v. 5, n. 5, 2025

Journal of
Engineering
Research

Acceptance date: 20/05/2025

MULTI-ROBOT EXPLORATION INSPIRED BY SWARM ROBOTICS USING DYNAMIC FUZZY COGNITIVE MAPS

M. Mendonça
Universidade Tecnológica Federal do Paraná, Campus Cornélio Procópio, PR, Brazil

R. H. C. Palácios
Universidade Tecnológica Federal do Paraná, Campus Cornélio Procópio, PR, Brazil

L. B. de Souza
Universidade Tecnológica Federal do Paraná, Campus Cornélio Procópio, PR, Brazil

M. A. M. Laia
Universidade Federal de São João del Rei, MG, Brazil

M. G. Palácios
Universidade Tecnológica Federal do Paraná, Campus Cornélio Procópio, PR, Brazil

V. B. Milani
Universidade Tecnológica Federal do Paraná, Campus Cornélio Procópio, PR, Brazil

Abstract: This work presents a multi-robot system (MRS) inspired by swarm robotics concepts for exploration and rescue. The exploration is done using a dynamic fuzzy cognitive map (DFCM) and fuzzy logic (FLC) to control the robots, and the results are compared for performance verification. The MRS must complete an exploration task in a semi--unknown flat environment. Thus, the MRS is not aware of the limits of the searching area, obstacles, and only knows the number of victims to be rescued as a stopping criterion. The proposed task has the purpose of simulating the rescue of victims in disasters. In the tests, the simulations used three environments to verify the global robustness in unpredictable situations, check the autonomy of the robots, and explore the areas of the performance space. The computational time between the two controller approaches is estimated using an architecture based on reactive subsumption. In general, results imply that the DFCM-based MRS approach completed the proposed tasks using less processing time and the robots had to travel less to explore a similar amount of area as seen in the FLC approach. Finally, the authors show an initial prototype, which can be used in real life experiments, and address future work.

Keywords: Autonomous Mobile Robots; Swarm Robotics; Semi-unknown Environments; Dynamic Fuzzy Cognitive Maps; Exploration

INTRODUCTION

The application of robots is increasing over recent years. Robots are now used in the educational and medical fields, in industries, agriculture, for firefighting and rescuing lost people, surveillance, in military operations (in land and underwater), for lunar and Mars exploration, among others [1,2]. Specifically, when a group of autonomous mobile robots is used, most of these tasks can be accomplished in a better way due to the parallelism yielded by the group, which can make tasks faster than using a more complex single robot [3–5].

In this way, the article addresses the concept of autonomous robotics investigation. It starts by focusing on the shortcomings of current hospital guide robots, which are marked by unclear functions, limited usefulness, and low serviceability. This study focuses on the users, aiming to identify their real needs during hospital visits and explore possible improvements in the design of guide robots. The research uses the Quality Function Deployment (QFD) and Analytic Hierarchy Process (AHP) to pinpoint essential design requirements. At the same time, the Function-Behaviour-Structure (FBS) model is applied to outline the appropriate structural components of the guide robot based on these needs. Several design innovations in appearance, operational areas, and interface are highlighted. Implementing these designs improves both the robots' functionality and user satisfaction. The QFD-FBS approach helps create a user-centered, innovative design process that enhances the hospital experience, streamlines consultations and introduces a fresh take on service robot design in specific environments. [6]

The research field of MRS with collective intelligence is called swarm robotics, which deals with large numbers of homogeneous or heterogeneous robots [7]. It differs fromtraditional systems due to its non-centralized and non-hierarchical control. In recent years,swarm robotics is emerging as an alternative approach for the coordination of an MRS,consisting of a large number of physically simple and homogeneous robots, without acentral coordination (controller), among other aspects.

Swarm robotics systems are fully-distributed and self-organized, and are often inspired by the behavior of collective animals, such as birds, ants, and bees to develop similar behaviors in the MRS [8]. However, the demonstration of physical robot swarms is both a hardware and a software challenge [2]. To make cooperative decisions, each robot must have an insight into the environment in which it is found and might have to perform a processing of tens or hundreds of megabytes per second [9]. Besides, local communication, robustness, flexibility, stigmergic behavior and decentralized control are recurrent challenges for swarm robotics [2].

The use of soft computing techniques is increased in MRS aiming at these features, resulting in more flexible robot architectures. These techniques deal with the infinity of states that robots can achieve in unknown environments, as seen in [10,11] in which multiple robots should explore cooperatively an unknown locations to find and deactivate landmines.

A recurrent scenario seen in MRS is to assign tasks to the team of robots to hazardous environments. In these situations, robots finish the tasks in a more efficient and cost-effective way, adaptively, and more robustly than humans [11]. A possible task in an MRS is called foraging, and it is the behavior of capturing targets [12]. The foraging problem of the proposed work consists of a stochastic search for a well-defined number of victims in unknown areas, resulting in semi-unknown environments, e.g. disaster sites. In these hazardous situations, the main challenge is the response time: the survival rates decrease significantly after 48h [13].

Several papers dealt with exploration and S&R objectives using swarm-based MRS. In [13] authors present a behavior-based fuzzy-controlled robot swarm to perform S&R operations in unknown environments, while [14] uses a robot leader to guide a swarm in S&R operations through a heuristic algorithm to rescue the victims at a disaster site. Authors in [15] present a survey of methods in swarm robotics S&R applications. Finally, [16] combines aspects of swarm intelligence with FLC for a collective search of an MRS. This work is part of a research initiated in 2016 in the autonomous mobile robotics field using fuzzy cognitive maps (FCMs). The first paper [17] presented one of the early applications of FCMs in mobile robots, and implemented two navigation systems using hybrid-dynamic fuzzy cognitive maps (HD-FCM) and hierarchical fuzzy logic controllers (FLC) in one robot. The next work [18] dealt with the construction of a prototype, and the robot' kinematic model validation through the overlapping of the simulated and real trajectories using an embedded HD-FCM.

In the third paper [19], the architecture was scaled to its use in swarm robotics, applying concepts of Ant Colony Optimization (ACO). This approach uses one robot released at a time in the environment, leaving pheromone during its ride. The last two works [5,20] use FLC and FCM to a group of homogeneous robots working simultaneously. The results of both approaches were compared with 1 and 4 robots operating, in two simulation scenarios.

The present paper improves the last one [5] by adding a third environment and eight robots scenarios to the simulations. In this sense, the robots can be characterized as autonomous if they complete their tasks successfully in three environments [21]. In order to test the global robustness of the MRS against uncertainties, the authors added non-ideal conditions in all three environments: robots are placed close to each other (dynamic obstacle avoidance condition) and the eight robots scenarios deal with a higher ratio of robots per victim in order to verify the sensing and rescue capabilities of the swarm.

Regarding the organization of this paper, the section 1 presents an overview of the proposed architecture, fuzzy logic and cognitive maps, in addition to some related works. In addition, the concepts of multi-robot systems (MRS) and swarm robotics are presented. Section 2 shows the development of the proposed controllers (FLC and DFCM) with the MRS behaviors and architecture, and explains the simulation configuration and procedures. In the section 3, all experiments with their results are reported. Then, in Section 5 the conclusions are presented together with future work.

ASSUMPTIONS AND BACKGROUND

In this section, authors discuss the system's architecture, and present an overview of multi-robot systems and swarm robotics considering the focus of this work. In addition, some fundamentals of fuzzy logic, FCMs, and related papers using these techniques are gathered and presented.

ROBOT'S ARCHITECTURE

An architecture can be understood as the set of instructions that define the behavior/functioning of a system according to its sensory and physical characteristics. In the context of autonomous mobile robots, [22] developed the subsumption architecture, widely used for its simplicity of implementation and understanding.

The subsumption architecture was probably one of the first attempts to compose complex behaviors through layers of simpler behaviors, hierarchical or not [23]. This characteristic meets the concept of organization of behaviors through symbolic representations of environments, as seen in the classic concepts of artificial intelligence [22].

A possible interpretation for this architecture, initially classified as reactive, can be seen in Fig. 1 [2]. In short, the perception system activates behaviors through a set of conditions (external stimuli). Thus, each behavior or sub-behavior, process the information received and directs the corresponding actions, which are transcribed as commands to the system.

In the context of this work, factors such as the mapping (navigation) system, the recurring need for decision-making and the need to fulfill varied sub-objectives suggest the use of a subsumption-based architecture. In it, the adaptation of agents (robots) is observed by the addition of new sub-behaviors or modules.

There is a hierarchy in the behaviors of the MRS in the present work. Each sub-behavior of the system respects a priority level, starting from low-level (reactive) to high-level layers, represented by intelligent controllers (FLC and DFCM) and the deliberative characteristics of robots. The modeling of these sub-behaviors, as well as the use of the subsumption architecture, was inspired by [24] incompatibility principle. It states that the greater the complexity of a system, the more improbable and imprecise the prediction of its reactions to environments becomes. This fact is seen in the work of [25], which models the functioning of MRS based on human emotions and behaviors.

Finally, the sub-behaviors used in this work are alternated as a result of the sensors' data operating sequentially, while the MRS works in parallel. In other words, at any given moment, all robots can have a different sub-behavior activated. However, in each robot, only one sub-behavior operates at a time.

FUZZY LOGIC AND COGNITIVE MAPS

Humans' understanding of most physical processes is based on imprecise aspects. When compared to the precision usually used by computers, this inaccuracy can be useful for modeling systems. Thus, the ability to incorporate this form of reasoning into previously unmanageable and complex problems is one of the main characteristics of fuzzy. In this context, there is a good impact of using

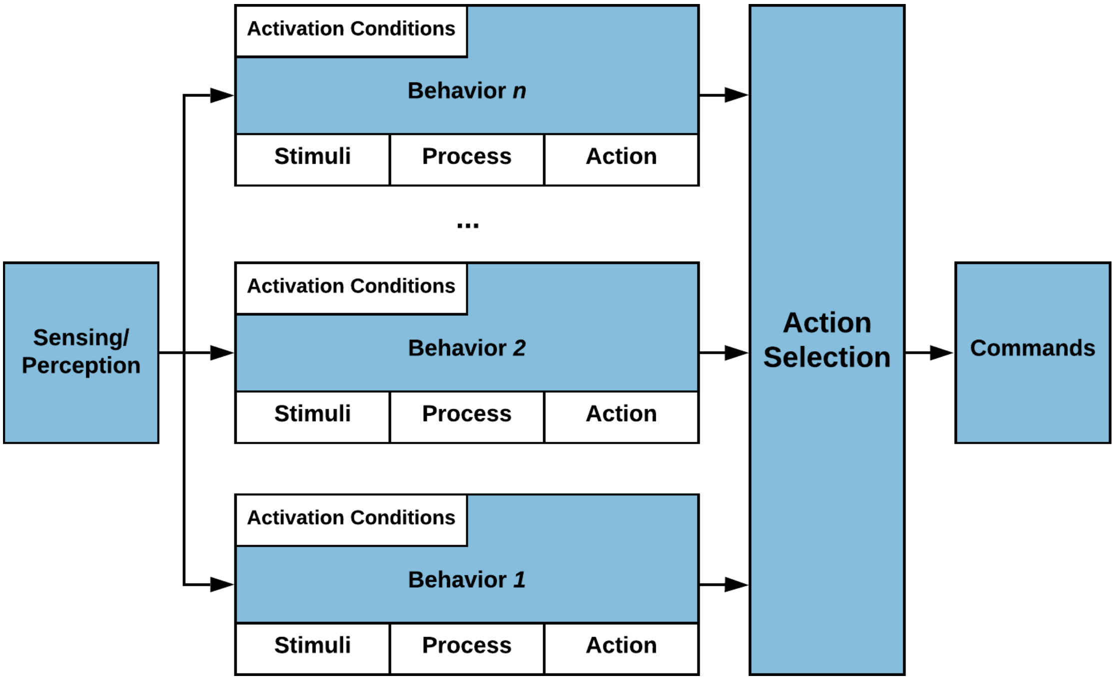

Figure 1. Behavior distribution

fuzzy concepts in problems that do not require such a scale of precision, like maneuvering and navigating a vehicle [26].

Zadeh [27] created the theory of fuzzy sets aiming a method capable of handling systems with imprecise information. Hence, a fuzzy variable/set/system has a pertinence level, given between [0 1], that encodes experts' knowledge about the system.

Fuzzy cognitive maps (FCMs) are known to be convenient, simple, and yet a powerful enough solution for controlling dynamic systems [28]. FCMs have fuzzy-interconnected concepts that symbolize linguistic concepts. These connections model the relationships between concepts through a level or intensity of causality [29]. Combined with the graphic representation of a FCM – a directed graph – these features make FCMs simple and intuitive to understand in terms of its formal model and processing [30]. However, it has a drawback of not modeling time [31].

