Wheeled Mobile Robot Model and Cooperative Formation Control

ADRIAN KORODI, MARIAN CORMAN
Department of Automation and Applied Informatics
Faculty of Automation and Computers
"Politehnica" University of Timisoara
Bd. Vasile Parvan no. 2, 300223
ROMANIA
adrian.korodi@aut.upt.ro, corman_marian_89@yahoo.com

Abstract: - As a part of a more complex study about cooperative search and rescue wheeled mobile robots, the current work focuses first on defining a complete mathematical model of a wheeled mobile robot that moves in partially known environments. The implemented Matlab/Simulink model of the kinematic, sensorial and fuzzy control parts assures the movement of the robot from an initial point to the destination avoiding possible obstacles. The paper continues to present the development of a control strategy that defines the leader-follower concept for cooperative wheeled mobile robots which have to keep a certain formation and to reach a target considering possible obstacles along the path.

Key-Words: wheeled mobile robots, mathematical modeling, cooperative navigation, fuzzy control, multi-robot control.

1 Introduction

The wheeled mobile robots (WMR) are representing a challenging research domain. The mobile robots are mechatronic structures and therefore the demanded improvements are from various domains. Our investigation is focused on modeling and controlling WMRs. The general purpose of the research is to control groups of cooperative robots, particularly search and rescue robots, which are moving in unknown or partially known environments.

There are several steps that have to be undertaken in order to reach the foreseen objective, the first being to develop a WMR model including the kinematic part, the sensorial part and the control part. The robot has to be able to sense objects, to avoid them and to reach the final destination.

Then, it is necessary to divide the work into three directions, depending on the required task, all of them being essential for the final goal, to control groups of robots:

- The first task that is required for groups of cooperative robots is to move from an initial point to a destination keeping a certain formation. The main purpose is that all the robots must be present in the mentioned formation at the destination. This task is particularly required when a group of robots is moving in certain rooms (door to door movement). The structure of the environment is generally known, but unknown objects may be present along the way. Usually, this task is prior for those that will be described in the followings. A control strategy based on an artificial potential function in an environment without obstacles, together with individual wheel slippage controllers is presented in [1]. Also, a control strategy that assures navigation of cooperative robots in a certain formation is presented in [2] and [3]. The formation of robots is assured throughout the movement in the environment by a spring-damper structure. Also, there is a fuzzy approach in controlling the formation in [4], and the results are provided in an environment with no obstacles. The study from [5] reveals an approach where the robots are moving in formation but the shape of the formation, respectively the position of the robots towards each other is not defined. The robots group in an ad-hoc manner.
- The second task is required when the group of cooperative robots has to reach a target with a known position. The main objective of this task is to arrive at the mentioned target as soon as possible by any of the robots. The control strategy is relying on a set of general rules that governs the movement. Each of the robots has to have its own hierarchically smaller controller that assures the obstacle avoidance. Usually, until meeting the first obstacle, the robots are

moving in a certain formation. In some situations multiple targets are present and a group of robots must accomplish tasks at each destination point concomitantly. In [6], an algorithm is presented for choosing which robot should be sent for the current target. The issue that may occur when multiple robots are trying to reach the same target is the congestion. In [7], a control algorithm is proposed in order to avoid traffic congestions. In papers like [8] and [9], a visual SLAM is used by a team of robots and a general map is created using the individual maps of each robots.

- The third task supposes that the position of the target is unknown and the group of cooperative robots is used to find it. The target is identified through certain particularities. The space exploration is realized in a cooperative manner, following a defined strategy. The search ends when a robot finds the target and it provides the necessary coordinates to the others. A rule based neuro-fuzzy navigation is presented in [10], and a sensor-robot search and rescue perspective is presented in [11]. Papers like [12], [13] are approaching unknown space exploration problems but without the specific objective of search and rescue. Approaches like [14] are focusing on improving the mechatonic structure of the mobile robots operating in unknown environments (field exploration).

  Alternative approaches to the study of collaborations between robotic agents could be represented by the methodologies used by researchers active in the agent based modeling field as it has been done, for instance, in [15], for modeling a financial domain where interacting entities are individuals.

The current paper focuses on developing the model for a WMR which moves in an environment with obstacles, including the kinematic, the sensorial and the control parts. Also, the paper provides a solution for cooperative robot control that involves moving from an initial point to the final point in a certain formation, considering an environment with obstacles. The cooperative behavior is relying on a leader robot and follower robots. Fuzzy controllers are designed for each robot in order to avoid obstacles. Also, fuzzy controllers are implemented for the follower robots in order to evolve in the desired formation.

The paper describes first the mathematical model used to simulate the movement of the robot in an environment without obstacles and the methodology to calculate the distance and the angle that has to be covered in order to meet the destination. In order to obtain a complete Matlab/Simulink model of the robot, the third chapter describes the model of the sensorial part, which is responsible of detecting obstacles along the path. The fourth part is describing the fuzzy controllers used for the individual navigation of the WMR and the principle that guides the cooperative leader-follower navigation in an environment with obstacles. The fifth chapter is illustrating the results for the individual movement and for the cooperative navigation of the WMRs, and finally, conclusions and development directions are presented.

2 Mathematical Modeling of the WMR

The general block scheme used to represent the dynamics of the robot, the sensorial part and the control part for a WMR is presented in fig. 1.

