

Cooperative robot team navigation strategies based on an environment model

P. Urcola and L. Montano

Instituto de Investigación en Ingeniería de Aragón, Universidad de Zaragoza (Spain)

Email: {urcola, montano}@unizar.es

Abstract— This paper addresses a flexible cooperative navigation system for small robot teams. The members of these groups move in formation, but with enough flexibility to be able to avoid obstacles and adapt to the environment shape. This approach is based on an analysis of the environment on a grid-based model that permits each robot to continuously evaluate the situation in which it is, in order to decide the best of the three developed navigation strategies, one for each situation type. Moreover, by using this model the robot cooperation for a coordinate navigation is improved, allowing each robot to share the observations with the other robots. So the field of view of each robot of the team on which it makes the decisions is enlarged. The system is implemented in a decentralized way so the number of robots is not a main issue, thus being scalable. Eventually, initial formation topology can be modified, in order to comply with the environment while the mission is accomplished. The system and the strategies are evaluated by means of simulations, showing the robustness against possible blocking situations.

I. INTRODUCTION

Actual robots are able to perform very complex tasks on their own. But some of these tasks, furthermore in very large scenarios, can be enhanced using groups of robots. Exploration of unknown environments, surveillance, mapping and many other missions can be improved by means of cooperating robots.

Concretely, robots inside a formation navigate keeping the relative localization with each other while they achieve their tasks. This navigation strategy is able to cope with communication constraints and also to reduce the path planning computation costs because all the members inside a group navigate using close trajectories.

The main contribution of this paper is a cooperative navigation system for small robot teams. Flexible formation techniques are applied to adapt the configuration of the group to the environment. Robots continuously evaluate the environment to select the best navigation strategy defined for three possible situations. Furthermore, single robot perception is improved with other robots observations. The size of the team is not, at first, a constraint because of a decentralized implementation.

Next section provides a summary of the background work related to the techniques proposed here. In section III, an overview of the system is shown and then, in sections IV, V and VI the core of the navigation strategies is detailed.

This work was partially supported by the Spanish project DPI2006-07928, and the European project IST-1-045062

Sections VII and VIII present the results gathered from the different simulations and the conclusions obtained.

II. RELATED WORK

Many papers have been published focused on cooperative control for the robots like in [1]. Some of them apply physics analogies to the motion controller, in which the stability analysis using Lyapunov functions [2] becomes easier. Concretely, potential fields have been widely used to model the influences among robots and to define the control laws for the formation maintenance. As a derived technique from potential fields, virtual spring-damper systems have recently been proposed because motion controllers using this kind of approach let the robots perform smooth trajectories while the formation structure is maintained with flexibility. Proposals in [3] and [4] use this kind of methods to keep the formation shape.

The flexibility of the formations that use spring-damper systems as links can be used to adapt the formation topology to the shape and constraints of the surrounding environment. However, robots need the capabilities to sense the scenario and to avoid the obstacles perceived by the sensors. Some navigation methods for formations to navigate in a scenario with obstacles have been proposed. One typical solution, as in [5] and [6], is to change the interconnections between robots to adapt the shape of the formation. But the problem of deciding the best robot interconnection graph is not easy to solve for the general case. Another kind of approach proposed in [7], is to compute a configuration space for the whole formation, considering the team of robots as one entity. This solution presents a lack of flexibility in the shape of the group.

Formation control based on potential fields leads to an easy integration with obstacle avoidance methods based also on potential fields. This is the main issue in the proposal in [8] and [9]. However, the local minima problem is still present on these navigation techniques, and there is a need of a more complex system to avoid realistic obstacles with U-trap shapes.

It is also important for the formation movement that the controller provides a smooth trajectory for all the components of the group. Furthermore, kinematics and dynamics of the robots impose constraints that can provoke abrupt and sudden movements, if they are not taken into account. In [10], the authors propose a multi-robot navigation function

that considers all the robots movement restrictions at the same time. However, this centralization is a limit for the number of robots in the formation and also can be a problem in the application in realistic scenarios, where wireless communications could not provide such a good capacity.