In addition, as the use of FCMs increases, several limitations of its classic approach have emerged, described as follows. Lack of knowledge of the system, when the nature of the concepts (inputs, states, products) is not taken into account [32]. Dependence on experts: FCMs are highly dependent on expert intervention for the design of fuzzy cognitive maps [33]. Inability to self-learn: the design of adaptation approaches is more difficult due to their complex structure and variability [34]. Definition of causalities (relationships and weights): how causalities can be defined with a high degree of credibility/reliability [35]. Calculation equation: how can a unified value be calculated, to describe the total change caused by one of the concepts by all the others [36]. Ignorance of the time factor: how can a different time delay for each causal relationship be implemented [37]. Use of the sigmoid function and interpretation of results: the use of the curve to adjust the results to the desired interval can distort the final interpretation of the results [38].

Several papers can be found in literature that are related to the present work, applying FCMs and its extensions on navigation purposes. The first study developed an autonomous navigation system using event-driven fuzzy cognitive maps (ED-FCM). In it, the arcs of the ED-FCM are updated based on the occurrence of events such as dynamic obstacle detection. The model is capable of representing the dynamic behavior of the robot in the presence of changes in the environment, and constitutes a flexible and robust navigation system, capable of processing inaccuracies and uncertainties in the environment [39].

The author from [34,40] used FCMs for simulated mobile robot navigation. The first one compares several learning methods for the proposed FCM, while the second paper is focused in gathering concepts of internet of things (IoT) and intelligent space (IS) to navigate a ubiquitous robot (Ubibot) using an adaptive FCM. The contribution from [41] developed an expert FCM control a Pioneer platform through different scenarios. The causal inference mechanism FCM was used to coordinate the system's motion using expert-defined inputs, outputs, and its interactions. The results indicate that the use of FCMs is advantageous over conventional rule-based techniques, especially preventing exponential growth in the rules as the system gains complexity.

The last one [42] compares a neuro-adapted FCM with other techniques such as a genetic algorithm-based PID. The FCM was tested both in simulation and experimentally in circular and square trajectories, being advantageous over the other techniques in position error and number of maneuvers by the mobile robot.

MULTI-ROBOT SYSTEMS AND SWARM ROBOTICS

Multi-robot Systems (MRS) has been widely used in complex and hazardous scenarios due to its adaptation capabilities, flexibility and high reliability when comparison to single robot systems [43]. These aspects defined the choice of the MRS in the present paper. In addition, authors aimed some aspects of swarm robotics in the behavior of the robot group.

The inspiration from nature used to control large groups of relatively simple MRS is called swarm robotics [13]. In this field, robots are often unaware of the actions of other distant robots, i.e. the communication is made only between nearby robots, relying on stigmergy [2]. Another assumption is that individual robots cannot solve complex tasks on their own, only in a team-level. Finally, the increase in the number of robots in a group can reduce significantly the task completion time [44]. Swarm robots have collective, cooperative, collaborative, and coordinative behaviors. [45,46].

The movement coordination in an MRS can only be relative to other robots, the environment, external agents, and combinations between them [2]. In this research, the robots make movements like the other robots, to avoid static and dynamic obstacles and find the target.

For an MRS, the first definition must be how the robots will be developed, and how they will interact with each other and/or the environment. In this work, some aspects of swarm robotics inspired the behavior of group of robots, i.e. an MRS must present a robust, flexible, and scalable operating level [8,47]. Robustness is within the completion of the objectives despite robots' malfunction or environmental disturbances. Flexibility deals with the capability of generate solution for different tasks using different coordination strategies in response to dynamic environments. In addition, the swarm is scalable to be not disturbed due to changes in the robots' group sizes [3].

DEVELOPMENT

The methodology of this contribution is divided into three main parts. In the first one, authors define which architecture will be used in the MRS and its desired behaviors. The second one is the controller modelling using a dynamic fuzzy cognitive map (DFCM). Finally, the last one is responsible by the assumptions to model the simulations, such as the type of detection of the robots' sensors and the creation of test environments.

MRS ARCHITECTURE

The robots used in this proposal are reactive to external stimuli, as observed in the works of [5] and [22]. In the robot modeling proposed in this research, the linear and angular velocity are defined by v and w; b is the axis length (14.4cm), R_R and R_L are the radius of rotation of right and left wheels (2.5cm). It is noteworthy that this kinematic model can also be used in four wheeled robots without steering front wheels. The kinematic model given by (1) and (2) was used to in this work: two powered front wheels and a dummy back wheel to stabilize the turns.

$$v_{R,L} \cdot dt = w_{R,L} \cdot [\pm \tfrac{b}{2}] \cdot dt \quad (1)$$

$$\begin{bmatrix} v \\ w \end{bmatrix} = \begin{bmatrix} \frac{R_R}{2} & \frac{R_L}{2} \\ \frac{R_R}{b} & \frac{R_L}{b} \end{bmatrix} \cdot \begin{bmatrix} v_R \\ w_L \end{bmatrix} \quad (2)$$

Furthermore, the pulses of the motors coupled to the wheels (right and left) are generated by the controller outputs (w_L and w_R, 0 to 100%). In case the pulses are positive, it will activate the robot's forward command, on the other hand, if $w_L > w_R$ the robot will make a maneuver to the right, otherwise it will be to the left. On the other hand, if it is necessary to rotate the robot, a positive pulse on one wheel and a negative pulse on the other must be executed [5]. The controller receives input from three ultrasonic sensors arranged in front (*FS*), left side (*LS*), and right side (*RS*), with a range of 12cm, beam angles of 45 and white noise to verify the robustness of both the FLC and DFCM.

To model the sub-behaviors, the first step is to determine the scenarios to classify them into sub-domains and describe its expected behaviors, as shown in Table 1.

Sub-domain	Description
Individual goals	Environment exploration, collision avoidance and victims rescue Global goal
Global goals	The robots must rescue all the victims
Individual sub-behaviors	Depicted in the finite-state machine (see Fig. 3)
Global behavior	The robots must collect victims and then continue with the exploration

Table 1. Sub-domains

In this research, the authors use a solution close to previous works, as observed in [20] and [5]. Thus, this contribution enhances its comparison by adding a third environment, using a DFCM this time. In addition, the authors tested possible stress scenarios (8 robots) to verify the robustness of the MRS.

Therefore, more assertively, the MRS has three functions: 1) to detect obstacles; 2) locate and rescue all victims/targets; and 3) map environments. However, to complete the tasks, four sub-behaviors were needed [3], as follows:

- Sub-behavior 1, related to free motion, is executed if the sensors find something and generate a unit pulse for the motors at half speed and increase after each iteration until it reaches maximum speed or is triggered by another event.
- The second sub-behavior, responsible for obstacle avoidance, is activated if the robot detects an obstacle in the range between the safety distance (5cm) and the maximum detection of the sensors (12cm). Thus, the controller is responsible for assigning smoother curves to the robots.

- The third sub-behavior deals with impending collision detection: it is activated in case the ultrasonic sensors have detected an obstacle smaller than the safe distance determined by 5*cm*. Thus, the robot performs a fast turn. In addition, so that the robot does not loop, this algorithm changes the direction of the curve every time the robot enters sub-behavior 1.

- Sub-behavior 4 addresses the rescue of victims. When the robot finds it, its color changes in the simulation. In this sub-behavior, DFCM slows the robot down when approaching the victim or if there is an obstacle under the safety zone. This feature can disqualify robots as purely reactive due to their path planning type (DFCM). Furthermore, according to the sensor data, the DFCM controller modifies its weight matrix through each sub-behavior, as described in Section 2C.

In addition, robots present a memory algorithm that maps the environment, and if a robot finds a victim, but cannot rescue her after a certain amount of time, nearby robots receive their location to do the rescue. This is because instantly, after locating a victim, their rescue becomes the robot's primary objective. Thus, if an obstacle is encountered, the robot will forget its target and let another one rescue it with better chances [20]. Other mapping approaches that are suitable for the application of this article are seen in [48,49]. An overview of the system's architecture can be seen in Fig. 2, and can be used to assist in the building of DFCM models for autonomous navigation systems. The signals received through the sensors enter a decision-making process that modifies the sub-behaviors inside the state machine according to the occurrence of the events from Table 2.

Event Description	
A	Obstacle detected
B	The three ultrasonic sensors detected an obstacle within the robot's safety zone
C	The robot found a victim/target
D	Victim/target found on the way to being rescued
E	Activation of the robots' memory algorithm

Table 2. Events Description

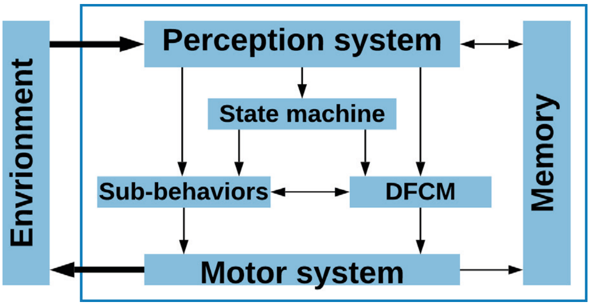

Figure 2. Overview of the system's architecture

Consequently, every sub-behavior triggers a different weight matrix of the DFCM, as shown Table III (Section 2.1). In the next step, the DFCM's outputs are sent to the motor system. The pose data and the locations visited are stored in the robots' memory for mapping. For future works, the second stage of the DFCM block will be devoted to the weights' tuning through a reinforcement-learning algorithm or the Hebbian rule.

As seen in Fig. 3, the subbehaviours according to the finite state machine operate in parallel and individually. Furthermore, the real data coming from the sensors inhibits or activates the routines producing the desired outputs [5]. The events in the transition of states are presented in Table 2. The inscriptions on the arrows in Fig. 3 indicate the event (letter) that triggered the behavior listed in the alphabet from Table 2. Thus, the first number indicates the sub-behavior before the occurrence of the event, while the second indicates the current sub-behavior generated by the occurrence.

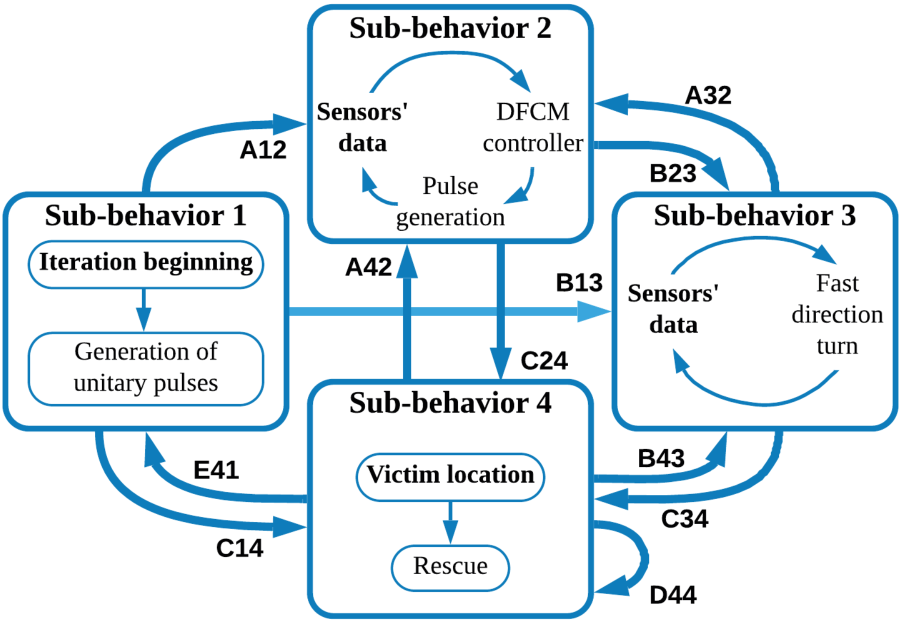

Figure 3. Finite state machine for robot operation

DYNAMIC FUZZY COGNITIVE MAP CONTROLLER

The fuzzy control system developed for this work is a Mamdani-type described as follows. It has three inputs (ultrasonic sensors), and two outputs (pulses sent to DC motors). The universes of discourse of the FLC variables are divided into five pertinence functions: two trapezoidal at the edges and three triangular at the center.

The input range is [0 12], and the output range is between [0 1]. Authors used an rule (tuned by empirical trial) weighted FLC to compare it to a FCM in two environments, as seen in the previous work [5].

However, to define these 125 rules is not a trivial task. The use of FLCs in complex systems like the present MRS could cause an "explosion" in the number of rules as higher the system's complexity goes, e.g. by adding new sub-behaviors. This scenario can be avoided by using FCM-like controllers, such as the proposed DFCM.

In this work, the DFCM updates its weight matrix according to the sensor data and the real sub-behavior of the system. This is the main difference between DFCM and FCM, which has only one weight matrix [39].

In Fig. 4 the DFCM is shown. In this case, the diamonds are the decision processes. In particular, if an obstacle is inside the safe zone, all W_{ij} weights are modified to allow robots to slow down and make sharper turns [5]. The authors adjusted these causality levels empirically according to the simulation results.