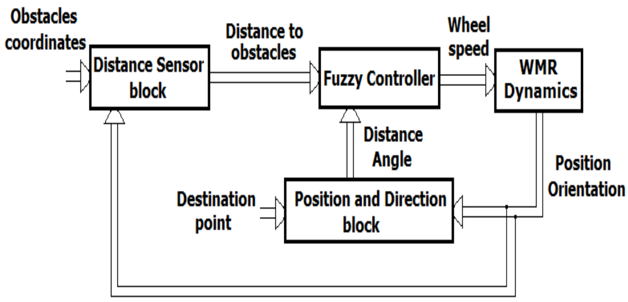

Fig. 1 General block scheme of the robots model

The mathematical model (implemented in the Matlab/Simulink environment) that describes the dynamics of the mobile robot in fig. 1, is represented by the WMR Dynamics (WMRD) block. The subsystem that assures the calculus of the distance and the angle that has to be covered in order to reach the destination is the Position and Direction block.

2.1 WMR Dynamics Block

The WMRD block is responsible for calculating the new position of the robot and its orientation, by using the left and right wheels speed. The inputs and the outputs of the WMRD block are illustrated in fig. 2.

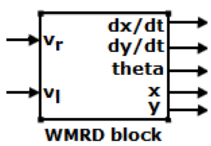

Fig. 2 The WMRD block

The two inputs of WMRD block are vr and vl representing the speed of the right and left wheal provided by the Fuzzy Controller.
Using the standard kinematic equations, the dynamics of the WMR is described by the following equations [16], [17]:

$$\dot{\theta} = \frac{v_l - v_r}{l} \quad (1)$$

$$\dot{x} = \frac{v_l + v_r}{2} \sin\theta \quad (2)$$

$$\dot{y} = \frac{v_l + v_r}{2} \cos\theta \quad (3)$$

where l is the distance between the wheels, and θ is the angle theta (as it is shown in fig. 6).
The outputs of WMRD block are:

- dx/dt, the horizontal speed on the X-axis;
- dy/dt, the vertical speed on the Y-axis;
- theta, the orientation of the robot;
- (x, y), the position of the robot.

2.2 Position and Direction Block

The Position and Direction Block (P&D) determines the distance and the angle that have to be covered from the current position and orientation of the robot in order to navigate towards the target. The inputs and outputs of the P&D block are illustrated in fig. 3.

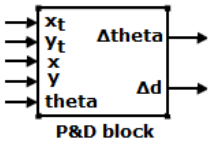

Fig. 3 The Position and Direction block

The inputs and the outputs of the Position and Direction block are:

- (x, y), theta, the current position and the angle of the robot;
- (x_t, y_t), the targets coordinates;
- Δtheta, the angle correction that has to be made in order to go straight towards the target;
- Δd, the distance to the destination. Δd is calculated using the simple equation that describes the distance between two points.

Considering that $theta_F$ is the desired angle associated to the straight distance from the current position towards the target,

$$theta_F = \text{atan}\left(\frac{x_t - x}{y_t - y}\right) \quad (4)$$

the angle correction Δtheta is provided at the output considering three possible situations:

- IF ($theta_F$>0) THEN *Δtheta*=theta- $theta_F$;
- IF ($theta_F$<0) AND (*theta*>0) THEN *Δtheta*=*theta*- $theta_F$ +π ;
- IF ($theta_F$<0) AND (*theta*<0) THEN *Δtheta*=*theta*- $theta_F$ -π;

The two outputs of the Position and Direction block are sent to the Fuzzy Controller.

3 Sensorial Part

In order to move autonomously in unknown or partially known environments, a WMR has to interact with the environment through a localization module used to scan the surroundings and to determine the robot's position. As we can see in fig. 4a (representing the X80 robot) from [18], generally for the localization module, the WMRs are basically equipped with: distance sensors (sonar and/or infrared sensors) to determine the distances to the obstacles along the path, encoders (positioned at the wheels) to calculate permanently the current position and camera. In order to have a model of the mobile robot that behaves as close as possible to the reality, it is necessary to model the distance sensors in order to simulate the obstacle avoidance. Therefore, this chapter is presenting the Matlab/Simulink model of the sensorial part.

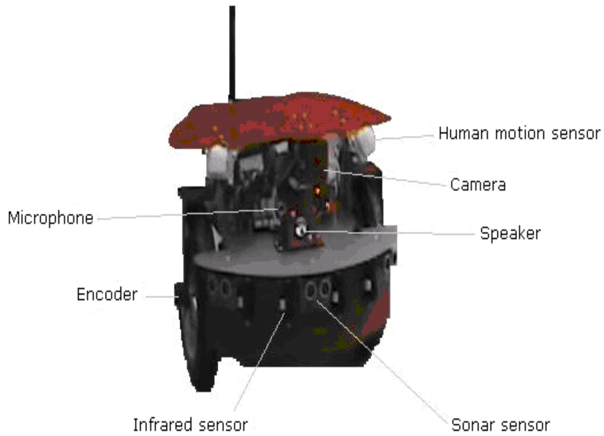

Fig. 4a The X80 robot

3.1 General Structure

The purpose of the Distance Sensor (DS) block is to calculate the distances from the robot to the obstacles along the path, the distances being provided to the controller. The DS block is illustrated in fig. 5.