In [11], a navigation system for robot formations with obstacle avoidance and path planning capabilities that takes in consideration kinematic and dynamic movement restrictions is proposed. However, the obstacle avoidance technique provided presents some problems because it does not bound the influence of the environment in the formation, which may lead to unstable behaviors, mainly in complex scenarios. Another issue is related to the reactivity of the method, which is not able to cope with hard obstacles such as U-traps. The solution to these problems is the main contribution of this paper.

III. SYSTEM OVERVIEW

The navigation system for a formation of robots proposed here uses a leader-followers approach. Planning and tracking a safe trajectory to the goal is in charge of one robot in the team, which is the *leader*. The rest of the members of the group, the *followers*, do not have a direct information about the goal, but just follow the leader under certain relative localization constraints. These restrictions are imposed to set the desired formation topology by specifying for each follower the required relative distance and angle to the leader.

In its basic behavior, the proposed system uses a physical analogy based on a spring-damper model described in [11]. All the robot movements are produced by virtual forces that are transformed into direct velocity commands by means of a dynamic and kinematic model-based controller, as published in [12].

The navigation strategies for the leader and for the followers are different, because the leader is charged on driving the formation to the goal, and the followers are driven by the leader, although adapting the formation topology and movements to the environment. Next sections are devoted to describe the navigation strategies for the leader and for the followers.

IV. LEADER NAVIGATION STRATEGY

The navigation techniques applied by the leader are mainly those used in a hybrid sensor based single robot navigation system, like [13]. A D* path planner computes a trajectory to the goal and the robot tracks it using an obstacle avoidance method (*ND*, the Nearness Diagram [14]).

But because of the fact that the leader is inside a formation, this movement is influenced by the movement of the followers, in order to keep the constraints mentioned above. To integrate the formation movement and the path tracking to the goal we have two elements:

- The force \mathbf{F}_{ND} that generates the movement towards the goal, which is computed from the output of the obstacle avoidance method and presents two main characteristics:
 - The direction is safe. It should not be changed.

- A force applied in that direction cannot be higher than \mathbf{F}_{ND} , in other words, the robot will acquire the maximum speed allowed for safety.

- The force \mathbf{F} the formation induces on the leader, which is the sum of the forces computed from all the followers.

In order to take advantages of the characteristics of the \mathbf{F}_{ND} force, the projection of the followers force into the goal movement one is computed.

$$\mathbf{P} = (\mathbf{F} \cdot \mathbf{u}_{ND}) \cdot \mathbf{u}_{ND} \quad (1)$$

where \mathbf{u}_{ND} is the unit vector in the *ND* motion direction. Three different situations may appear at this point, shown in fig. 1. If \mathbf{P} points to the same direction as \mathbf{F}_{ND} , see fig. 1(a), then followers are forcing the leader to move forward and then, the leader apply the \mathbf{F}_{ND} force towards the goal. When \mathbf{P} and \mathbf{F}_{ND} forces do not point in the same direction, as shows fig. 1(b), then the leader moves using the resulting force $\mathbf{F}_{ND} + \mathbf{P}$. There is one exception, depicted in fig. 1(c). In case the followers force is bigger than the goal force, to avoid backward movements, the leader just stops and waits for the followers to come closer.

V. FOLLOWER ROBOTS NAVIGATION STRATEGIES

The main idea is that, in spite of computing a trajectory to reach the goal while accomplishing the formation constraints, followers can use the movement of the leader. Our proposal is to decide each time which is the best follower movement strategy to apply in order to keep the formation. A complete situation analysis is done and the best navigation technique is used in each case. This analysis is performed using the data needed for obstacle avoidance, so the computational cost is very limited.

In the literature there are many navigation strategies focused on different aspects in the robot motion such us safety, smoothness, kinematics or dynamics. Each one presents advantages and limits that require a very careful selection in each possible situation in which the robots can be at any time (navigation with no obstacles, simple scenarios, complex scenarios, ...).

