team based on the number of pixels present in the target's visualization that represent the target [10]. Moreover, the input $x_2$ is the confidence in which this observation is made with respect to the distance between the robot and the estimated target position. If the robot is near the target, it means that its observation has high confidence and if it is far, it has low confidence. The visibility distance usually varies from 0 to 10 m (maximum) and this is the range of confidence in estimation.

Each robot in the formation measures the position and speed of the ball (the state of the ball) in local frame. This information is sent to the Coach software that is located in a central computer and merges the information originated from the robots into a single ball state in the world frame. Afterwards, that is shared among the robots in the formation through wireless communication. All robots' poses are also shared throughout the formation. It is assumed that the merging of the ball position is performed using Smith and Cheeseman's formulation [22] and it provides the fused data of the ball speed and position, as well as the quality of this data. In each rule, the weight given to each measurement (distance of quality of perception) can be seen through a set of membership functions, which in turn can be seen in 2.

### 4.1. Membership functions

The membership functions can be seen in Fig. 2. These membership functions are designed by considering the quality decay of target perception from the robot's vision software (HAL). It is important to notice that the present work does not focus on the vision system. The vision system delivers the blob of the target as a set of pixels with a center point that in turn is translated into a quality decay range. This range is used to create the membership functions to evaluate the quality of the observed target [23,24].

The Trust (confidence) function represents the reliability in the quality of the target's observation based on the distance between the robot and this observed target. Therefore, the Trust (confidence) function represents the reliability of the vision based on the distance to the target. If the distance is too far, the trust (confidence) is low but if it is close, the trust (confidence) is high. This membership function is rated in meters of distance. The membership functions are adjusted to the maximum performance by setting the functions proportionally, as per the implementation in real robots. In Fig. 2 for example, by analyzing the Trust function based on the robot's vision, it is noticeable that the value that separates the state of seeing and not seeing the target is 5. In the Quality function of target perception, the value is 450. In the Trust function, between 3 and 7 m the confidence in the visual is regular, for less than 3 m the confidence is strong and any measurement above 7 m is not considered. The same situation happens with the quality function of target perception.

### 4.2. The problem example

The example bellow was created to better understand the contribution of this paper and only to illustrate a situation where the existence of a false target would influence the formation's behavior. Here, robot 1 is the leader robot, while robots 2 and 3 are the followers. Robots 1, 2 and 3 depart from coordinates $(4.5, -3)$, $(4.5, -1)$ and $(6.5, -3)$, respectively. To simulate the appearance of a false target during the search process, the ball was initially placed in the coordinates $(2, 2.9)$. After a while during the simulation the ball is changed to the coordinates $(-4.5, 6)$ where the "true" ball would be located.

This simulates a situation where the robot or the whole formation sees a false ball and, getting closer to the observed object, discards it as the target (either by the object format or any other vision process that occurs with the color segmentations process used to recognize the ball). In this case, without the proposed approach, the formation would be as shown in Fig. 3. Note that in this example, one of the robots sees the false target and sends the information to the other robots. Then, all robots in formation try to converge towards this false target. When the

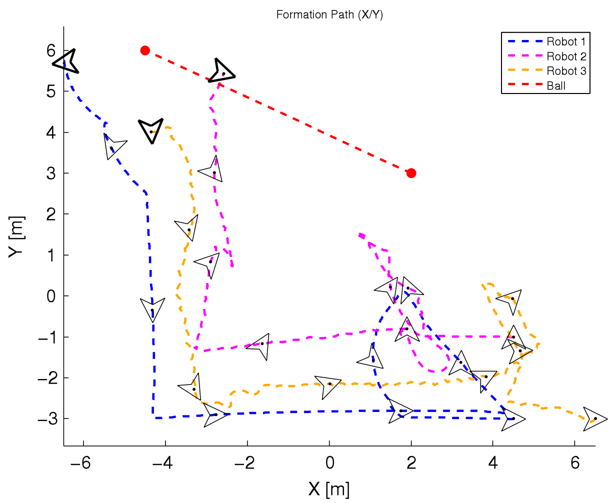


**Fig. 3.** The problem simulation: Formation behavior without the TSTFA approach.

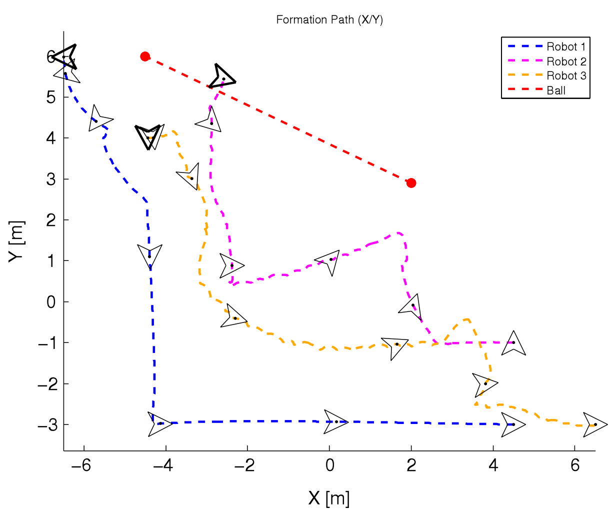


**Fig. 4.** The solution simulation: Formation behavior with the TSTFA approach.

formation approaches the observed object, the robots consider this as a false target and have to restart the search once more.