The inputs of the DS block are:

- (x, y), representing the Cartesian coordinates of the current position of the robot, calculated and provided by the WMRD block;
- (*theta*), representing the orientation of the robot (the angle with the vertical axis). The value of *theta* is also provided by the WMRD block.
- (*px1, py1, px2, py2*), representing one dimensional arrays, containing the coordinates that are defining the segments considered to be obstacles in the environment. The ensemble (*px1[i], py1[i], px2[i], py2[i]*) represents one segment defined through two points, where $i = \overline{1;n}$, and n is the number of lines. As an example, fig. 4b is presenting the arrays that constitute the considered obstacles (4 segments).

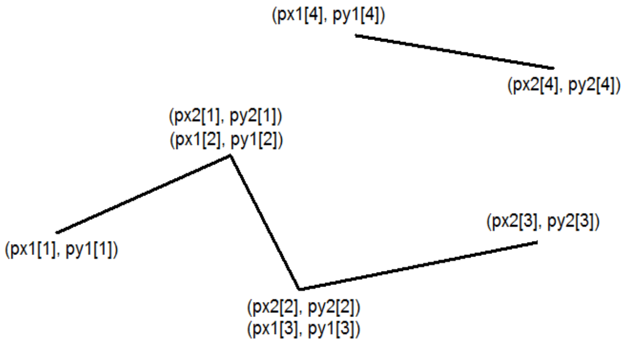

Fig. 4b Associating the (*px1, py1, px2, py2*) input vectors to obstacles

Also, the obstacles from fig. 15 are stored in the *px1, py1, px2, py2* arrays as:

px1=[-50 -100 -150 -200 -250 -300];
py1=[100 200 100 -100 200 100];
px2=[-52 -101 -151 -202 -253 -301];
py2=[-160 -50 -200 100 -100 -200];

According to the current application, the outputs of the DS block are e_1, e_2 and e_3, representing the distances from the robot to the obstacles from the environment, provided by three distance sensors. The distance sensors are placed on the robot in order to provide the frontal distance, and $\pm$ 30 degrees left and right from the direction of movement (see fig. 6). The distance sensors are considered to have a range of 30cm.

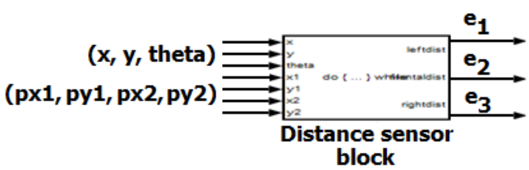

Fig. 5 The DS block

It has to be mentioned that in order to follow the design requirements some initial parameters of the localization module can be easily modified: the number of distance sensors, their placement (angles) on the robot, respectively the range of the sensors.

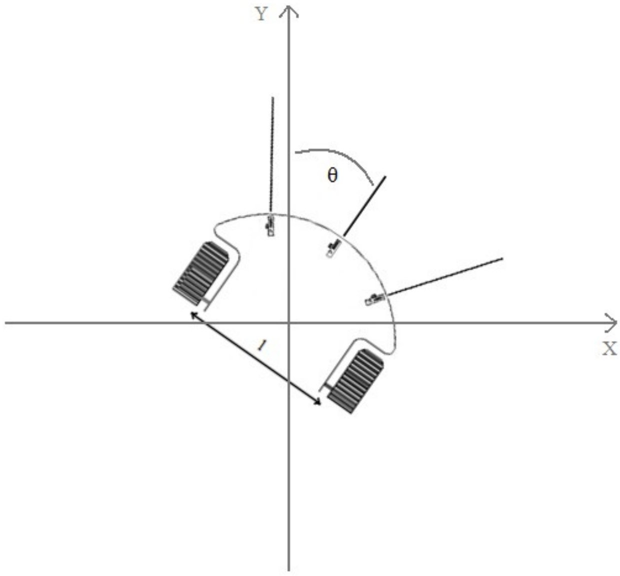

Fig. 6 The three distance sensors

All three outputs of the DS block are provided to the fuzzy controller.

3.2 Detailed Model of the DS Block

The internal structure of the distance sensor is shown in fig. 7 and it contains the following subsystems:

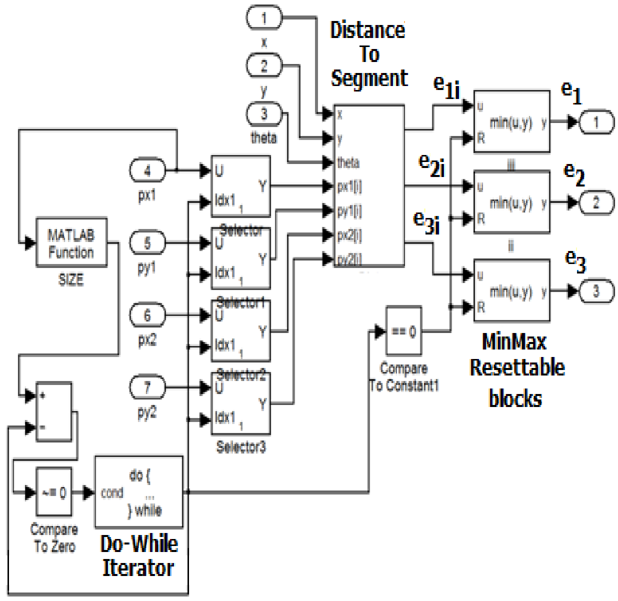

Fig. 7 The internal structure of the DS block

- Do-While Iterator block, which allows, through iterations, to take over all the segments considered being the obstacles. It assures a complete cycle of determining the correct distance from the robot to the obstacle by resetting the MinMax Resettable blocks.
- Four Selector blocks (Selector 0-3), used to extract the coordinates of the segments from the input arrays. The behavior of the Selector blocks is guided by the Do-While Iterate block.
- Three MinMax Resettable blocks, memorizing the minimum distance to the obstacles from all the computed distances at a given time for each of the three sensors. Therefore the inputs of these blocks are (e_{1i}, e_{2i}, e_{3i}), where $i = \overline{1;n}$, and the outputs are e_1, e_2 and e_3.
- DistanceToSegment block (DTS block), which computes the actual distances to the obstacles from the environment, provided through the distance sensors.