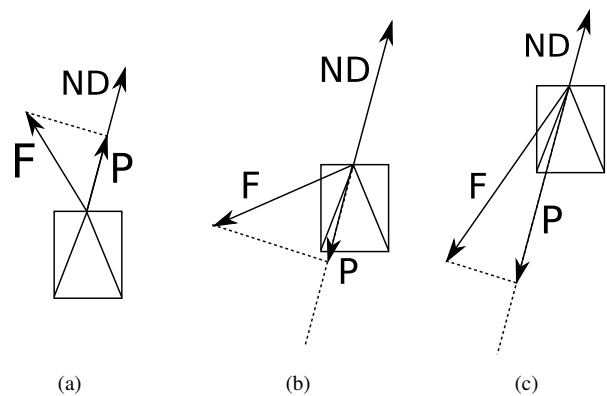


Fig. 1. Forces induced on the leader of the formation. In figure (a), the leader is pushed towards the goal. In (b), the leader slows down because of the pulling force from the followers. In (c), followers force can stop the leader movement.

In this system, a set of three strategies have been decided to be available for the follower robots. Depending on the relative environment-formation situation, one of them is applied. Lets firstly explain the navigation strategies:

- Cooperative formation control (*SD*)
- Obstacle avoidance method integrated inside the cooperative control (*OA*)
- Path planning and obstacle avoidance for individual movement (*PP*)

All these techniques present a compromise between smoothness and safety in the movement, which let choose the correct one in each situation.

A. Spring-Damper System Formation Control

The formation control presented here is based on the leader-followers approach using a spring-damper system published in [11]. Each follower is virtually linked to the leader to propagate the movement properly, as shown in fig. 2. These propagation links behave like a physical system with linear and rotational springs and dampers that allows the robot to move coordinately but with no so rigid constraints. This technique is labeled as *SD* strategy.

In spite of the fact that this model gives a general smoothness and keeps the relative localization among the robots inside the formation, there are some unacceptable behaviors when it is applied to non-holonomic real robots. There are speed and acceleration limits and backwards movement is not allowed because of the lack of rear perception. To avoid these bad situations, some rules are applied:

- All robots have limited speed and acceleration.
- Robots, because of the lack of perception sensors, can not move backwards.

These rules let the springs have asymmetric behaviors, emphasizing the difference between the leader and the followers. With this strategy, formation movement becomes really

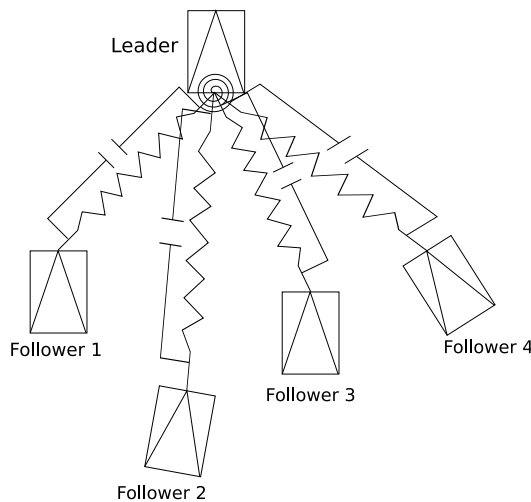


Fig. 2. Five robots in formation with virtual links between the leader and the followers. Each link has a linear and torsion spring and a linear and torsion damper.

smooth and coordinated. It is also very light in computational costs and easily decentralized. Main drawbacks of the approach are the problems of unbounded behaviors, and the parameter estimation and tuning, mainly to cope with constraints imposed by the environment. To solve some of this problems, an hybrid technique combining the spring-damper model and an obstacle avoidance method is proposed.

B. Formation Control and Obstacle Avoidance Integration

A reactive method based on potential fields generated by the obstacles may appear easy to integrate with the formation control proposed above. But an integration based on simple potential addition leads to unbounded behaviors that cannot be controlled. For this reason, we propose to apply the Nearness Diagram technique from [14] to provide obstacle avoidance capabilities. This technique is labeled as *OA* strategy.

The output of this method is fused with the formation control. To do this, the virtual force that applied would generate the obstacle avoidance behavior is computed and then integrated in the formation control. This integration is not just a force addition. From the Nearness Diagram, a safe direction and a maximum speed is obtained. Then, the magnitude of this force is modulated using the spring-damper system to adequate it to the movement of the other robots. This process is equal as the one presented in IV for the leader with the difference that the force induced by the formation in each follower comes from the link to the leader and that the goal for the followers is the localization of the leader.