However, when applying the proposed technique, the behavior of the formation is improved. Instead of all the formation converging to a false target, only the robot that sees it weights between the quality of its observation and the distance that it is from the observed object. Then, this robot breaks itself from the formation trying to converge towards the possible target while the other robots keep searching. By analyzing Fig. 4, it can be seen that when the observed object is considered as a false target, the robot converges back to the formation which was still searching for a "true" target. Once the formation finds the "true" target, it converges successfully.

## 5. Results

The results are presented in two sections. The first outlines the simulation results using the SimTwo simulator [25]. These simulations were created with three omnidirectional 5 dpo soccer robots moving at a maximum velocity of 0.7 m/s. In the simulations the robots should achieve a final formation configuration around the ball, forming a 120° (degree) angle between them. The three simulations presented here are three

**Table 1**
Membership Functions-State Transitions. The values NV, VR and VM in the Quality (q) membership function means, respectively, *Not Visible*, *Visible Reasonably* and *Visible Much*, and HT, TR and ST in the Trust (t) membership function mean *High Trust*, *Trust Reasonably* and *Small Trust*, respectively.

| q | t | | |
|---|---|---|---|
| | HT | TR | ST |
| NV | S/SF | S/SF | S/SF |
| VR | F | S/SF | S/SF |
| VM | F | F | S/SF |

different starting situations: *Simulation* 1: *Robot* 3 *is relatively near to the target and robots* 1 *and* 2 *are relatively far from the target*, *Simulation* 2: *Robot* 2 *is relatively near to the target and robots* 1 *and* 3 *are very far from the target* and *Simulation* 3: *Robots* 1, 2 *and* 3 *are very far from the target*. Furthermore, the Section 5.2 presents the results of the experiment with real robots, where the real 5 dpo soccer robots were used. The experiment was only conducted with the last simulation case to validate this theory. As a safety measure, the maximum robot velocity allowed was 0.7 m/s.

### 5.1. Simulations

This section presents all three simulations. It is important to remember that here the ball is always stopped, the robots move at velocities of up to 0.7 m/s and robot 1 is the robot leader.

#### 5.1.1. Simulation 1

In this simulation, the ball was placed at the coordinates (2, 3.5) and robots 1, 2 and 3 were placed at the coordinates (4.3, −3.1), (4.3, −1.8) and (4.3, 1.6), respectively. All robots have an initial $\theta = 270°$ in world frame. Fig. 5 shows that robot 3 maintained its initial state *Formation* and moved towards the ball using the NMPC, without ever changing its state. Then, robot 1 receives *Search* as its initial state with an $A^*$ path planner and a reactive controller and it kept this state until it reached the coordinates (−3.2, 2.8) where its confidence was high enough to change its state to *Formation*. Finally, robot 2 receives *Search Follower* as its initial state with a NMPC controller with the cost function to follow a leader. Meanwhile, as the formation moves, robot 2 passes through coordinates (3.5, −2) where it was close enough to the target to change its state to *Formation*. If other robots see a target, each robot weighs up the global quality of the ball perception and its distance from the target. If the robot is too far from the target, it will not change its state, as shown in Table 1. Here, the change in state corresponds to a change in the controller.

By analyzing Fig. 5, it can be seen that there is no disturbance in the robots' trajectories or any presence of instability in the state changes. The controller is the same but the cost functions are different and once again, no instability could be noted. The distance to the target and their path in an *XY* plot can be seen in Fig. 5, respectively. The behavior of the robots in this situation can be seen in Fig. 6 where it is possible to see the change in states during each instant taken from the Coach's view of the simulations.

#### 5.1.2. Simulation 2

Here, Fig. 7 shows, through the plot *XY*, that the ball was placed at the point with coordinates (−5.3, 0) and robots 1, 2 and 3 were placed at the coordinates (4.3, −3.1), (−6.3, 0) and (4.3, 3.1), respectively. Furthermore, all robots have an initial $\theta = 90°$ in world frame. Note in the *XY* plot from Fig. 7 that there is no disturbance in the robot's trajectory and there is no presence of instability in the state change. In the distance graph from Fig. 7,