3.3 Detailed Model of the DTS Block

The DTS block assimilates in the current situation the three distance sensors, represented in fig. 8 through the Frontal sensor, Left side sensor and Right side sensor.

The inputs of these three blocks are: x, y, theta (previously defined) and (px1[i], py1[i], px2[i], py2[i]), the coordinates of only one segment. The outputs of these blocks represent the distances from the robot to the specified segment (e_{1i}, e_{2i}, e_{3i}). Also, nonlinear saturation blocks are considered in order to set a superior limit of 30 cm.

3.4 Detailed Model of the Distance Sensor

The internal structure of a distance sensor block (particularly the Frontal sensor) is shown in fig. 9.

The subsystems of a distance sensor block are:

- PointOfIntersection block, which computes the coordinates of the point of intersection between the line that symbolize the sensorial range and the line that represents the input segment. It outputs the coordinate of the intersection point (x_i, y_i).
- ConditionalInclusion block, which determines through logical operations that the point of intersection (x_i, y_i) is actually placed in the correct position related to the orientation of the robot. The inputs of this block besides the coordinates of the intersection point are: the coordinates of the segment (x_1, x_2), the actual position of the robot (x, y) and the angle theta. Therefore, if the ConditionalInclusion block validates the intersection point, then the distance from the robot to this point is provided as output, otherwise the output will be 30.

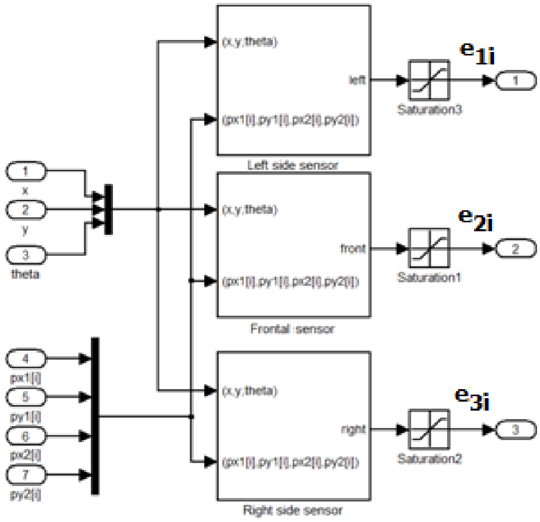

Fig. 8 The internal structure of the DTS block

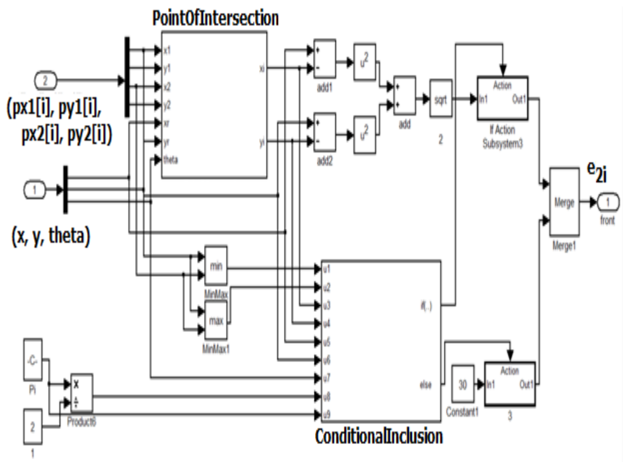

Fig. 9 The internal structure of a distance sensor

4 Control Strategy

Different control strategies for single WMRs are available in the literature [19], [20], depending on the required task, the ability to sense the environment, the type of environment, the structure of the robot, etc. An interesting assignment of control strategies for multi-robot navigation regarding the varying environment, required task and constraints is presented in [21].

This chapter presents first the fuzzy controller used to control the WMR in order to reach a destination and to avoid obstacles detected by the sensorial part. Then, the study focuses on cooperative robots that have to keep a certain formation. A leader-follower cooperative multi-robot control strategy is defined that assures the movement of cooperative robots in partially known environments. For the cooperative control strategy, a WMR has to be equipped with a wireless communication module.

4.1 The Fuzzy Controller

The fuzzy controller is designed to assure the movement of the WMR in a partially known environment. Particularly, the robot has to cover the distance from an initial point to the final destination, and to avoid obstacles. As it can be seen in fig. 10, the inputs of the fuzzy controller are: the distances to the obstacles (e_1, e_2 and e_3) provided by the left, front and right distance sensors, the distance that has to be covered to the final destination (Δd), and the angle correction (Δtheta) in order to obtain a correct orientation. The controller assures the correct speed at the left and right wheels (v_l and v_r). The actuators are considered ideal, non-inertial amplifiers with the gain value 1.

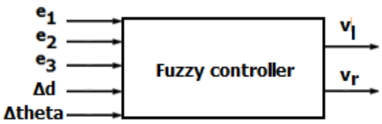

Fig. 10 The fuzzy controller

4.2 Rules and Membership Functions

The membership functions considered for the sensorial inputs, the distance Δd and the angle Δtheta are presented in figure 11a, 11b and 11c.