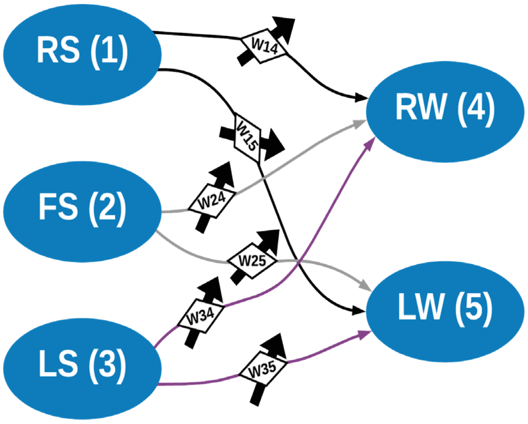

Figure 4. Proposed DFCM

The DFCM weight matrix is updated according to the actions/events presented in Section 2.1. Each action/event that occurs is then defined as the robot's subbehavior. The relationships between weights and sub-behaviours are presented in Table 3, in which sub--behaviours 1 and 4 have the same weights.

Weight	Sub-behavior			
	1	2	3	4
W14	-0.40	0.60	0.80	-0.40
W15	0.50	-0.20	-0.60	0.50
W24	0.30	-0.30	-0.60	0.30
W25	0.30	-0.30	-0.60	0.30
W34	0.50	-0.20	0.60	0.50
W35	-0.40	0.60	-0.70	-0.40

Table 3. DFCM's Weights per Sub-Behavior

SIMULATION PROCEDURES

In the last step, simulations with three environments were performed in Matlab® software. The PC used has the following specifications: eight-core CPU, 16GB RAM, and a SATA 3 SSD ($R : 500MB/s$, $W : 350MB/s$).

The comparison between FLC and DFCM used the same approach from previous works [5,20] with new experiments made for all three environments. FLC and DFCM were compared in the same environmental conditions: simulations started with one robot, followed by four, and then expanded to eight. The simulations assume that all ultrasonic sensor have a maximum perception distance of 12cm, and a safety distance of 5cm (which triggers sub-behavior 3).

Considering the simulated semi-unknown environments, the role of robots is to find victims, map the environment and avoid collisions.

In Matlab®, the sensors were simulated using the cartesian distance between the robots and obstacles/victims. The rescue perimeter was chosen based on the distance between the robots' center of mass and the victims, considering a maximum error of 3cm.The inputs of the controllers are inspired by the actual data from three HC-SR04 ultrasonic sensors located in front of the robots. Its range is 12cm and has beam angles of 45.

In order to approximate the model to possible real conditions, a white noise was added to the simulations. In addition, it is important to note that, at simulation level, a victim is automatically detected when it is within the range of any ultrasonic sensor, which would not be possible in a real application without the use of another sensor, such as a camera.

The victims start red in the simulations, and turn green when found by any robot. Similarly, static obstacles, initially blue, turn black when detected by the MRS. These obstacles delimit all environments' rescue area of 100cm · 100cm. However, these dimensions are suitable to changes in real scenarios of victims' search. In all environments, the victims' locations are the same of the previous work [5], and unknown to robots: (10 40), (10 90), (40 20), (50 80), (90 20), and (90 80).

Robot	X (cm)	Y (cm)	Angle (o)	Color
1	15	5	90	Blue
2	30	5	90	Green
3	45	5	90	Red
4	65	5	90	Cyan

Table 4. Poses: Four-Robots Scenario

Robot	X (cm)	Y (cm)	Angle (o)	Color
1	15	5	90	Blue
2	25	5	90	Green
3	35	5	90	Red
4	45	5	90	Cyan
5	55	5	90	Magenta
6	65	5	90	Yellow
7	75	5	90	Black
8	85	5	90	Blue

Table 5. Poses: Eight-Robots Scenario

The parameters used to compare the computational effort of both controllers are the number of iterations needed to rescue all victims, and Matlab® processing time (through build-in functions *tic* and *toc*). In addition, other aspects were compared such as the smoothness of the pulses sent to the wheels (the angular velocities from right and left wheels, respectively w_R and w_L). Finally, effectiveness and efficiency of both controllers were compared by means of the explored area and traveled distance of each robot and by the entire MRS.

The robots' initial poses consist of its positions (x, y) and angle. These parameters (and the colors of the robots) are shown in Tables 4 and 5 according to the number of robots, and are distributed along the *x-axis*. In the scenario of the solo robot, the pose is (40, 10, 0).

RESULTS

In this section, the results of the FLC and DFCM approaches are presented in the initial condition of plane environments.

SOLO ROBOT

The first step was to deploy the FLC and DFCM-controlled robots in all environments to serve as a basis for comparison with the simulations considering 4 and 8 robots. In this scenario, it is expected that the robots may not complete the task (rescue six victims), or complete it taking more time than desired, considering that a rescue task must be completed at the shortest time possible.

The robot's trail can be seen in Fig. 5 for each environment. In the first one, the robot faced some expected difficulties in the upper-left quadrant, a confined space. This fact led the robot not to explore the central-left area of that environment, since the simulation stopped when the robot found the last victim.

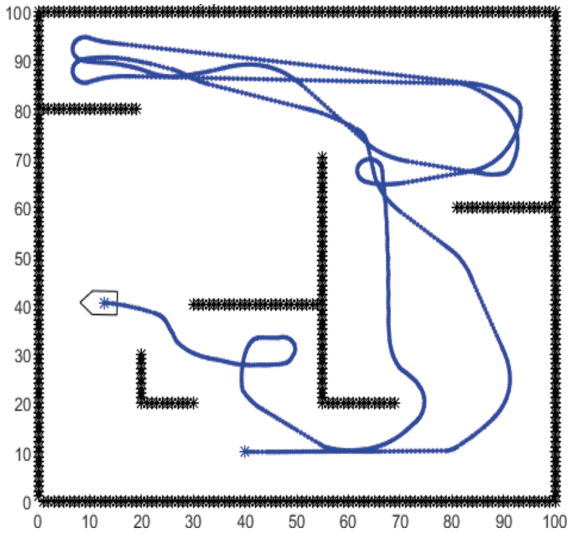

Figure 5. Environments' exploration with one robot

Table 6 depicts the results from both controllers. It is noteworthy that, the DFCM-MRS approach took longer to complete the task in Environment I. In the other two it showed to be at least two times faster considering both iterations and processing time. The similar behavior with the traveled distance and explored area. However, considering the ratio between these two parameters in Environments II and III, the DFCM-MRS covered more area even travelling less in comparison to the FLC approach. This feature represents that the robots will consume less battery to search larger areas in real scenarios, for example.

In the second environment, the corridor-like shape also led the robot to travel the lower area twice, when the randomization algorithm actuated to change the robot's orientation, which allowed the robot to explore the upper section. The last environment also presented a similar situation, in which the robot traveled the lower-diagonal section a couple times before have its orientation changed.

From the motors' pulses given by Fig. 6, it can be noticed that the robots do not entered in collision imminent states (sub-behavior 3) since there are no negative pulses in the motors, which indicate a fast turn. The areas where the pulses are more concentrated are those in which the decision-making process occurs, which suggest to be improved in future work in order to reduce the decision-making processing time.

The explored area results are depicted in Fig. 7. In the first environment, less than 70% of the area was covered by the DFCM, and 83% by the FLC. Considering a standard of 85% as the desired minimum threshold by the authors, both controllers do not reach this ratio. However, these results were improved with Environments II and III. This time, the DFCM-MRS covered an average of 92.27% of the area, and the FLC-MRS 85.81%.

1 Robot	Environment I		Environment II		Environment III	
	FLC	DFCM	FLC	DFCM	FLC	DFCM
Iterations	976.00	1212.00	3701.00	1260.00	5536.00	2215.00
Processing time (s)	339.00	600.95	707.01	192.94	1063.00	331.38
Traveled distance(cm)	508.70	668.75	2062.58	662.33	3524.79	1134.47
Explored area (cm2)	8374.00	6894.00	8800.00	9258.00	8362.00	9196.00

Table 6. Results: One Robot

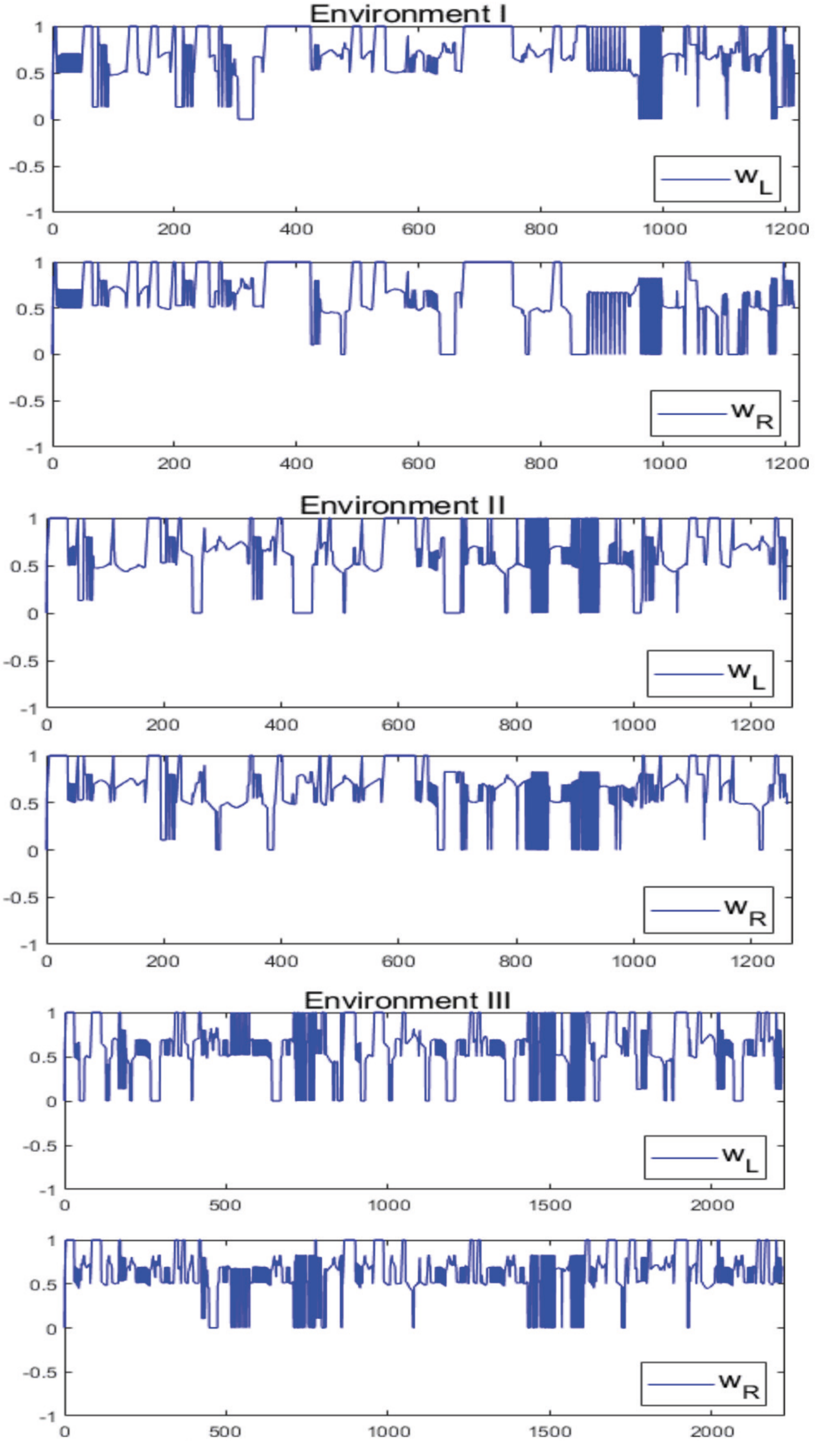

Figure 6. Pulses for the simulations with one robot

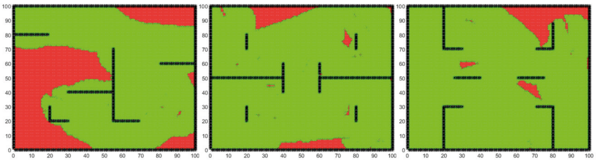

Figure 7. Total explored area with one robot

These parameters can be noticed at the Environment I, where a significant area was not covered by the DFCM-MRS. In this particular case, a victim located in the red section would not be detected nor rescued.

FOUR ROBOTS

After the simulations with a robot working solo, the next step was to increase the number of robots in the proposed MRS. In general, despite the DFCM-MRS showed similar number of iterations as the FLC, its processing time (Table 7) from Environments I and II is significantly lower. In the third one, the difference is due to the fact that the robots 1 and 3 were stuck in the left section of the environment, as seen in Fig. 14. The trails from each robot in the three environments are shown in Figs. 8, 11 and 14.