With this navigation strategy we obtain a safe movement with a deviation from the desired relative position to avoid th eobstacle which leads to a change in the shape of the group. However, the problem of the local minima can appear, which is a very well known problem in reactive navigation systems. A path planner is needed to solve these situations.

C. Path Planning and Obstacle Avoidance

A D* path planning algorithm is used to solve the local minima problems in the follower robots navigation. It is integrated with the obstacle avoidance system, using the techniques developed in [13]. This technique is labeled as *PP* strategy. No previous map is assumed and only a robocentric path planning is considered. However, formation members can provide the others extra perception information which can be very useful for planning. To model the environment, we use a binary occupancy grid whose cells are updated whenever a new sensory measurement is available. The grid has a limited size, it is centered on the robot reference, and it is continuously recomputed to include the robot location. By means of every robot observation sharing, reflected in the model, the perception of each individual robot is improved, and so the paths planned. This way, a true cooperative mapping and planning technique for the robots in the formation is used.

To make all the maps consistent with each other, robots send to the others the actual observations. Then, local grids

are updated mapping this information directly. As data received are direct observations, they replace the local values stored in the grid. One problem appears when robots are perceiving each other, because they consider themselves as obstacles and this information is shared. To avoid such a problem, a zone around each robot is in charge of that robot only, and it cannot be modified by other information. In fig. 3 and in the video [15], the local grid of the leader is shown. It can be observed that other robots are contributing to this local map also.

This strategy is able to find a safe path to any reachable location in the scenario. However, the topology and the coordination in the movement between the robots can be dramatically reduced.

The following section describes the technique used to analyze the environment in order to select the corresponding navigation strategy.

VI. ENVIRONMENT ANALYSIS

From the advantages and drawbacks of the navigation strategies proposed above, an analysis of the environment has to take place to decide which one is the best in every situation. This decision takes into account the limitation that each strategy has. Concretely, direct formation control strategy (*SD*) is only functional in obstacle free conditions, *OA* strategy is useful for follower obstacle avoidance, and path planning strategy (*PP*) is required to cope with local minima situation such as U-traps.

A. Environment Diagram

To detect these situations, the environment analysis is based on the Nearness Diagram provided by the *ND* obstacle avoidance method. It also uses the grid based environment representation mentioned in section V-C, in which the obstacles perceived by the robot range-finder sensors are mapped.

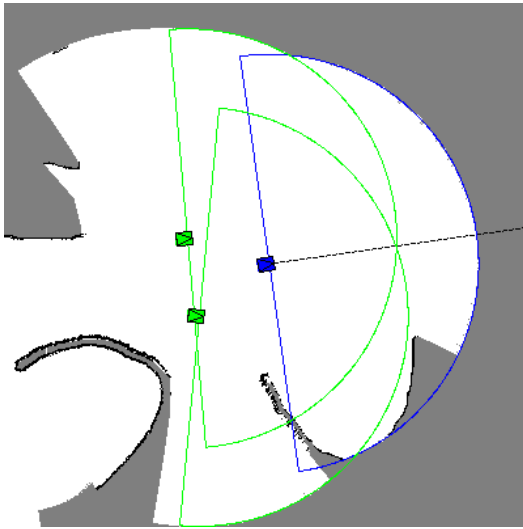


Fig. 3. Leader (the robot that is planning) view of the environment, augmented with other robots information. The actual perception range is shown to emphasize that the leader is using information from the followers perception.

The proposed method has two advantages: first, the Nearness Diagram processing made by the *ND* navigation is well adapted to identify the situations that allow to choose the navigation strategy; second, profiting this processing no great additional computations are needed.

We will name *ED* to this Environment Diagram. The *ED* reference frame *R* is robocentric, its *y* axis is aligned to the robot-goal direction, the *x* coordinate is discrete (5cm in our implementation), and the *y* coordinate represents the distance to the obstacles for every *x* value. Fig. 4 represents this diagram on a given robot reference.

Definition 1: Let *P* be the set of all the points perceived by the sensors and represented in the *ED* reference frame *R* with the *x* coordinate discrete.