The controller outputs are considered to be the followings: zero, slow, medium and fast, represented through triangular membership functions.

In order to cover the distance from the initial point to the final destination the following rules are considered (the weight of the rules is 1):

IF (left is *far*) and (front is *far*) and (right is *far*) and (Δtheta is *negative*) and (Δd is *not zero*)
THEN (vr is *fast*) and (vl is *slow*)

IF (left is *far*) and (front is *far*) and (right is *far*) and (Δtheta is positive) and (Δd is *not zero*)
THEN (vr is *slow*) and (vl is *fast*)

IF (left is *far*) and (front is *far*) and (right is *far*) and (Δtheta is *zero*) and (Δd is *not zero*)
THEN (vr is *medium*) and (vl is *medium*)

IF (Δd is *zero*)
THEN (vr is *zero*) and (vl is *zero*)

In order to avoid obstacles the following principle is considered:

- if at some point the distance determined from one side (front and side sensor) is getting smaller (entering into the *medium* zone) then the robot will turn towards the other direction;
- if at some point the indication of one side sensor becomes *close* but the other two sensors are indicating *far* then the robot will slowly continue the forward movement;
- if at some point both side sensors are indicating *close* then the robot will turn guided by the value of Δtheta;
- if at some point one sensor indicates a value that enters into the *medium* zone and the side sensors are indicating *far* then the robot will slowly approach the object.

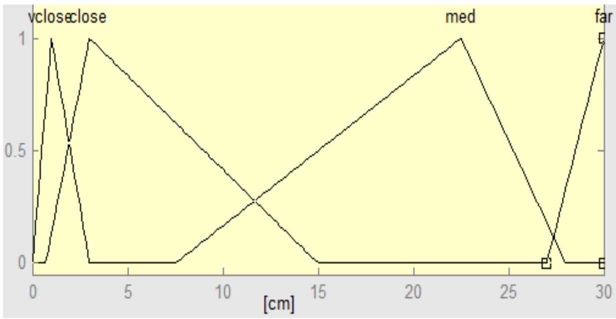

Fig. 11a The membership functions corresponding to the sensorial inputs

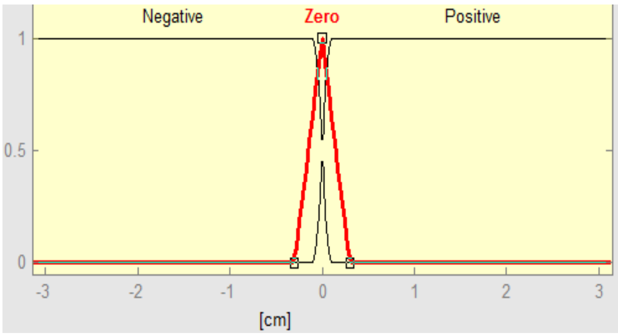

Fig. 11b The membership functions for the Δd distance

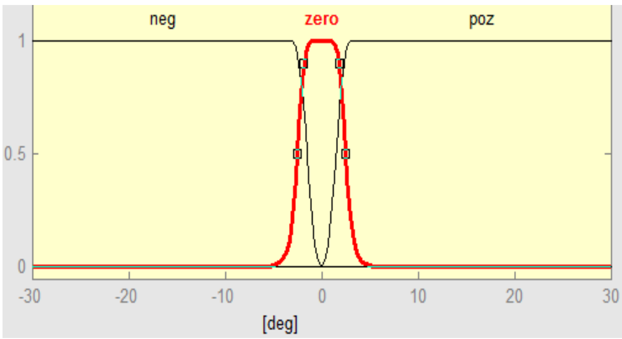

Fig. 11c The membership functions for the Δtheta angle correction

The inference is realized through the Max-Min inference procedure (Mamdani/Assilian), which is based on the T_M t-norm and S_M t-conorm. For defuzzyfication, the COG (center of gravity) method is applied.

4.3 Cooperative Multi-Robot Navigation

As mentioned in the introductory part, this research focuses on the cooperative robots that have to reach a target with a known position, moving in partially known environments. The cooperative robots must also keep a certain formation as long as possible. This approach is often used for door to door movements, when the goal is to move the robots in formation.

The cooperative multi-robot navigation relies on the following principles:

- a leader-follower robot behavior when there are no obstacles;
- the robots are moving in formation when it is possible;
- individual obstacle avoidance procedures;
- regrouping in the desired formation after individual avoidance of the obstacles.

The mentioned principles can be resumed into two scenarios, defined by the presence of obstacles. The first scenario is represented by the obstacle avoidance procedure and the second scenario occurs when the group of robots is going straight towards the target, no obstacle being present.

The first scenario is characterized by the followings:

- an obstacle is detected by the robots and the avoidance procedure began;
- each robot avoids the obstacle individually;
- the obstacle avoidance is realized using the fuzzy controller;
- obviously, the robots are not evolving in the desired formation.

The second scenario has the following attributes:

- no obstacle is detected, the robots are going straight toward the target;
- navigation in a formation is required;
- a leader is placed in front and it is guiding the movement of all the robots;
- the robots are following the leader using a second fuzzy controller;

In consequence, the follower robots are equipped with two fuzzy controllers, the first one (C_1) being used to follow the leader robot in a specific manner in order to maintain the formation, and the second one (C_2) for the individual obstacle avoidance.