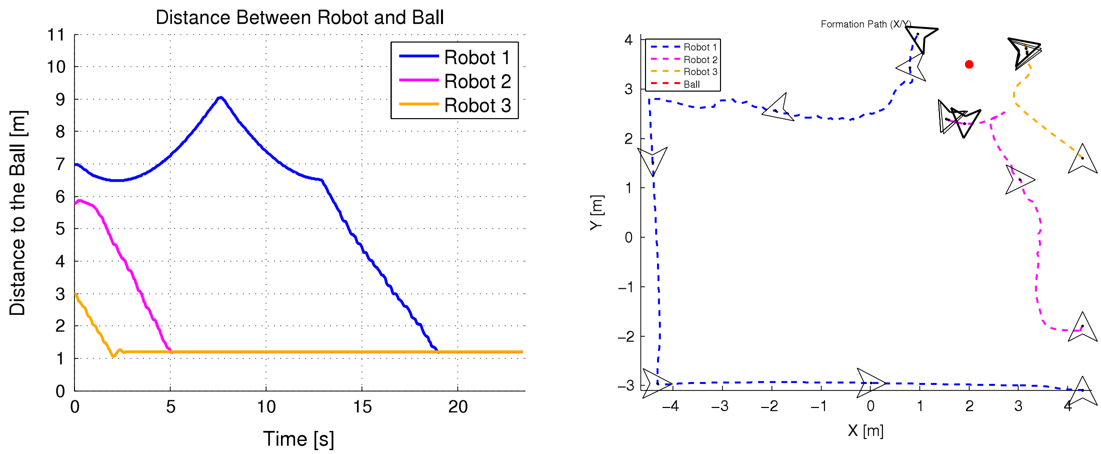


**Fig. 5.** Simulation 1: Distance robot-ball and plot *XY* of the robots path.

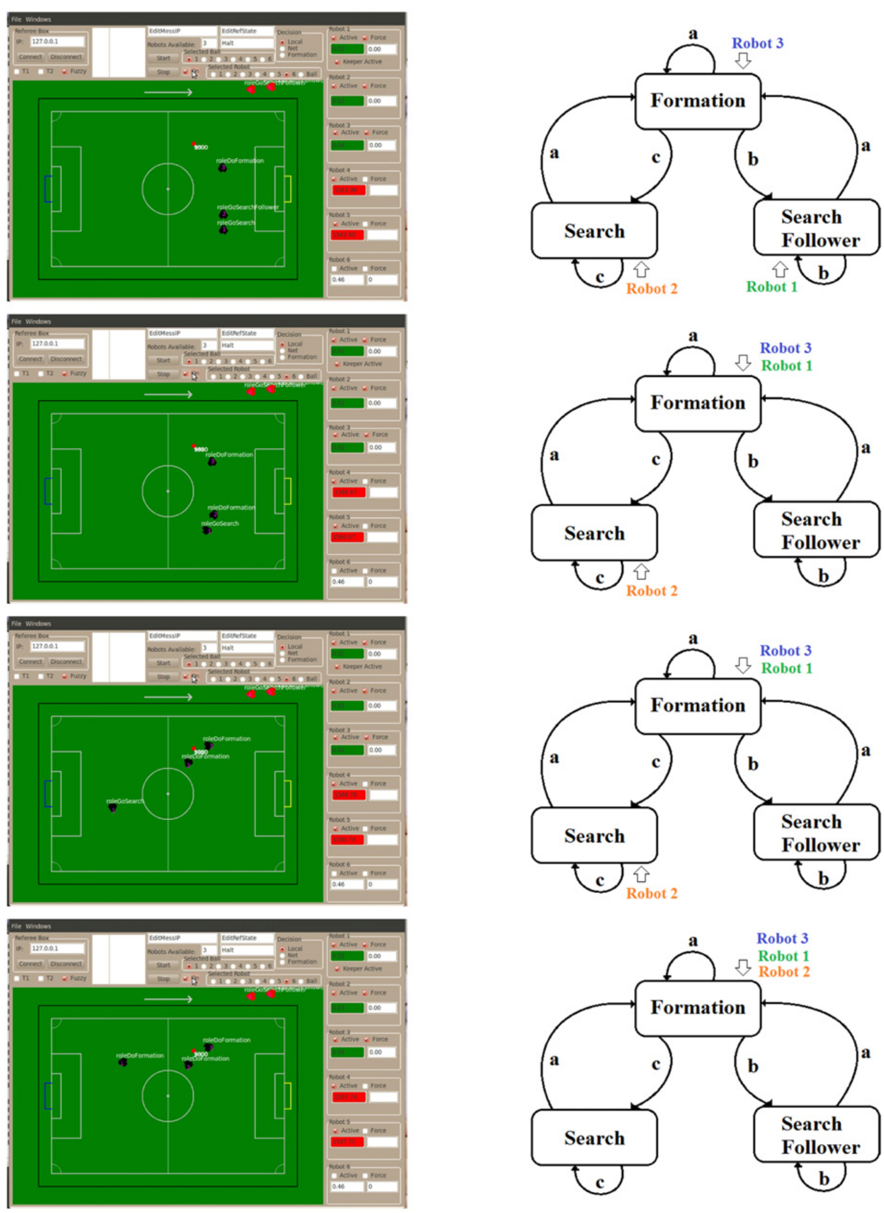


**Fig. 6.** Simulation 1.

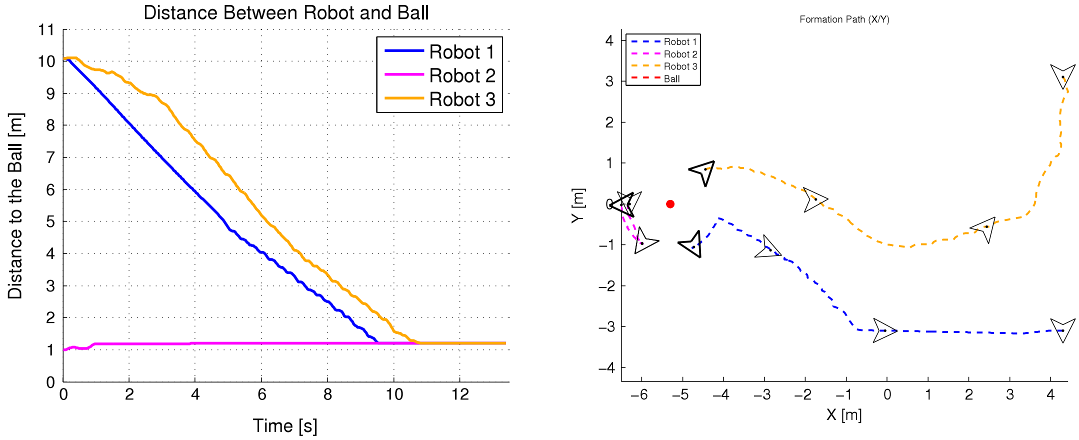


**Fig. 7.** Simulation 2: Distance robot-ball and plot *XY* of the robots path.

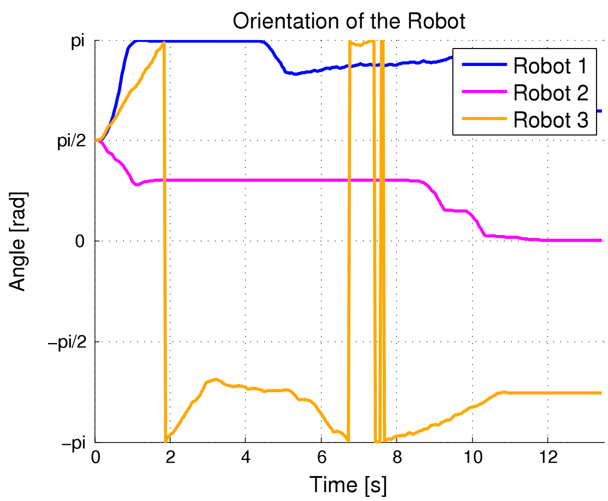


**Fig. 8.** Simulation 2: Orientation of the robots.

it can be seen that the robot converged successfully towards the target.

The *XY* plot from Fig. 7 demonstrates that robot 2 maintained its initial state *Formation* and moved towards the ball using the nonlinear model predictive controller (NMPC). Furthermore, robot 1 received *Search* as its initial state with an $A^*$ path planer and a reactive controller and it maintained this state until it reached a position where its confidence was high enough to change its state to *Formation* (coordinates $(1.1, -3)$). It is important to note that the change in state is also a change in the controller and no instability could be seen in the robot's path. However, the "irregularity" seen in the path of robot 1 is due to the change in its orientation around 5 s after the start of the experiment, as it can be seen in Fig. 8. The rotation is decided and governed by the low-level control system.

Finally, robot 3 receives *Search Follower* as its initial state and it starts moving towards the robot leader using a NMPC with a cost function to follow the leader. As it can be seen in Fig. 7, when robot 3 reached the point $(0, -1)$, it places itself with enough distance to the target in order to change its state to *Formation*. Here, the controller is the same but the cost functions are different and once again, no instability can be noticed.

### 5.1.3. Simulation 3

The last simulation case started with robots 1, 2 and 3 on one side of the field placed at coordinates $(4.3, -2)$, $(4.3, 0)$ and $(4.3, 2)$, respectively. Robot 1 is the leader robot here and the ball was positioned at coordinates $(-3, 1)$. Furthermore, all robots have an initial $\theta = 270°$ in world frame. Note in Fig. 9 that there is no disturbance in the robot's trajectory and there is no presence of instability in the state change.

By analyzing Fig. 9 this experiment shows that the robots started performing the search for the target by moving around the field. When robots 1 and 3 reached the coordinates $(-0.4, -3.1)$ and $(-0.1, -1.4)$ respectively, they began to see the target, which