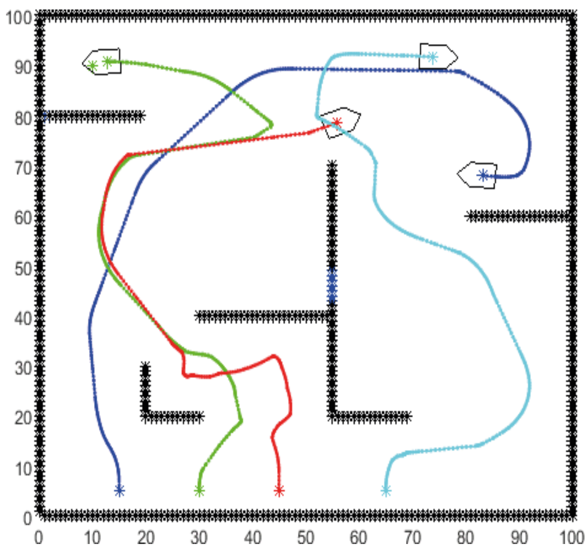

Figure 8. Environment I with four robots

The regions covered by each robot of the MRS are shown in Fig. 10 for Environment I, Fig. 13 for Environment II, and Fig. 16 for the third one; and depict the results from Table 8. Robot 1 covered an inverted L-shaped area. Robots 2 and 3 covered a C-shaped region in the left side of the Environment I. Robot 4 took place at the right side in this scenario. The total area covered by the entire DFCM--MRS (Fig. 10) consists in the interpolation of its individual results. When not considering the number of victims as stopping criterion, there are not critical issues of the non-explored areas: if the simulations continue, the robots are capable of reaching them.

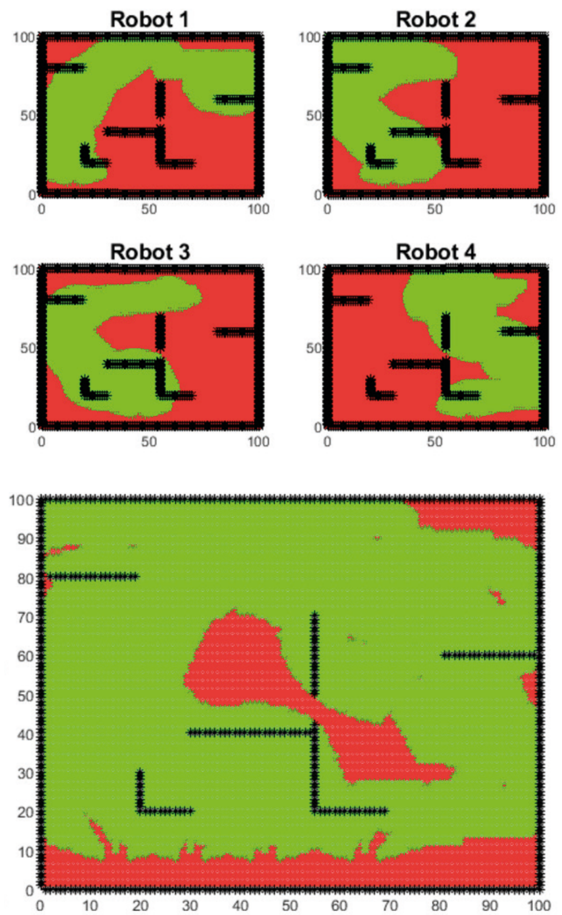

Figure 10. Individual and total explored areas in Environment I

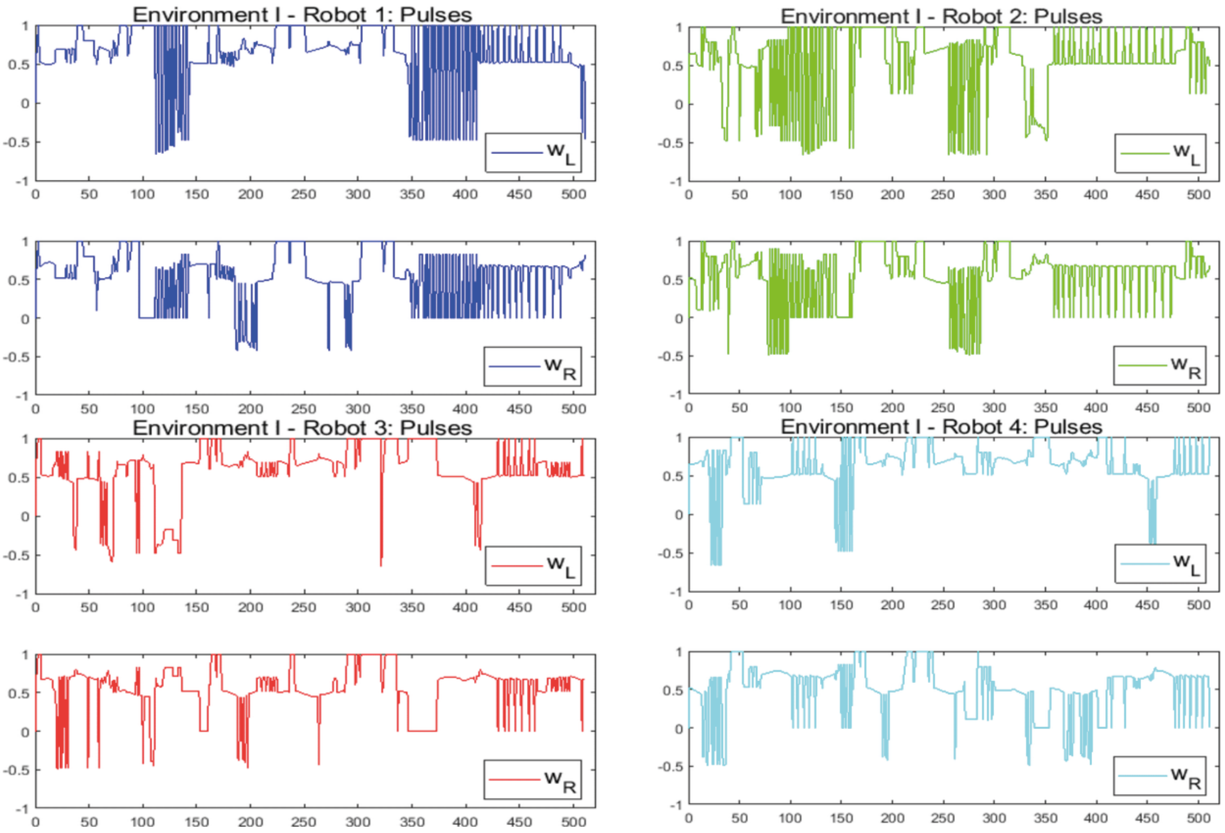

Figure 9. Pulses for Environment I with four robots

4 Robots	Environment I		Environment II		Environment III	
	FLC	DFCM	FLC	DFCM	FLC	DFCM
Iterations	717.00	279.00	727.00	409.00	381.00	356.00
Processing Time (s)	604.67	239.30	713.29	323.47	228.11	541.76

Table 7. Iterations and Processing Times: Four Robots

Explored area (cm2)	Environment I		Environment II		Environment III	
	FLC	DFCM	FLC	DFCM	FLC	DFCM
Robot 1	4178.00	3812.00	3778.00	4082.00	2545.00	2593.00
Robot 2	5239.00	3399.00	4391.00	3345.00	3614.00	3680.00
Robot 3	3724.00	3327.00	4471.00	5058.00	4023.00	3564.00
Robot 4	3176.00	3415.00	6047.00	2582.00	4498.00	3753.00
Total	9277.00	7849.00	9015.00	8739.00	9622.00	9012.00

Table 8. Explored Area: Four Robots

Traveled distance (cm)	Environment I		Environment II		Environment III	
	FLC	DFCM	FLC	DFCM	FLC	DFCM
Robot 1	427.39	166.63	471.06	162.12	224.54	158.83
Robot 2	450.03	143.96	429.58	168.96	192.53	182.71
Robot 3	472.49	136.09	456.00	238.21	268.83	174.10
Robot 4	412.62	142.85	430.02	162.58	216.30	197.27
Total	1762.53	589.53	1786.66	731.87	902.20	712.91

Table 9. Traveled Distance: Four Robots

The results from Tables 8 and 9 are justified since the stop criterion is the rescue of all six victims. In this sense, the DFCM-MRS traveled less and covered a smaller area than the FLC-MRS but, at the same time, the first one rescued the victims faster. This feature grants robots with extended battery life in real scenarios. However, the ratio between traveled distance and explored area indicates that the DFCM approach covers more area per unity of distance.

As seen in Figs. 9, 12, and 15, the concentrated areas indicate dynamic obstacle avoidance, i.e. the robots avoiding collisions among them. In the DFCM approach, negative pulses greater than -0.5 represent fast turns.

Considering the second environment, the main issue was that only one robot reached the upper quadrant, as seen in Figs. 11 and 13. However, the explored area was not prejudiced since presented 87% of cover (Fig. 13), and the individual values of this parameter are close enough to the FLC-MRS, which took longer and consumed more processing time in the first two environments. From Fig. 11, the proximity of robots 1 and 4 can be noticed in Fig. 12, where the pulses alternate repeatedly in the interval from 300 to 400 iterations.

In Environment III, the robots found difficulties as seen in the simulation with a solo robot (Fig. 5). However, in this case (Fig. 14) the sub-behavior of collision imminence between robots worked as a mechanism to change the robots' orientation. This behavior is depicted by robots 3 and 4 near (30-45, 15), between iterations 25 and 50 (Fig. 15): both robots changed its orientation to avoid contact, and this fact leaded robot 3 to search in the right side of the third environment.

The robots explored 90.12% of the Environment III (from Table 8), as shown in Fig. 16. The region surrounding (35, 90) was not explored, again, due to the stopping criterion, i.e. the victims were all found before that area was covered. In this scenario, robots 1 and 2 covered the sides of the environment, while robots 3 and 4 (mainly) covered the central area.