The formation, in which the robots must evolve, was initially conceived for two robots (a leader and a follower) as illustrated in fig. 12. The follower, in the current application, is supposed to be situated at 50cm, and 30 degrees angle related to the leader robot.

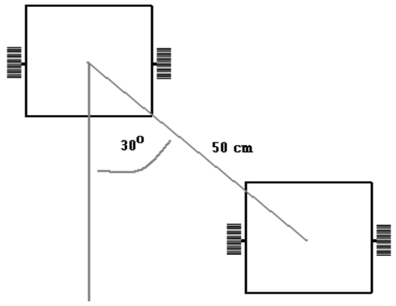

Fig. 12 The formation of two robots

The controllers are conceived in the same manner as described in the previous subchapter, and they are acting conditioned by the presence of obstacles. Considering that (x, y, θ_l) are the position and the orientation of the leader robot, the reference (x_{ref}, y_{ref}) for C_1 (used when no obstacles are present) will be permanently set as:

$$x_{ref} = 50 \cdot \sin\left(\theta_l + \frac{5 \cdot \pi}{6}\right) + x$$
$$y_{ref} = 50 \cdot \cos\left(\theta_l + \frac{5 \cdot \pi}{6}\right) + y \quad (5)$$

Therefore, considering that $\Delta\theta_f$ and $\Delta\theta_l$ are the orientation corrections for the follower, respectively the leader robot (provided by the P&D block), the basic logic that describes the switching procedure between C_1 and C_2 for the follower robot is:

- if $\Delta\theta_l \neq 0$ then C_2;
- if $\Delta\theta_f == 0$ then C_1;

5 Results

A Graphical User Interface (GUI) was conceived and implemented for a better usage of the model. The GUI is shown in fig. 13, respectively its state transition diagram is illustrated in fig. 14.

Three scenarios were imagined in order to illustrate the behavior of the modeled robot controlled by the fuzzy controller. The initial position of the robot is represented by the coordinates (0, 0) and the 0 degrees angle (meaning that it sees the positive side of the vertical axe). In the first scenario, presented in fig. 15, six obstacles were considered and the robot had to reach the final destination with the coordinates (-350, 100). The second and the third scenarios, presented in fig. 16 and fig. 17, are considering two, respectively three obstacles, and the destination is expressed by the coordinates (300, 200), respectively (300, -200).