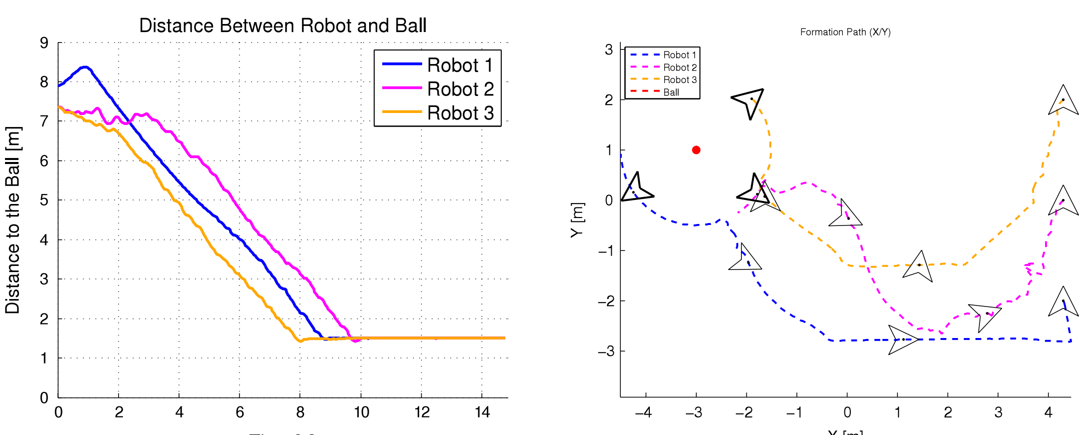


**Fig. 9.** Simulation 3: Distance robot-ball and plot *XY* of the robots path.

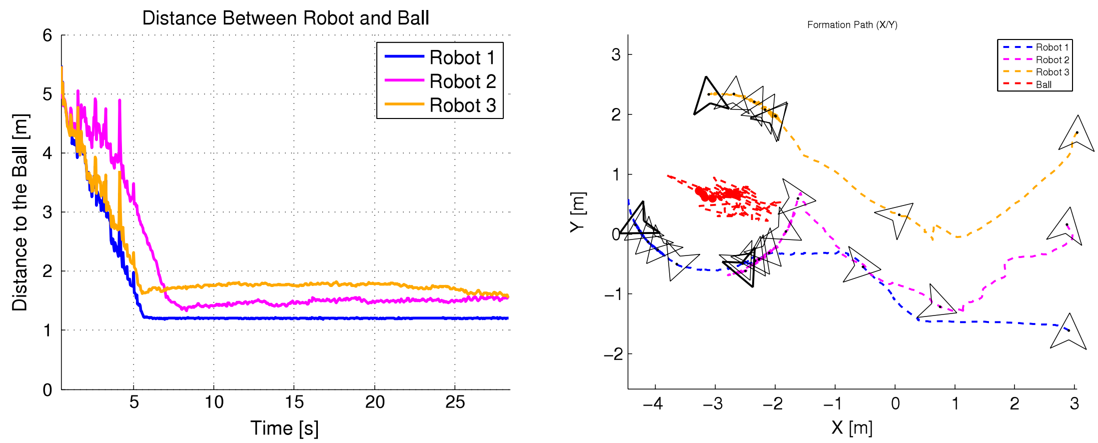


**Fig. 10.** Real experiment: Distance robot-ball and plot *XY* of the robots path.

in turn changed the state of each robot. Therefore, these two robots started the process of pursuing the target. At this instant robot 2 could not see the target yet. Meanwhile, robot 2 was a little bit behind and continued looking for the target and following robot 1 until the group gained more confidence in the observation while in turn robot 2 saw the target as well.

### *5.2. Results of the experiments with real robots*

This case was similar to the last simulation case where robot 1 was the leader and the search should be performed by departing from an target absent state. However, as the real field of experiments is small and the fact that the search routine makes the robot leader perform a square path in the field independent from its size the initial coordinates for robots 1, 2 and 3 were respectively $(3, -2)$, $(3, 0)$ and $(3, 2)$.

The small difference between this experiment and the last simulation is that the robots depart with a greater distance in the $X$ axis which in turn does not represent any significant difference for the comparison between simulation and real experiment. Moreover, Fig. 10 shows the distance from the robot to the ball during both states *Search/Search Follow* and *Formation*.

It is important to note that before changing to state *Formation*, the quality of the target's perception is affected by various vision noises due to the large distance between the robot and its target and the camera precision. This fact is demonstrated by the *XY* plot and by the distance graph from Fig. 10, and it makes the importance of this contribution more notable. In this experiment, the robots started performing the search for the target by moving around the field. When the robots reached a certain point in the field (coordinates $(0.2, -1.5)$ for robot 1 and $(0.5, 0.2)$ for robot 3 which were the firsts to see the target as seen in Fig. 10), the robots that reached this specific distance changed their state and started to pursue the target. Meanwhile, robot 2 was a little bit behind and continued looking for the target and following robot 1 until it saw the target as well. By analyzing the distance graph in Fig. 10, the main difference between the simulation and the real robot experiment are noticeable. Here, the variation in the perception of the target can be seen while the robots are searching and coincidentally approaching the target. It is the variation in the quality of the perception that makes the fuzzy system discard the ball positions. The point where the target is reasonably visible is reached in approximately four seconds when it can also be noticed that the perception quality has improved. The system works rapidly and smoothly, preventing the robots from colliding, which is possible at high velocity. However this is not the case when vision perception is not of a high quality. The *XY* plot of the movement of the robots is shown in Fig. 10. In this figure, the positions of the ball do not signify that the target moved, they only mean that the position of the target measured by the robots was not precise until the robot approached the target. This lack of precision in measurements makes the ball perception move in this plot. The behavior of the robots in this situation can be seen in Fig. 11.