Definition 2: Let *Q* be a subset of *P* with all the points that are between the robot and the goal.

$$Q = \{(x_i, y_i) | (x_i, y_i) \in P \wedge 0 < y_i < y_{goal}\} \quad (2)$$

Definition 3: A function *d* is defined to compute the minimum distance from the *x* axis to a point in *Q* in each interval of the diagram, i.e., for every *x_i*.

$$d(x_i) = \min_{(x_i, y_i) \in Q} y_i \quad (3)$$

Using all the definitions above, the value of the diagram *ED* for each value *x_i* is computed as follows:

$$ED(x_i) = \begin{cases} d(x_i) & \text{if } \exists y_i : (x_i, y_i) \in Q \\ 0 & \text{if } \nexists y_i : (x_i, y_i) \in Q \end{cases} \quad (4)$$

Two kinds of regions may appear in the diagram:

Definition 4: A *free region* is a pair (*i, j*) so that *i, j* ∈ ℤ and *i* ≤ *j*, such that *ED*(*x_k*) = 0, ∀ *k* ∈ [*i, j*].

Definition 5: A *non-free region* is a pair (*i, j*) so that *i, j* ∈ ℤ and *i* ≤ *j*, such that *ED*(*x_k*) ≠ 0, ∀ *k* ∈ [*i, j*].

Small free regions where the robot can not navigate are filled with virtual points interpolating the surrounding non-free regions.

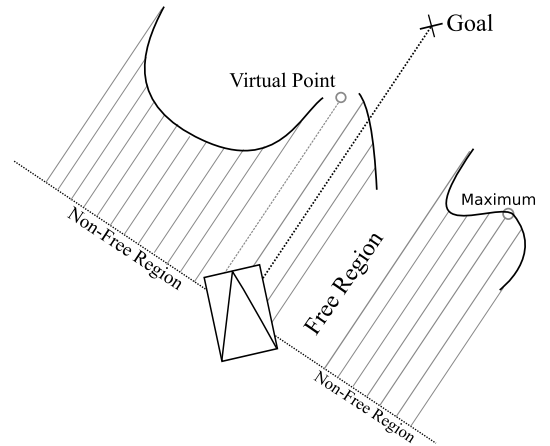


Fig. 4. Example of an *ED* diagram. The robot find different regions in his way to the goal. A virtual point has been added in a free region that is too small to navigate.

Three different cases are defined depending on the environment conditions:

- 1) *Free* situation, when the robot can navigate towards the goal directly with no obstacles in the surroundings.
- 2) *Avoidance* situation, when there are some obstacle that affects the robot movement.
- 3) *Trap* situation, when the obstacles between the robot and the goal have probably local minima zones.

Several criteria are used to determine the actual situation using the decision tree shown in fig. 5.

- The first criterion is the kind of region where the goal is placed: a *free region* or a *non-free region*.
- The second criterion determines if the region is wide enough to let the robot navigate with no care about obstacles. The minimum region width is defined as $w_{min} = w_{robot} + 2 \times d_{safety}$ where w_{robot} is the width of the robot and d_{safety} is the distance fixed to start avoiding an obstacle.
- The third criterion checks whether the region has, at least, one local maximum inside, which corresponds directly to a potential minimum.
- The fourth criterion computes the size of the local maximum, height and width, to check whether the robot is able to reach a potential local minima inside it or not. This may happen if the robot fits the concavity defined in the neighborhood of the extreme point.

As an output of this decision tree, one of the three different defined situations is obtained.

B. Strategy selection

Each one of the situations defined in the analysis phase maps directly to one of the navigation strategies mentioned above. Concretely, if a *Free* situation is found, then there is no collision danger and a direct formation control (*SD*) can be applied to navigate.