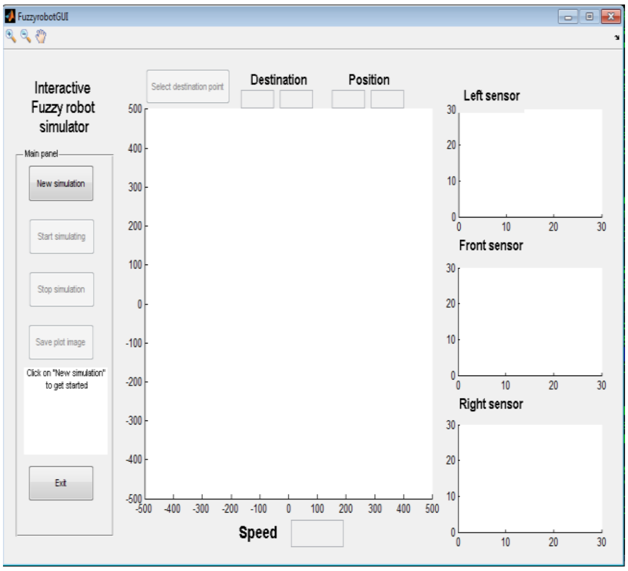

Fig. 13 The GUI

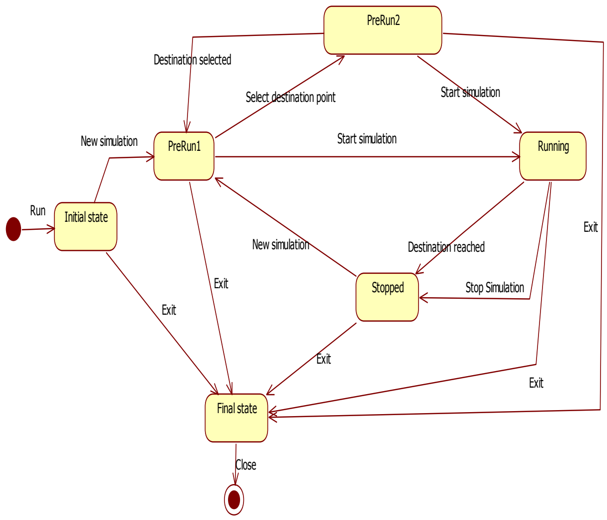

Fig. 14 The state transition diagram of the GUI

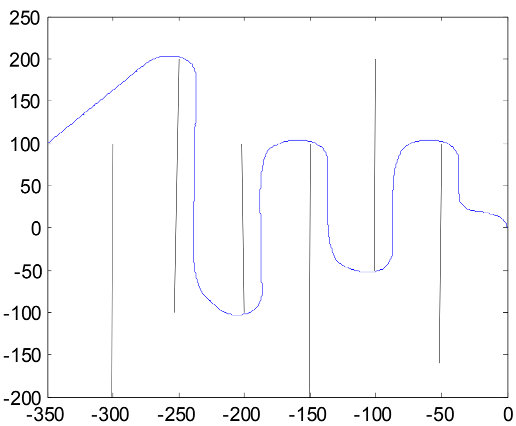

Fig. 15 The first experimental scenario for a single robot

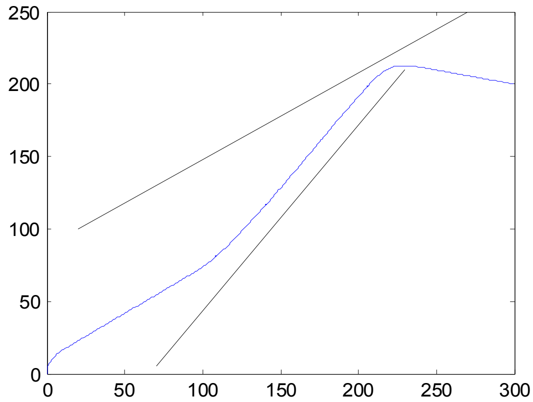

Fig. 16 The second experimental scenario for a single robot

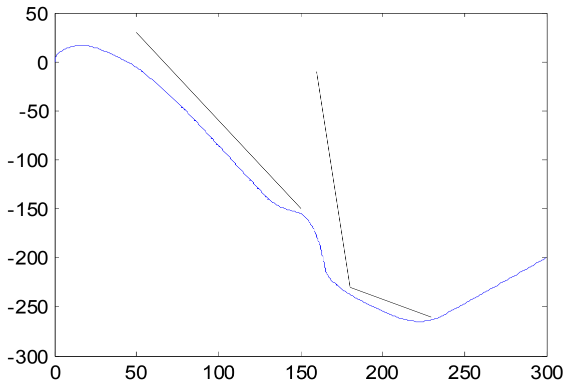

Fig. 17 The third experimental scenario for a single robot

As it can be observed in figures 15, 16, and 17, the modelled WMR (its evolution shown by the blue line) is able to move in a partially known environment and to avoid obstacles.

Other three scenarios were imagined in order to show the behaviour of two collaborative robots in an environment as described above. The initial position of the leader robot is represented by the coordinates (0, 0) and the 0 degrees angle.

The evolution of the leader robot is shown by the blue line and the movement of the follower robot is described by the dashed red line. In the first scenario, presented in fig. 18, no obstacles were considered and the leader robot had to reach the final destination with coordinates (-300, 400). The second and the third scenarios, presented in fig. 19 and fig. 20, are considering two obstacles, and the destination is expressed by the coordinates (400, 200), respectively (400, -200).

From figures 18, 19, and 20, it can be concluded that the cooperative robots are able to reach the destination using the mentioned control strategy.

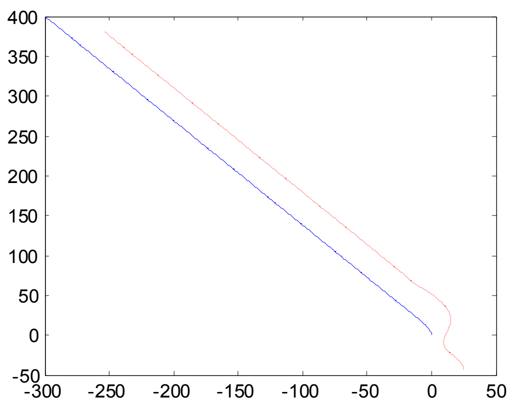

Fig. 18 The first experimental scenario for cooperative robots

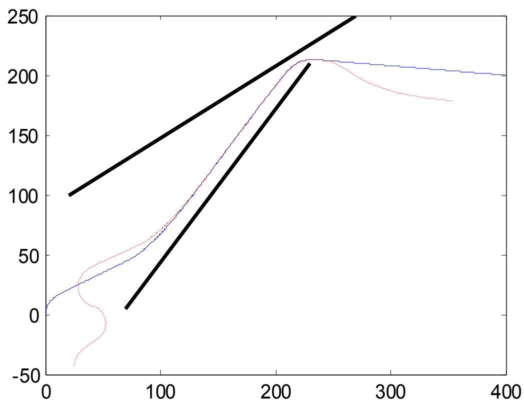

Fig. 19 The second experimental scenario for cooperative robots

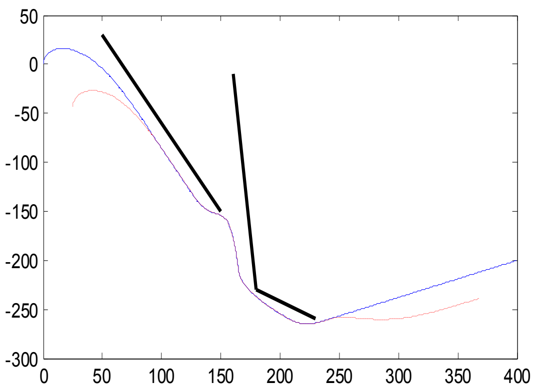

Fig. 20 The third experimental scenario for cooperative robots

6 Conclusions and Development Directions

A mathematical model of a WMR was developed and implemented in Matlab/Simulink environment. The WMR model consisted in the kinematic part and also the sensorial part used to avoid obstacles in partially known environments. A fuzzy controller was conceived to command the robots speed.

A first step in controlling the behaviour of a group of WMRs was realized in this study. A leader-follower strategy was implemented in order to keep the robots in a certain formation while moving towards a destination point in a partially known environment. Two fuzzy controllers were implemented for the follower robot, one being used for the obstacle avoidance and the other for the leader following procedure.