## 6. Conclusion

This paper presented a new methodology, the Takagi–Sugeno type fuzzy automaton (TS-TFA), applied to the formation control of multi-robot systems to solve the target searching problem where the formation is put to search for a target and they converge to it when it is found. This problem lies in the fact that the target is initially absent and the formation of a multi-robot system must search for this target in the environment. During the target search, false targets may appear. Therefore, in order to avoid the formation following a false target, this paper presented the use of a recently presented technique, the TS-TFA. This technique selects one of the three roles applied to formation regarding the target searching problem. The fuzzy automaton not only changed the robots' roles (automaton states) but it also changed the robots' cost function or controller (depending on the state change), while searching for a target.

The TS-TFA was modified slightly in order to adapt to the formation control theory with the leader-following approach. From the simulations and the results with real robots it can be noted that the formation is broken to maximize the perception range based on each robot's observation of a possible target. It can also be noted that the change in the controller and the change in the cost function do not make the system unstable in the representative cases presented. Finally, this application can be generalized and the formation can consider all sorts of environment applications, including an outdoor environment where in many circumstances, a false target can be found.

## Appendix. Supplementary data

Supplementary material related to this article can be found online at http://dx.doi.org/10.1016/j.robot.2012.10.011.

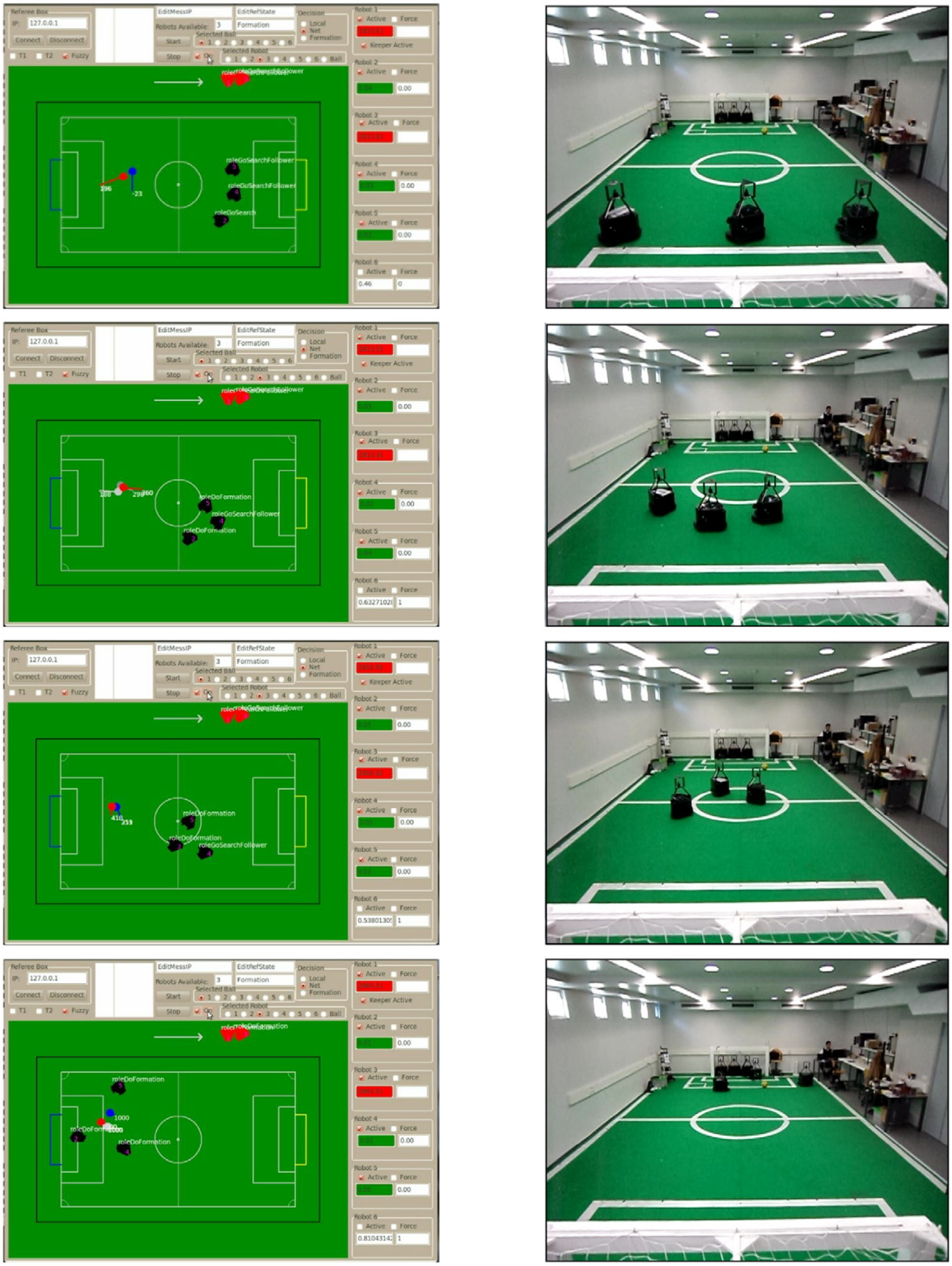

**Fig. 11.** Real experiment: Coach view.

## References


[1] S. Shaker, J.J. Saade, D. Asmar, Fuzzy inference-based person-following robot, International Journal of Systems Applications, Engineering and Development 2 (1) (2008) 29–34.