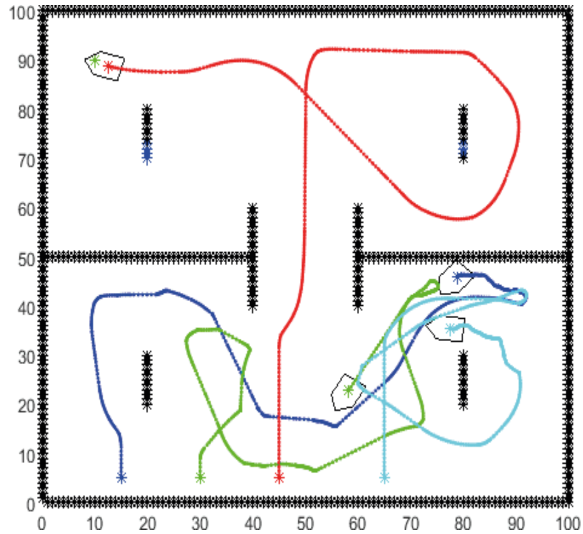

Figure 11. Environment II with four robots

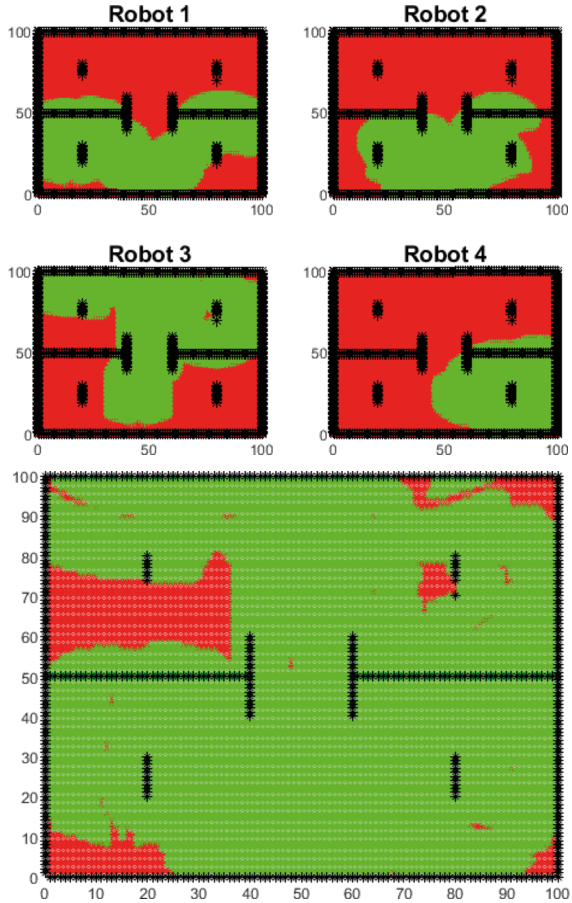

Figure 13. Individual and total explored areas in Environment II

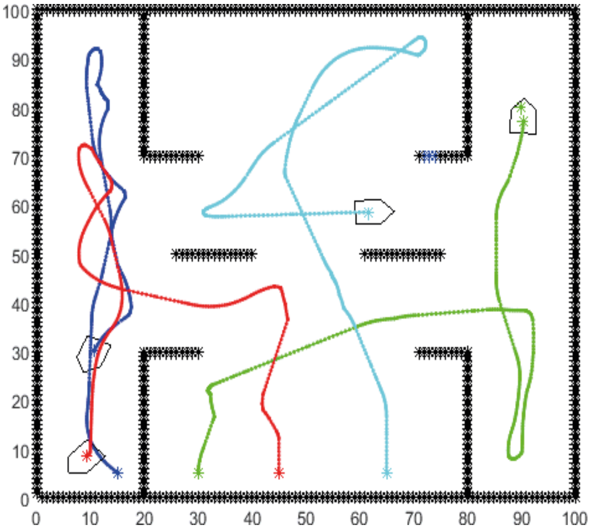

Figure 14. Environment III with four robots

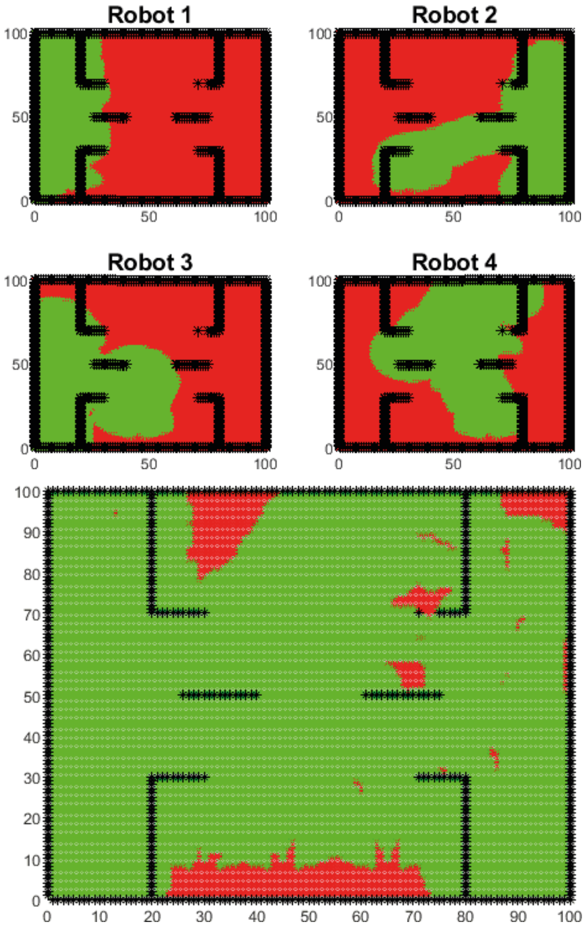

Figure 16. Individual and total explored areas in Environment III

EIGHT ROBOTS

The last proposed scenario deals with eight robots in the MRS. Unlike the others – one and four robots – the DFCM-MRS approach obtained a lower cost in all environments, as observed in Table 10.

According to Table 11, it is possible to verify that DFCM-MRS and FLC-MRS are similar in relation to the covered area. However, the DFCM-MRS approach has better performance than FLC-MRS in the explored area and in the processing time. On the other hand, Table 12 is possible to certify that robots with DFCM approach traveled less in all scenarios.

The simulation results of the first environment are depicted in Fig. 17. That can be noticed that, when comparing the three scenarios (one, four and eight robots), the number of iterations drop significantly. In particular, the DFCM-MRS in the first environment initially lasted 1212 iterations (one robot), and completed the task within 400 iterations with eight robots.

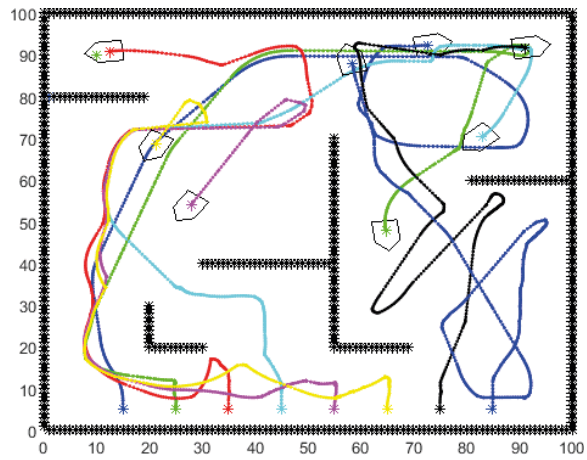

Figure 17. Environment I with eight robots

In Fig. 18 it is possible to verify that the impending collision sub-behavior occurred more frequently due to the increase in the number of robots, when compared to the other scenarios.

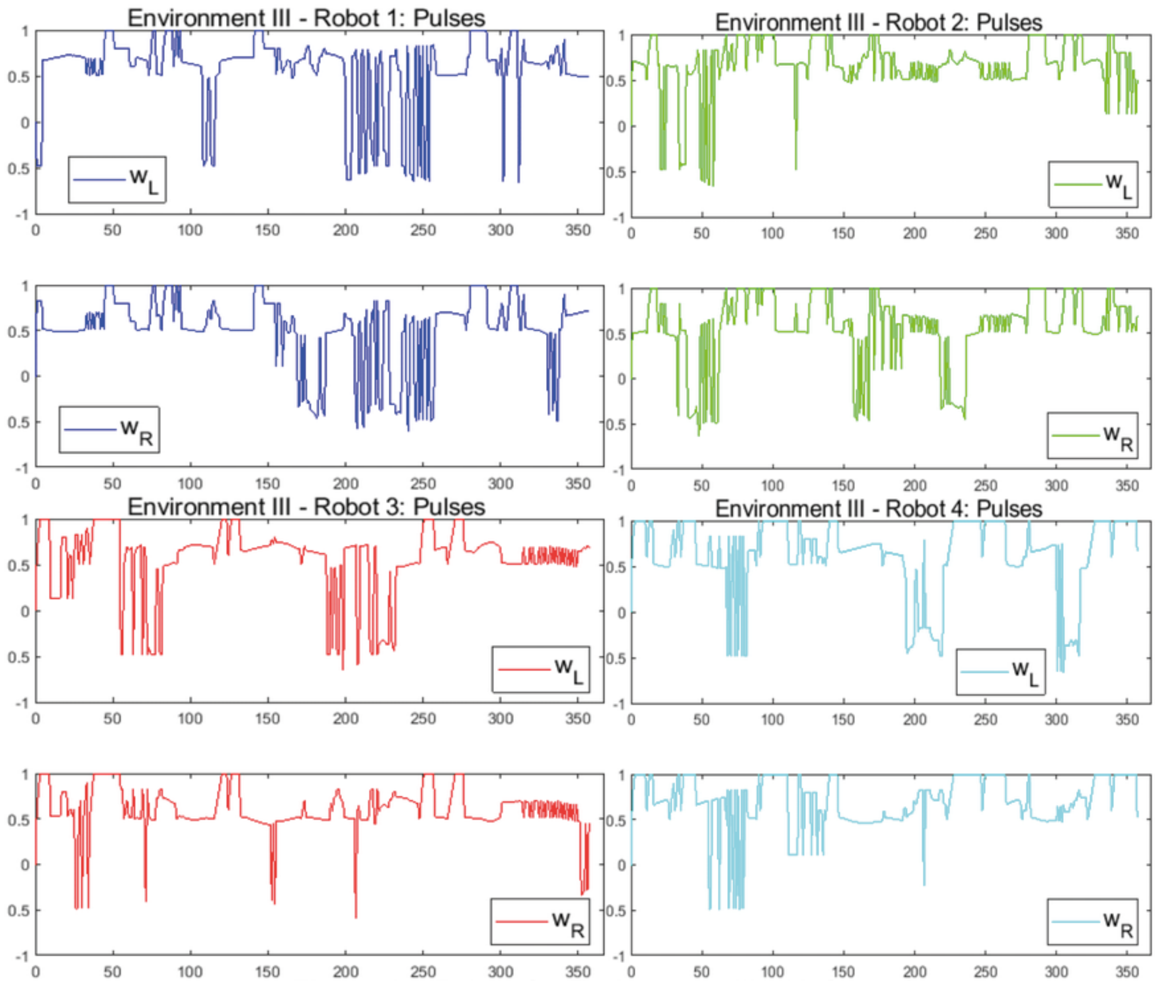

Figure 15. Pulses for Environment III with four robots

8 Robots	Environment I		Environment II		Environment III	
	FLC	DFCM	FLC	DFCM	FLC	DFCM
Iterations	355.00	400.00	435.00	283.00	155.00	162.00
Processing Time (s)	1204.04	647.99	976.84	493.59	487.63	297.41

Table 10. Iterations and Processing Times: Eight Robots

Explored area (cm2)	Environment I		Environment II		Environment III	
	FLC	DFCM	FLC	DFCM	FLC	DFCM
Robot 1	3937.00	4574.00	3352.00	1877.00	2157.00	2280.00
Robot 2	3737.00	5146.00	2990.00	2774.00	2541.00	1310.00
Robot 3	4830.00	4083.00	3282.00	2793.00	1860.00	2348.00
Robot 4	3884.00	4556.00	4896.00	3739.00	2518.00	2240.00
Robot 5	3535.00	4249.00	2850.00	3520.00	2824.00	2622.00
Robot 6	3765.00	3855.00	4886.00	3152.00	2224.00	1641.00
Robot 7	4460.00	4059.00	3046.00	3135.00	2013.00	1746.00
Robot 8	4109.00	3517.00	2287.00	3059.00	2069.00	2293.00
Total	9757.00	9840.00	9475.00	8896.00	8994.00	8787.00

Table 11. Explored Area: Eight Robots

Traveled distance (cm)	Environment I		Environment II		Environment III	
	FLC	DFCM	FLC	DFCM	FLC	DFCM
Robot 1	211.53	218.72	222.87	113.71	85.36	86.90
Robot 2	198.22	214.95	257.74	111.95	100.90	46.07
Robot 3	237.75	190.93	210.65	118.89	76.81	83.89
Robot 4	150.93	192.45	272.00	148.19	93.02	72.06
Robot 5	208.54	183.22	212.78	129.99	104.08	96.85
Robot 6	198.37	161.22	237.39	145.18	94.07	62.69
Robot 7	231.70	197.37	212.61	116.56	73.01	61.47
Robot 8	191.13	197.48	200.31	137.76	86.88	89.38
Total	1628.16	1556.34	1826.35	1022.23	714.13	599.31

Table 12. Traveled Distance: Eight Robots

Since the robots completed the rescue task earlier than the other scenarios, the explored area in Environments II and III was smaller. However, in the first environment the DFCM-MRS covered 98.40% of the area, as can be seen in Fig. 21. In the first environment, robots 5 and 6 covered almost the same portion of area, although not traveled side by side at any point of the simulations.

For the second Environment, robots 1, 2, 3, 6, 7, and 8 faced difficulties to explore the upper section, as seen in Fig. 22. This fact can be explained by the massive concentration of robots in this section aligned with the sub-behavior of obstacle avoidance, which considers a safety distance of 5 cm, limiting the robots' motion as seen in Fig. 19. This MRS behavior illustrates a stressed scenario where the number of robots is higher than the number of victims, since two or three robots could be used to complete the tasks.

However, in the particular case of victims' rescue, a larger number of robots is justified due to the need to reach the goal as fast as possible, since every second here counts to the preservation of the victims' lives.

As can be seen in Fig. 22, only two robots explored the upper section of the second Environment (4 and 5). This fact is depicted in Fig. 23. The sensors' detection through the environment walls (robots 1, 3, 6, 7, and 8) was caused by the white noise inserted in the ultrasonic sensors, and not showed any emergent behavior.

From the total area (Fig. 23), the unexplored regions illustrate the stopping criterion once again. If not implemented, the MRS could continue to explore the environment. The eight-robot MRS configuration was the most succeeded in the third environment in terms of the robots' trails. In this scenario, the lateral areas do not offered challenges due to the number of robots in the environment, as seen in Fig. 24.

When comparing the pulses sent to the motors from Fig. 20, it is noteworthy that robots 2, 4, and 6 showed the imminent collision behavior during whole the simulation time. The area explored by each robot is shown in Fig. 25. The DFCM-MRS robots covered 87.87% of the Environment III, in comparison to 89.94% from the FLC-MRS approach, travelling approximately 115 cm less in total.

Through a brief overviewing of the processing time of FLC and DFCM controllers in the proposed MRS, the perception is that the DFCM-MRS strategy only consumed more processing time than the FLC-MRS two times of nine possible. In the rest of them, it overcomes the FLC-MRS, significantly when using eight robots. This aspect suggests that this approach is highly scalable, i.e. making simple its implementation in larger groups or even swarms of robots.