The current work will be followed by several development directions. The purpose is to investigate and implement search and rescue robots behaviour. A general control strategy considering the transmission problems that may occur in the case of a cooperative behaviour with robots evolving in a certain formation will be developed. Also, the final steps for the search and rescue robots are the cooperative target search procedures. These will be conceived for known as well as for unknown target position.

Acknowledgement

This work was partially supported by the strategic grant POSDRU 21/1.5/G/13798, inside POSDRU Romania 2007-2013, co-financed by the European Social Fund – Investing in People.

References:

[1]Ray L., Joslin J., Murphy J., Barlow J., Brande D., Balkcom D. (2006). Dynamic Mobile Robots for Emergency Surveillance and Situational Awareness, *International Workshop on Safety, Security, and Rescue Robotics*, August 22-24, Maryland, USA.

[2]Urcola P., Riazuelo L., L'azaro M. T., Montano L. (2008). Cooperative navigation using Environment compliant robot formations, *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, September 22-26, Nice, France, pp. 2789-2794.

[3] MacArthur E. Z., Crane C. D. (2007). Compliant formation control of a multi-vehicle system, *Proceedings of the IEEE International Symposium on Computational Intelligence in Robotics and Automation*, pp. 479–484.

[4] Bazoula A., Maaref H. (2007). Fuzzy Separation Bearing Control for Mobile Robots Formation, *World Academy of Science, Engineering and Technology*, pp. 1-6.

[5]Wachter L., Murphy J., Ray L., (2008). Potential function control for multiple high-speed nonholonomic robots, *Proceedings of the IEEE International Conference on Robotics and Automation ICRA*, May 19-23, Pasadena, USA, pp. 1781 - 1782.

[6] Traub M., Kaminka G., Agmon N (2011). Who goes there? Selecting a robot to reach a goal, *Proceedings of the 10th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, May 2-6, Taipei, Taiwan, pp. 91-98.

[7] Marcolino L. S., Chaimowicz L. (2009). Traffic Control for a Swarm of Robots: Avoiding Target Congestion, *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems IROS*, October 10-15, St. Louis, MO, USA, pp. 1955 – 1961.

[8] Ballesta M., Gil A., Reinoso O., Julia M., Jimenez L. M. (2010). Multi-robot map alignment in visual SLAM, WSEAS Transactions on Systems, Issue 2, Vol. 9, pp. 213-222

[9] Ballesta M., Gil A., Reinoso O., Paya L., Jimenez L. M. (2010). Map Fusion in an Independent Multi-Robot Approach, WSEAS Transactions on Systems, Issue 9, Vol. 9, pp. 959-968

[10]Pradhan, S. K., Parhi, D. R., Panda, A. K. (2007). Navigation of Multiple Mobile Robots using Rule-based-Neuro-Fuzzy Technique, *International Journal of Computational Intelligence*, pp. 142-152.

[11] Kantor G., Singh S., Peterson R., Rus D., Das A., Kumar V., Pereira G., Spletzer G. (2003). Distributed search and rescue with robot and sensor teams, *Field and Service Robotics*, pp. 327–332.

[12] Burgard W., Moors M., Fox D., Simmons R., Thrun S. (2000). Collaborative Multi-Robot Exploration, *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, April 24-28, San Francisco, CA, USA vol. 1, pp. 476 – 481

[13] Rooker M. N., Birk A. (2007). Multi-robot exploration under the constraints of wireless networking, *Journal of Control Engineering Practice*, vol. 15, no. 4, pp. 435-445.

[14] Weiyan S., Yang Q. F., Zhengshuhua C. (2011). Structure Design and Performance Analysis for Locomotion System of the Field Exploration Robot, WSEAS Transactions on Systems, Issue 10, Vol. 10, pp. 331-341

[15] Neri F. (2011). Learning and Predicting Financial Time Series by Combining Evolutionary Computation and Agent Simulation, *Transactions on Computational Collective Intelligence*, Vol. 6, Springer, Heidelberg, Vol.7, pp. 202-221.

[16] Nitulescu, M. (2007). Solutions for modeling and control in mobile robotics, *Journal of Control Engineering and Applied Informatics CEAI*, vol. 9 (3;4), pp. 43-50.

[17] Nitulescu M. (2008). Theoretical Aspects in Wheeled Mobile Robot Control, *Proceedings of the IEEE International Conference on Automation, Quality and Testing, Robotics*, vol. 2, pp. 331-336.

[18] Korodi A., Dragomir, T.L. (2010). Correcting Odometry Errors for Mobile Robots Using Image Processing, *Proceedings of the IAENG International Conference on Control and Automation*, 18-20 March, Hong Kong, vol. 2, pp. 1040-1045.

[19] Halal, F. and Dumitrache, I. (2006). Genetic algorithm in mobile robot control, *Journal of Control Engineering and Applied Informatics CEAI*, vol. 8 (2), pp. 21-30.

[20] Pacheco, L., Ferrer, J., and Luo, N. (2008). Local model predictive control experiences with differential driven wheeled mobile robots, *Journal of Control Engineering and Applied Informatics CEAI*, vol. 10 (2), pp. 59-67.

[21] Issa, F. and Dumitrache, I. (2009). Optimization of Control Strategy Assignment. A Framework Structure for Multiple Mobile Robots, *Journal of Control Engineering and Applied Informatics CEAI*, vol. 11 (4), pp.36-44.