[2] H. Hagras, V. Callaghan, M. Colley, Learning and adaptation of an intelligent mobile robot navigator operating in unstructured environment based on a novel online fuzzy genetic system, Fuzzy Sets and Systems 141 (1) (2004) 107–160.

[3] J. Virant, N. Zimic, Fuzzy automata with fuzzy relief, IEEE Transactions on Fuzzy Systems 3 (1) (1995) 69–74.

[4] A. Bonarini, G. Invernizzi, T.H. Labella, M. Matteucci, An architecture to coordinate fuzzy behaviours to control an autonomous robot, Fuzzy Sets and Systems 134 (1) (2003) 101–115.

[5] C. McMillen, Maneula Veloso, Distributed, play-based coordination for robot teams in dynamic environments, in: Gerhard Lakemeyer, Elizabeth Sklar, Domenico Sorrenti, Tomoichi Takahashi (Eds.), RoboCup 2006, Robot Soccer World Cup X, in: LNCS, vol. 4434, Springer, Berlin, 2007.

[6] Nuno Lau, Luis Paulo Reis, FC Portugal—high-level coordination methodologies in soccer robotics, in: Pedro Lima (Ed.), Robotic Soccer, Itech Education and Publishing, Vienna, Austria, 2007.
[7] N. Lau, L.S. Lopes, N. Filipe, G.A. Corrente, Roles, positionings and set plays to coordinate a RoboCup MSL team, in: Proceedings of EPIA, Aveiro, Portugal, 2009, pp. 323–337.
[8] Vadim Kyrylov, Serguei Razykov, Pareto-optimal offensive positioning in simulated soccer, in: U. Visserm, F. Ribeiro, T. Ohashi, F. Dellaert (Eds.), RoboCup 2007: Robot Soccer World Cup XI, in: LNAI, vol. 5001, Springer, Berlin, 2008.
[9] J. Grantner, G. Fodor, Takagi–Sugeno type fuzzy automaton model, in: IEEE International Conference on Fuzzy Systems, Barcelona, Spain, 2010, pp. 1–6.
[10] T.P. Nascimento, M.A. Pinto, H.M. Sobreira, F. Guedes, A. Castro, P. Malheiros, A. Pinto, H.P. Alves, M. Ferreira, P. Costa, P.G. Costa, A. Souza, L. Almeida, L.P. Reis, A.P. Moreira, 5 dpo robot soccer team description paper, Available from: http://paginas.fe.up.pt/~robosoc/en/doku.php, 2011. URL: http://paginas.fe.up.pt/~robosoc/en/doku.php.
[11] R. 2011, Robocup 2011, Istanbul, Turkey. Available from: http://www.robocup2011.org/en/, 2011. URL: http://www.robocup2011.org/en/.
[12] H.T. Dashti, N. Aghaeepour, S. Asadi, M. Bastani, Z. Delafkar, F.M. Disfani, S.M. Ghaderi, S. Kamali, S. Pashami, A.F. Siahpirani, Dynamic Positioning based on Voronoi Cells (DPVC), in: Itsuki Noda, Adam Jaco, Ansgar Bredenfeld, Yasutake Takahashi (Eds.), RoboCup 2005, Robot Soccer World Cup IX, in: LNAI, vol. 4020, Springer, Berlin, 2006.
[13] Ryota Nakanishi, Kazuhito Murakami, Tadashi Naruse, Dynamic positioning method based on dominant region diagram to realize successful cooperative play, in: U. Visserm, F. Ribeiro, T. Ohashi, F. Dellaert (Eds.), RoboCup 2007: Robot Soccer World Cup XI, in: LNAI, vol. 5001, Springer, Berlin, 2008.
[14] H. Akiyama, I. Noda, Multi-agent positioning mechanism in the dynamic environment, in: U. Visserm, F. Ribeiro, T. Ohashi, F. Dellaert (Eds.), RoboCup 2007: Robot Soccer World Cup XI, in: LNAI, vol. 5001, Springer, Berlin, 2008.
[15] P. Stone, M. Veloso, Task decomposition, dynamic role assignment, and low-bandwidth communication for real-time strategic teamwork, Artificial Intelligence 110 (2) (1999) 241–273.
[16] Luis Paulo Reis, Nuno Lau, Eugénio C. Oliveira, Situation Based Strategic Positioning for Coordinating a Team of Homogeneous Agents, in: LNAI, vol. 2103, Springer, Berlin, 2001.
[17] K. Miao, M. Li, Role assignment strategy for heterogeneous robots based on fuzzy evaluation and fuzzy inference, in: 9th International Conference on Control, Automation, Robotics and Vision, Singapore, 2006, pp. 1–5.
[18] S. Hamidreza Mohades Kasaei, S. Mohammadreza Mohades Kasaei, S. Alireza Mohades Kasaei, Mohsen Taheri, Design of an action selection mechanism for cooperative soccer robots based on fuzzy decision making algorithm, BRAIN: Broad Research in Artificial Intelligence and Neuroscience 1 (3) (2010) 5–18.
[19] T.P. Nascimento, A.G.S. Conceição, F.A. Fontes, A.P.G.M. Moreira, Leader following formation control for omnidirectional mobile robots: the target chasing problem, in: Proceedings of the 8th International Conference on Informatics in Control, Automation and Robotics, ICINCO, Noordwijkerhout, Netherlands, 2011, pp. 102–111.
[20] T.P. Nascimento, Controle de Trajetória de Robôs Móveis Omni-direcionais: Uma Abordagem Multivariável, Master, Federal University of Bahia, 2009.
[21] T.P. Nascimento, A.G.S. Conceição, A.P.G.M. Moreira, A modified A* application to a highly dynamic unstructured environment, in: Proceedings of the 11th International Conference on Mobile Robots and Competitions, Lisbon, Portugal, 2011, pp. 9–14.
[22] R. Smith, P. Cheeseman, On the representation and estimation of spatial uncertainty, The International Journal of Robotics Research 5 (4) (1986) 56–68.
[23] A.J.M. Sousa, Arquitecturas de Sistemas Robóticos e Localização em Tempo Real Através de Visão, Ph.D., Porto University, 2004.
[24] A. Sousa, P. Costa, A. Moreira, Vision based real-time localization of multiple mobile robots, in: 3rd Int. Conf. on Field and Service Robotics, Helsinki, Finland, 2001, pp. 103–106.
[25] P.J. Costa, Simtwo. Available from: http://paginas.fe.up.pt/~paco/wiki/index.php?n=Main.SimTwo, 2010. URL: http://paginas.fe.up.pt/~paco/wiki/index.php?n=Main.SimTwo.