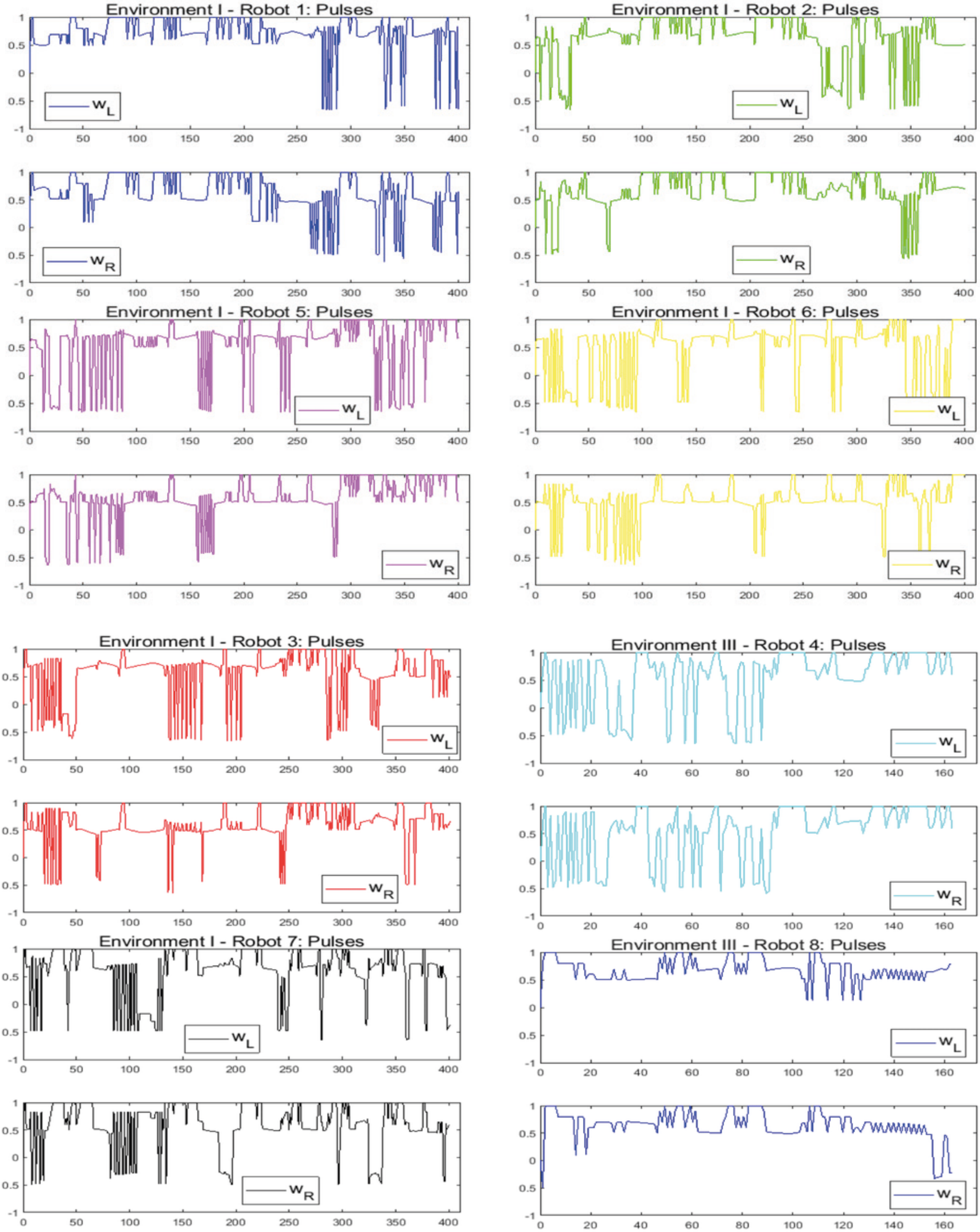

Figure 18. Pulses for Environment I with eight robots

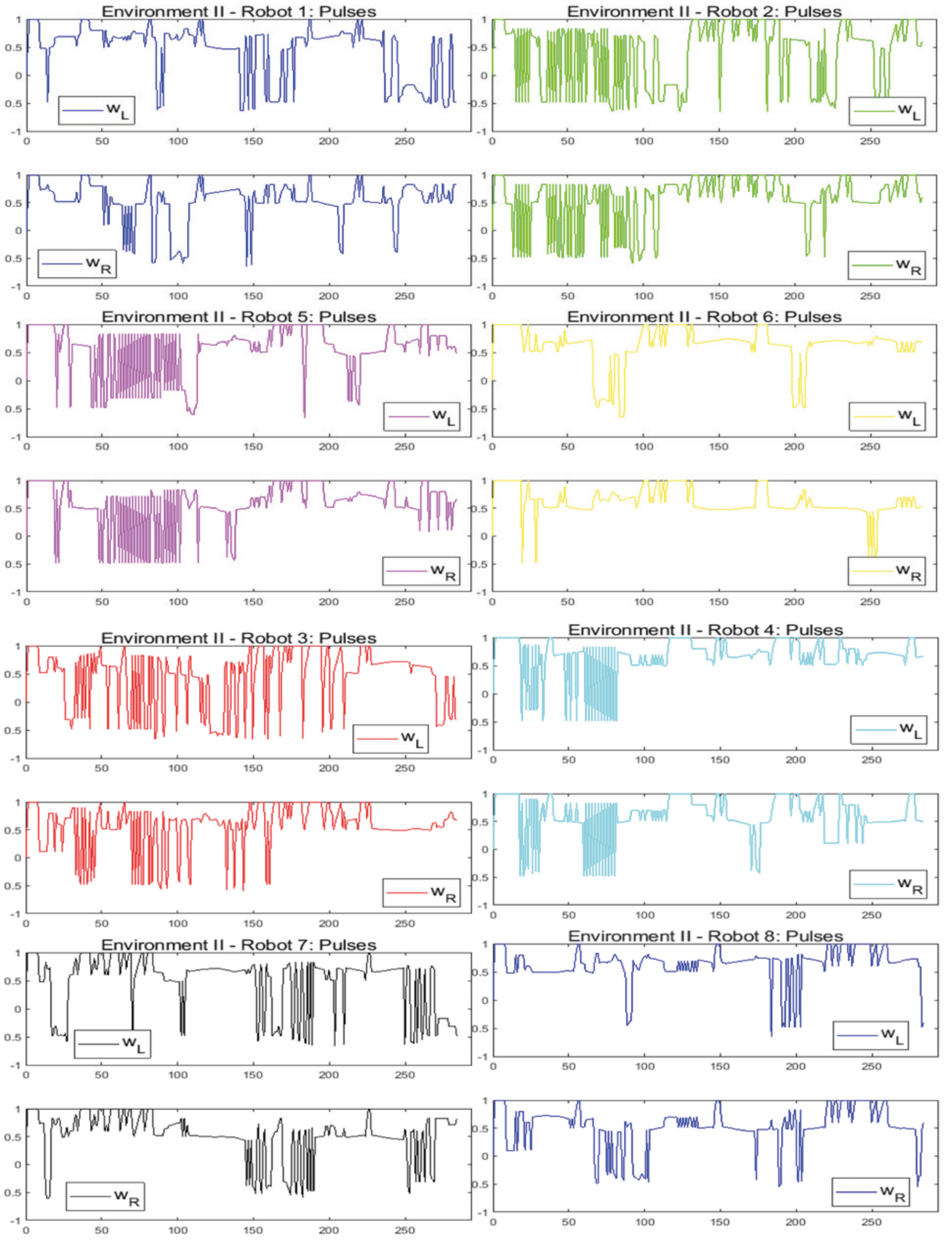

Figure 19. Pulses for Environment II with eight robots

INITIAL PROTOTYPE

Finally, the authors show aspects of the initial working prototype (Fig. 26) which will be used in practical experiments. The robotic platform has four wheels (with the same kinematic model as the simulated one), uses an Arduino® Mega 2560, a DC motor shield, and an adapted battery. In addition to the three ultrasonic sensors, the authors opted to use two laser depth sensors (more precise), which have its data interpolated with the other ones to ensure that the robots will not collide.

A real test circuit was made to verify the behavior of the embedded DFCM. To simulate a victim, a square of electrical tape was used, and its perception made through a color sensor positioned at the bottom of the robot. The initial results are shown in Fig. 27. This real environment was made in the 1:2 scale of Environment I, the numbers describe the sequence of events of the robot.

The proposed operation is to send the victims' GPS location when they are identified, and then send this information to possible human rescuers/fire- fighters to perform the rescue or plan the extraction. New additions in the robot platform are being made to measure both linear and angular velocities, and perform the mapping.

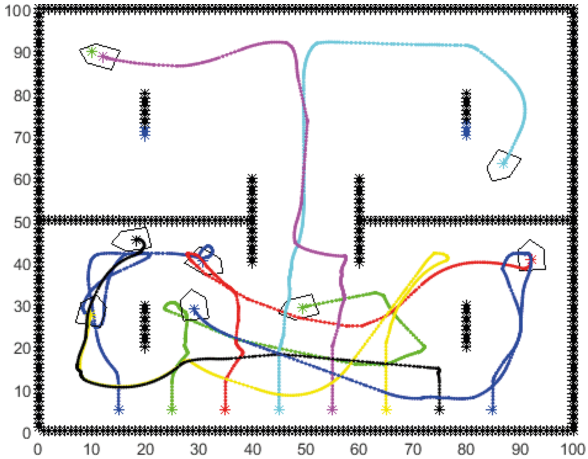

Figure 22. Environment II with eight robots

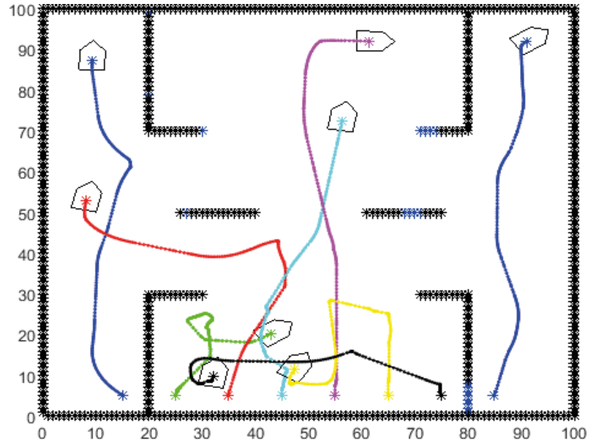

Figure 24. Environment III with eight robots

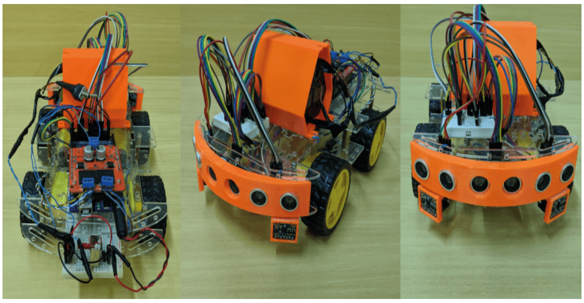

Figure 26. Initial prototype

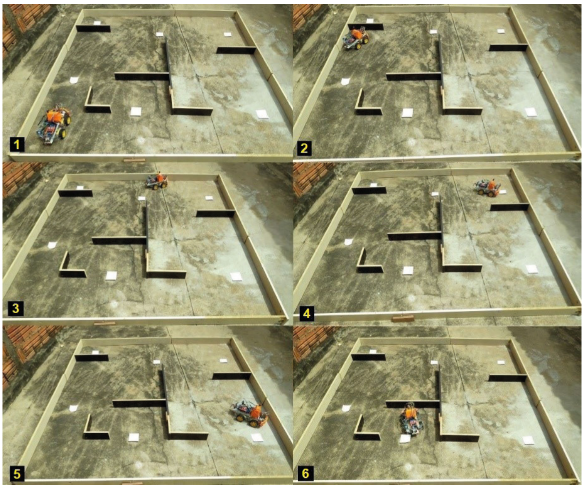

Figure 27. Initial prototype results

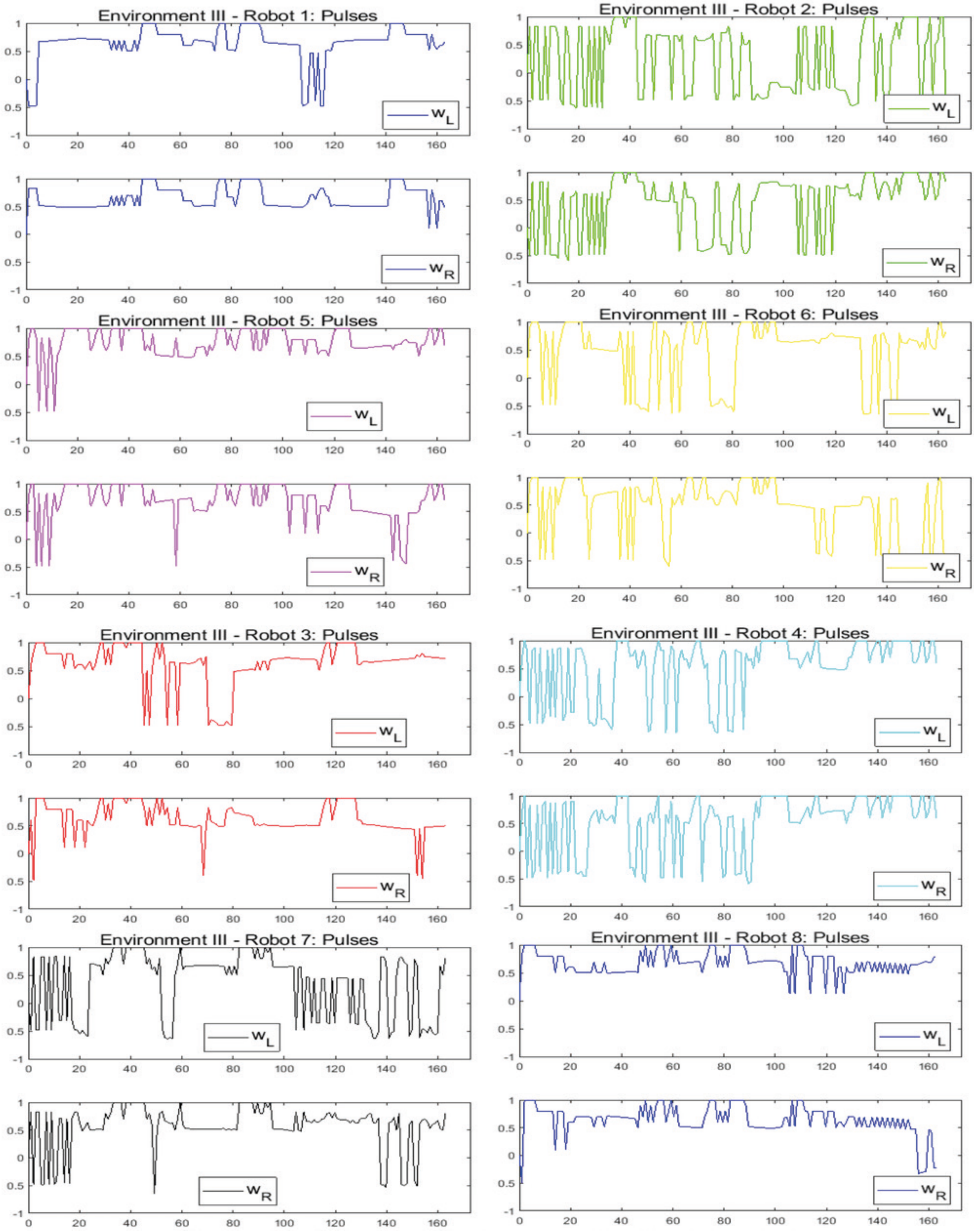

Figure 20. Pulses for Environment III with eight robots

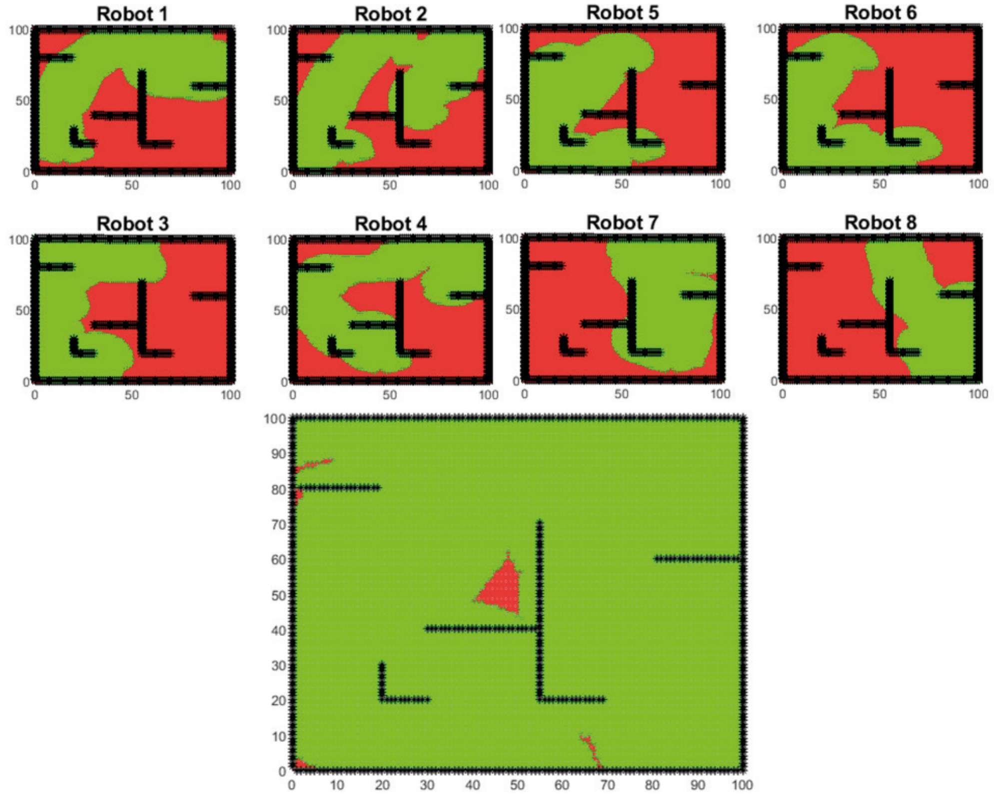

Figure 21. Individual and total explored areas for Environment I with eight robots

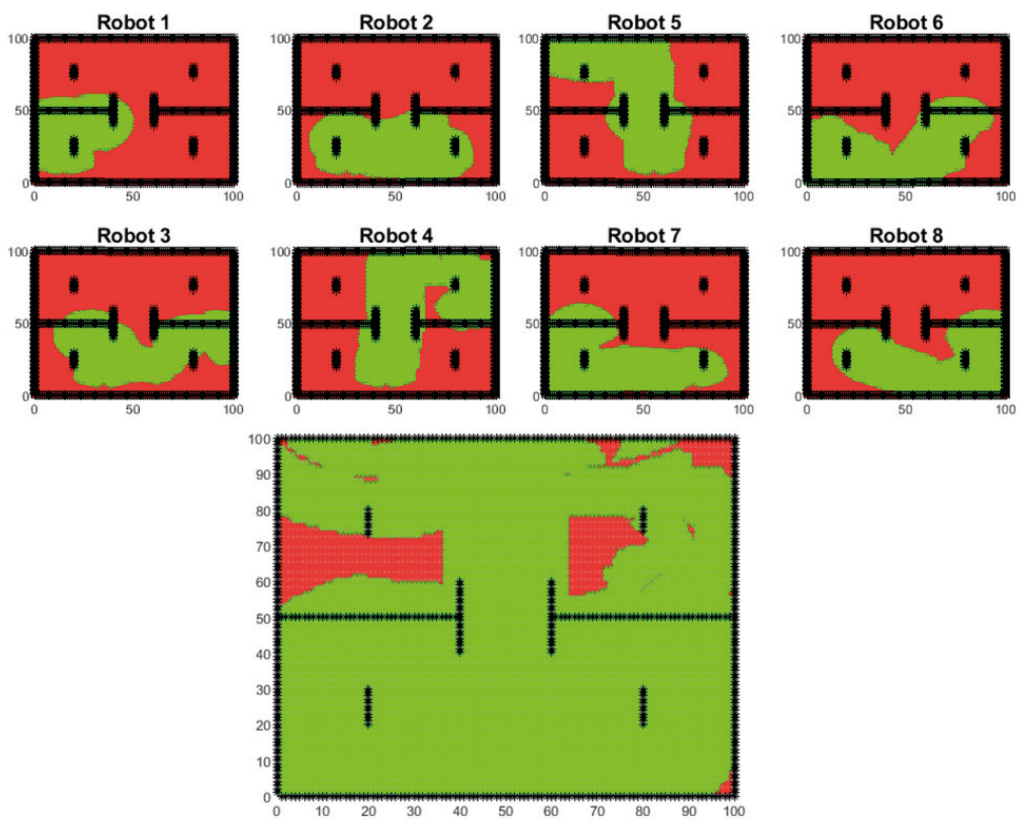

Figure 23. Individual and total explored areas for Environment II with eight robots

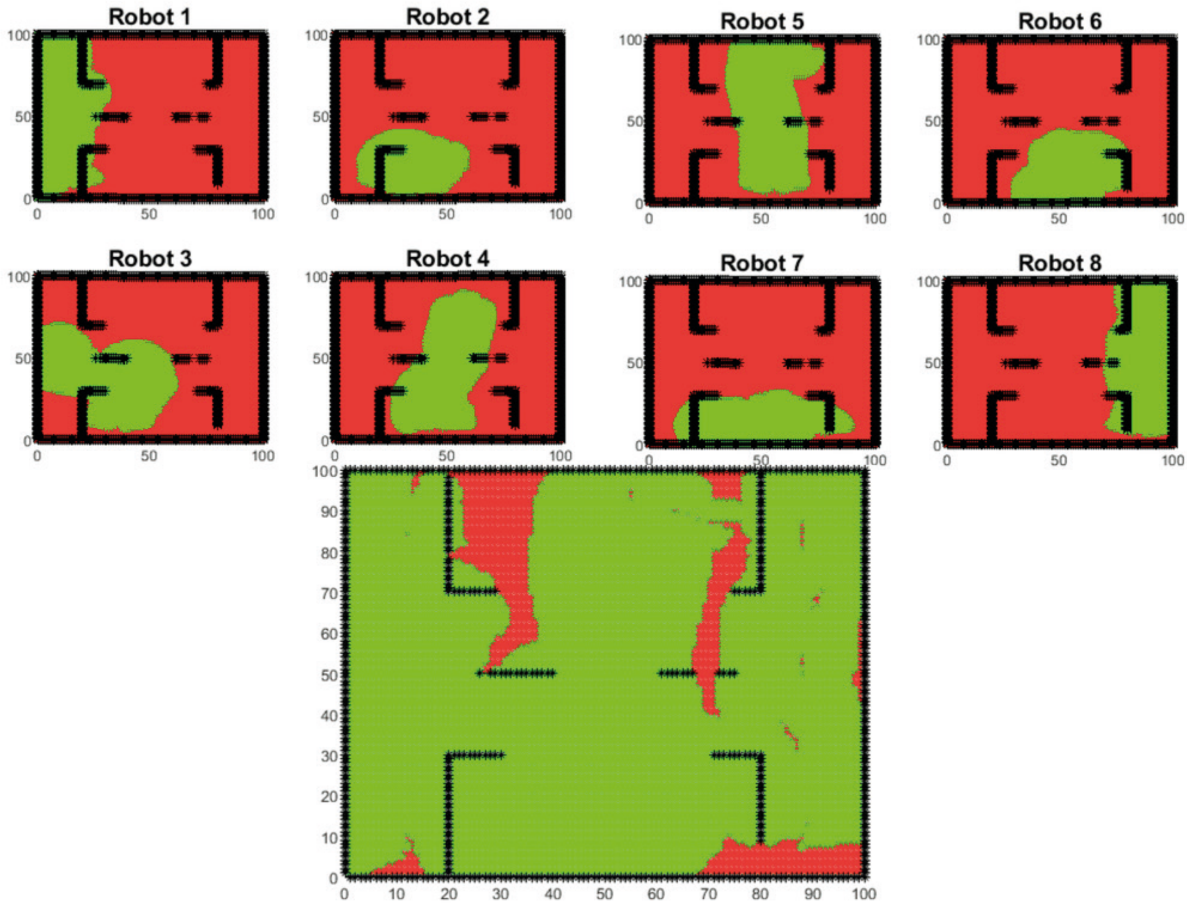

Figure 25. Individual and total explored areas for Environment III with eight robots

CONCLUSIONS

The results obtained from flat environments provided robots with autonomy in exploring environments in the tested scenarios and rescuing victims/targets. Furthermore, the tests refer to the understanding that the control using DFCM can be used in autonomous robots, with better results when compared to the FLC controller.

Moreover, the computational performance (number of iterations and processing time) is improved with the DFCM approach, since it enhances scalability in comparison to the FLC. Despite the fact that the DFCM approach achieved all its goals satisfactorily, the authors are working on an ACO-inspired path-planning algorithm. By employing this strategy, the trail left by the robots will be used as a repulsive pheromone to the other robots over a desired time. The authors expect that this strategy will expand the explored area, resulting e.g. in less battery consumption in real-life applications. Another objective for future work is the alternative of using more robots in non-flat scenarios. A possible strategy to be investigated is the use of a user-guided robot to lead the others, thus aiming at less battery consumption. Finally, authors will focus on updating/changing the sensors, e.g. cameras, responsible for identifying victims: the robots will be capable of locating these targets and evaluate its body temperature (in fire free scenarios) to "medical screening" in order to save as many lives as possible.

REFERENCES

1. Ben-Ari, M.; Mondada, F. *Elements of Robotics*, 1 ed.; Springer International Publishing: Cham, Switzerland, 2018; p. 308. https://doi.org/10.1007/978-3-319-62533-1.

2. Siciliano, B.; Khatib, O., Eds. *Springer Handbook of Robotics*, 2 ed.; Springer-Verlag Berlin Heidelberg: Heidelberg, 2016; p. 2227.

3. Nedjah, N.; Silva Junior, L. Review of methodologies and tasks in swarm robotics towards standardization. *Swarm and Evolutionary Computation* **2019**, *50*, 1–26. https://doi.org/10.1016/j.swevo.2019.100565.

4. Pugh, J.; Martinoli, A. Inspiring and modeling multi-robot search with particle swarm optimization. In Proceedings of the 2007 IEEE Swarm Intelligence Symposium; IEEE: Honolulu, HI, USA, 2007; pp. 332–339. https://doi.org/10.1109/SIS.2007.367956.

5. Mendonça, M.; Kondo, H.S.; de Souza, L.B.; Palácios, R.H.C.; de Almeida, J.P.L.S. Semi-Unknown Environments Exploration Inspired by Swarm Robotics using Fuzzy Cognitive Maps. In Proceedings of the 2019 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE); IEEE: New Orleans, USA, 2019; pp. 1–7.