**Tiago P. Nascimento** is a Ph.D. student from the Doctoral Program in Electrical and Computer Engineering from Oporto University (Portugal). He received his B.S. in mecathronics engineering at the College of Technology and Science—FTC (Brazil) in 2007 and his M.S. in electrical engineering at the Federal University of Bahia (Brazil) in 2009. Also in 2009 he worked as an assistant professor for the Electrical Engineering Department of the Salvador University and as DTI-2 researcher from CNPq (Brazil). Presently, he does research in mobile robotics at INESC of Oporto and is a member of the IEEE and IEEE RAS since 2004.

**António Paulo Moreira** graduated with a degree in electrical engineering from the University of Oporto, in 1986. Then, he pursued graduate studies at the Oporto University, obtaining an M.Sc. degree in electrical engineering—systems in 1991, and a Ph.D. in electrical engineering in 1998. Presently, he is a lecturer in electrical engineering, developing his research within the Robotic and Intelligent Systems Department of INESC of Oporto. His main research interests are process control and robotics. He is a member of the "Associação Portuguesa de Controlo Automático" and "Sociedade Portuguesa de Robótica" and member of IEEE and IEEE RAS.

**André G. Scolari Conceição** received his B.S. and M.S. degrees in Electrical Engineering from Pontifical Catholic University of Rio Grande do Sul (Brazil) in 2001 and 2004, respectively, and his Ph.D. from Oporto University (Portugal) in Electrical and Computer Engineering, in 2007. In 2009, he joined the Department of Electrical Engineering, Federal University of Bahia, Salvador (Brazil), as an Assistant Professor. His main research interests are in mobile robotics, multi-robots systems, sensor fusion, state and parameter estimation, process control, and vehicle dynamics.

**Andrea Bonarini** received his Master in Electronics Engineering in 1984 and Ph.D. in Computer Engineering in 1989 from Politecnico di Milano. Master in Neuro-Linguistic Programming in 1993, from IIPNL. He is a full professor and member of the Ph.D. Board (Coordinator for the Computer Engineering Section since 2011) at the Politecnico di Milano, Department of Electronics and Information. Since 1984 he is a member of the Politecnico di Milano Artificial Intelligence and Robotics Project, and coordinator of the AIRLab since 1990. He has been nominated Fellow of the Alta Scuola Politecnica in 2012. He is among the founders of the Italian Association for Artificial Intelligence (AI*IA) and the Italian Regional Interest Group of the IEEE Neural Network Council, now Italian Chapter of the IEEE Computational Intelligence Society (Chair from 2008 to 2010). From 2003 to 2006, he has been coordinator of the Working Group on Robotics of the AI*IA. He participated, since 1997, in the Robocup initiative (member of the Executive Committee from 2002 to 2010). He is currently in charge of "Informatics", "Artificial Intelligence", and "Soft Computing" courses at the Politecnico di Milano.