6. Guo, Y.; Xu, J. Application of Quality Function Deployment and Function-Behavior-Structure Model in the Innovative Design of Hospital Guide Robot. In Proceedings of the 2024 International Conference on Culture-Oriented Science Technology (CoST), 2024, pp. 236–241. https://doi.org/10.1109/CoST64302.2024.00053.

7. Barca, J.C.; Sekercioglu, Y.A. Swarm robotics reviewed. *Robotica* **2013**, *31*, 345–359. https://doi.org/10.1017/S026357471200032X.

8. Bonabeau, E.; Dorigo, M.; Theraulaz, G. *Swarm Intelligence: From Natural to Artificial Systems*, 1 ed.; Oxford University Press: New York, USA, 1999; p. 320. https://doi.org/10.1017/CBO9781107415324.004.

9. Corke, P. *Robotics, Vision and Control. Fundamental Algorithms in MATLAB*, 1 ed.; Springer-Verlag Berlin Heidelberg: Heidelberg 2011; p. 570, [978 3 642 20143 1]. https://doi.org/10.1108/ir.2012.04939faa.005.

10. De Rango, F.; Palmieri, N.; Yang, X.S.; Marano, S. Swarm robotics in wireless distributed protocol design for coordinating robots involved in cooperative tasks. *Soft Computing* **2018**, *22*, 4251–4266, [1804.08096]. https://doi.org/10.1007/s00500-017-2819-9.

11. Palmieri, N.; Yang, X.S.; Rango, F.D.; Santamaria, A.F. Self-adaptive decision-making mechanisms to balance the execution of multiple tasks for a multi-robots team. *Neurocomputing* **2018**, *306*, 17–36. https://doi.org/10.1016/j.neucom.2018.03.038.

12. Bayindir, L. A review of swarm robotics tasks. *Neurocomputing* **2016**, *172*, 292–321. https://doi.org/10.1016/j.neucom.2015.05.116.

13. Din, A.; Jabeen, M.; Zia, K.; Khalid, A.; Saini, D.K. Behavior-based swarm robotic search and rescue using fuzzy controller. *Computers & Electrical Engineering* **2018**, *70*, 53–65. https://doi.org/10.1016/J.COMPELECENG.2018.06.003.

14. Bakhshipour, M.; Ghadi, M.J.; Namdari, F. Swarm robotics search & rescue: a novel artificial intelligence-inspired optimization approach. *Applied Soft Computing* **2017**, *57*, 708–726. https://doi.org/10.1016/j.asoc.2017.02.028.

15. Couceiro, M.S.; Vargas, P.A.; Rocha, R.P.; Ferreira, N.M. Benchmark of swarm robotics distributed techniques in a search task. *Robotics and Autonomous Systems* **2014**, *62*, 200–213. https://doi.org/10.1016/J.ROBOT.2013.10.004.

16. Venayagamoorthy, G.K.; Grant, L.L.; Doctor, S. Collective robotic search using hybrid techniques: Fuzzy logic and swarm intelligence inspired by nature. *Engineering Applications of Artificial Intelligence* **2009**, *22*, 431–441. https://doi.org/10.1016/J.ENGAPPAI.2008.10.002.

17. Mendonça, M.; Arruda, L.V.R.; Chrun, I.R.; Da Silva, E.S. Hybrid Dynamic Fuzzy Cognitive Maps Evolution for Autonomous Navigation System. In Proceedings of the 2015 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE); IEEE: Istanbul, Turkey, 2015; pp. 1–7. https://doi.org/10.1109/FUZZ-IEEE.2015.7337855.

18. Mendonça, M.; Da Silva, E.S.; Chrun, I.R.; de Arruda, L.V.R. Hybrid Dynamic Fuzzy Cognitive Maps and Hierarchical Fuzzy Logic controllers for Autonomous Mobile Navigation. In Proceedings of the 2016 IEEE International Conference on Fuzzy Systems, FUZZ-IEEE 2016; IEEE: Vancouver, BC, Canada, 2016; pp. 2516–2521. https://doi.org/10.1109/FUZZ-IEEE.2016.7738010.

19. Mendonça, M.; Chrun, I.R.; Neves-Jr, F.; Arruda, L.V. A cooperative architecture for swarm robotic based on dynamic fuzzy cognitive maps. *Engineering Applications of Artificial Intelligence* **2017**, *59*, 122–132. https://doi.org/10.1016/j.engappai.2016.12.017.

20. Soares, P.P.; de Souza, L.B.; Mendonça, M.; Palácios, R.H.C.; de Almeida, J.P.L.S. Group of Robots Inspired by Swarm Robotics Exploring Unknown Environments. In Proceedings of the 2018 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE); IEEE: Rio de Janeiro, Brazil, 2018; pp. 1–7. https://doi.org/10.1109/FUZZ-IEEE.2018.8491631.

21. Russell, S.; Norvig, P. *Artificial Intelligence: A Modern Approach*, 3 ed.; Prentice Hall: Upper Saddle River, USA, 2009; p. 1132, [arXiv:arXiv:gr-qc/9809069v1]. https://doi.org/10.1017/S0269888900007724.

22. Brooks, R.A. A Robust Layered Control System For A Mobile Robot. *IEEE Journal on Robotics and Automation* **1986**, *2*, 14–23, [1010.0034]. https://doi.org/10.1109/JRA.1986.1087032.

23. Kumova, B.; Heye, S.B. A survey of robotic agent architectures. In Proceedings of the 2017 International Artificial Intelligence and Data Processing Symposium (IDAP); IEEE: Malatya, Turkey, 2017; pp. 1–8.

24. Zadeh, L. A fuzzy-algorithmic approach to the definition of complex or imprecise concepts. *International Journal of Man-Machine Studies* **1976**, *8*, 249–291. https://doi.org/10.1016/S0020-7373(76)80001-6.

25. Braitenberg, V. *Vehicles: Experiments in Synthetic Psychology*, 1 ed.; MIT Press: Cambridge, USA, 1986; p. 186, [arXiv:1011.1669v3]. https://doi.org/10.2307/2185146.

26. Ross, T.J. *Fuzzy logic with engineering applications*, 3 ed.; John Wiley & Sons: New Mexico, USA, 2010; p. 585.

27. Zadeh, L.A. Fuzzy Sets. *Information and control* **1965**, *353*, 338–353. https://doi.org/10.1016/S0019-9958(65)90241-X.

28. Stach, W.; Kurgan, L.; Pedrycz, W.; Reformat, M. Genetic learning of fuzzy cognitive maps. *Fuzzy Sets and Systems* **2005**, *153*, 371–401. https://doi.org/10.1016/J.FSS.2005.01.009.

29. Nápoles, G.; Papageorgiou, E.I.; Bello, R.; Vanhoof, K. On the convergence of sigmoid Fuzzy Cognitive Maps. *Information Sciences* **2016**, *349-350*, 154–171. https://doi.org/10.1016/J.INS.2016.02.040.

30. Acampora, G.; Pedrycz, W.; Vitiello, A. A Competent Memetic Algorithm for Learning Fuzzy Cognitive Maps. *IEEE Transactions on Fuzzy Systems* **2015**, *23*, 2397–2411. https://doi.org/10.1109/TFUZZ.2015.2426311.

31. Mendonça, M.; Angelico, B.; Arruda, L.V.R.; Neves-Jr, F. A dynamic fuzzy cognitive map applied to chemical process supervision. *Engineering Applications of Artificial Intelligence* **2013**, *26*, 1199–1210. https://doi.org/10.1016/j.engappai.2012.11.007.

32. Papageorgiou, E.I., Ed. *Fuzzy Cognitive Maps for Applied Sciences and Engineering*; Springer-Verlag Berlin Heidelberg: Heidelberg, 2014; p. 395. https://doi.org/10.1007/978-3-642-39739-4.

33. Papageorgiou, E.I.; Froelich, W. Multi-step prediction of pulmonary infection with the use of evolutionary fuzzy cognitive maps. *Neurocomputing* **2012**, *92*, 28–35. https://doi.org/10.1016/j.neucom.2011.08.034.

34. Vašcˇák, J. Approaches in adaptation of fuzzy cognitive maps for navigation purposes. In Proceedings of the 2010 IEEE 8th International Symposium on Applied Machine Intelligence and Informatics (SAMI); IEEE: Herlany, Slovakia, 2010; pp. 31–36. https://doi.org/10.1109/SAMI.2010.5423716.

35. Koulouriotis, D.E.; Diakoulakis, I.E.; Emiris, D.M. Anamorphosis of fuzzy cognitive maps for operation in ambiguous and multi-stimulus real world environments. In Proceedings of the 10th IEEE International Conference on Fuzzy Systems; IEEE: Melbourne, Victoria, Australia, 2001; pp. 1156–1159. https://doi.org/10.1109/FUZZ.2001.1008860.

36. Mpelogianni, V.; Groumpos, P.P. A revised approach in modeling fuzzy cognitive maps. In Proceedings of the 2016 24th Mediterranean Conference on Control and Automation (MED); IEEE: Athens, Greece, 2016; pp. 350–354. https://doi.org/10.1109/MED.2016.7536070.

37. Kosko, B. Fuzzy cognitive maps. *International Journal of Man-Machine Studies* **1986**, *24*, 65–75. https://doi.org/10.1016/S0020-7373(86)80040-2.

38. Karagiannis, I.; Groumpos, P. Input-Sensitive Fuzzy Cognitive Maps. *International Journal of Computer Science* **2013**, *10*, 143–151.

39. Mendonça, M.; Arruda, L.V.R.; Neves-Jr, F. Autonomous navigation system using Event Driven-Fuzzy Cognitive Maps. *Applied Intelligence* **2011**, *37*, 175–188. https://doi.org/10.1007/s10489-011-0320-1.

40. Vašcˇák, J. Navigation Based on Fuzzy Cognitive Maps for Needs of Ubiquitous Robotics. In Proceedings of the SAMI 2019 -IEEE 17th World Symposium on Applied Machine Intelligence and Informatics, Proceedings; IEEE: Herlany, Slovakia, 2019; pp. 123–128. https://doi.org/10.1109/SAMI.2019.8782773.

41. Motlagh, O.; Tang, S.H.; Ismail, N.; Ramli, A.R. An expert fuzzy cognitive map for reactive navigation of mobile robots. *Fuzzy Sets and Systems* **2012**, *201*, 105–121. https://doi.org/10.1016/j.fss.2011.12.013.

42. Amirkhani, A.; Shirzadeh, M.; Shojaeefard, M.H.; Abraham, A. Controlling wheeled mobile robot considering the effects of uncertainty with neuro-fuzzy cognitive map. *ISA Transactions* **2020**, *100*, 454–468. https://doi.org/10.1016/j.isatra.2019.12.011.

43. Wang, D.; Wang, H.; Liu, L. Unknown environment exploration of multi-robot system with the FORDPSO. *Swarm and Evolutionary Computation* **2016**, *26*, 157–174. https://doi.org/10.1016/j.swevo.2015.09.004.

44. Khamis, A.; Hussein, A.; Elmogy, A. Multi-robot Task Allocation: A Review of the State-of-the-Art. In *Cooperative Robots and Sensor Networks*, 2015 ed.; Koubâa, A.; Martínez-de Dios, J., Eds.; Springer: Cham, Switzerland, 2014; Vol. 554, pp. 31–51. https://doi.org/10.1007/978-3-642-55029-4.

45. Parker, L.E. Distributed Intelligence: Overview of the Field and its Application in Multi-Robot Systems. *Journal of Physical Agents* **2008**, *2*, 5–14. https://doi.org/10.14198/JoPha.2008.2.1.02.

46. Mavrovouniotis, M.; Li, C.; Yang, S. A survey of swarm intelligence for dynamic optimization: Algorithms and applications. *Swarm and Evolutionary Computation* **2017**, *33*, 1–17. https://doi.org/10.1016/j.swevo.2016.12.005.

47. Beni, G. From Swarm Intelligence to Swarm Robotics. In Proceedings of the Swarm Robotics. SR 2004. Lecture Notes in Computer Science; S¸ahin, E.; Spears, W., Eds.; Springer Berlin Heidelberg: Heidelberg, 2004; Vol. 3342, pp. 1–9. https: //doi.org/10.1007/978-3-540-30552-1.

48. Colares, R.G.; Chaimowicz, L. A Novel Distance Cost Approach for Multi-robot Integrated Exploration. In Proceedings of the 2015 12th Latin American Robotics Symposium and 2015 3rd Brazilian Symposium on Robotics (LARS-SBR); IEEE: Uberlandia, Brazil, 2015; pp. 192–197. https://doi.org/10.1109/LARS-SBR.2015.46.

49. Colares, R.G.; Chaimowicz, L. The Next Frontier: Combining Information Gain and Distance Cost for Decentralized Multi-Robot Exploration. In Proceedings of the SAC '16 Proceedings of the 31st Annual ACM Symposium on Applied Computing; ACM: Pisa, Italy, 2016; pp. 268–274.