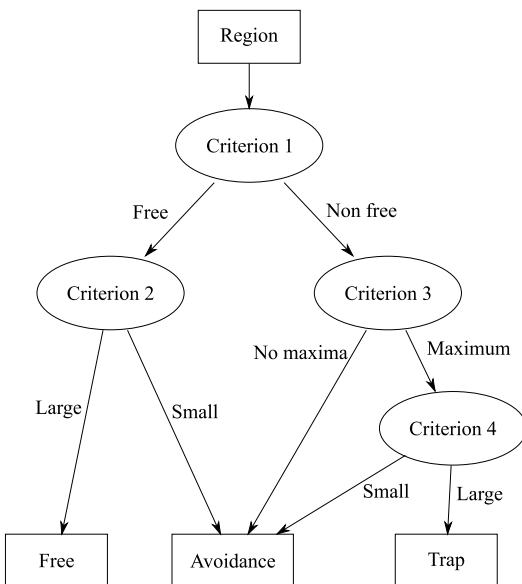


Fig. 5. Decision tree for the environment analysis

If the analysis determines that the situation is *Avoidance*, then the proximity of the obstacles requires an obstacle avoidance method (*OA*) for the correspondent followers. Finally, if the situation is considered as *Trap*, then there is a risk of a local minimum problem and then, a path planning method combined with obstacle avoidance (*PP*) is required to navigate. These three situations are shown on fig. 6, where each follower is using a different strategy while the leader moves towards the goal.

Strategy Switching

Once a strategy is assumed, the robot is able to follow it without any discontinuity in the movement. But there is a need to make the transitions between the different strategies as much smooth as can be. In fig. 7, an state machine with the transitions between the different strategies adopted.

The main consideration in the transition between *SD* and *OA* strategies is that the linear and angular velocities of the robots should not suffer discontinuities. To cope with that, all the parametrization is made so that in the boundary, when the robot is at the security distance from an obstacle, the behaviour of both strategies is similar.

The transitions with *PP* are more complex. Using the fact that the robot is planning a safe path to the goal, it is easy to make a previous analysis of the sub-goals in the path to find those points where the robot can take back any other strategy. It is precisely the first point in the path that presents this characteristic the one selected to be the end point of the path planning strategy. While the robot is tracking this path, no other strategy can be adopted.

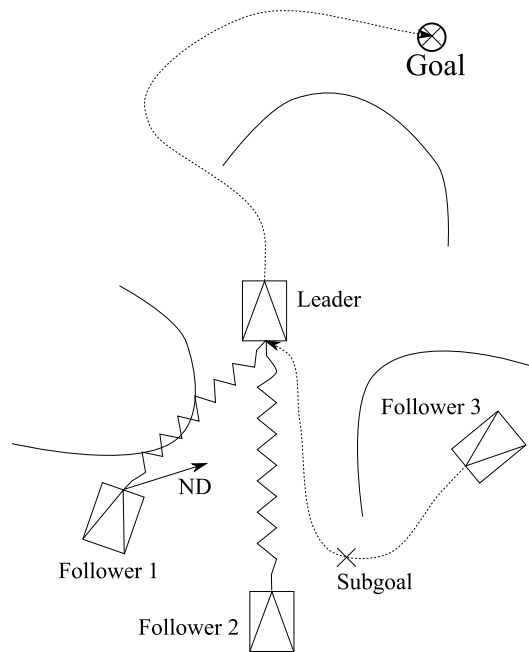


Fig. 6. Different strategies in use. Follower 1 is using *OA* strategy, follower 2 is using *SD* and follower 3 is using *PP*. The 'x' is the first point in the follower 3 path where *PP* strategy can be switched off safely.

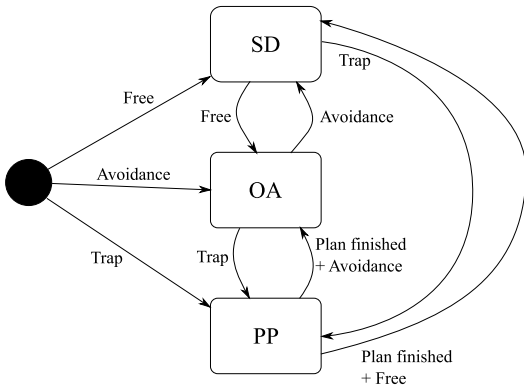


Fig. 7. State machine that defines the transitions between the different navigation strategies.

VII. SIMULATIONS AND RESULTS

The simulations presented in this section have been planned to show the main achievements of the system proposed above. For the sake of clarity in the pictures and the video, only a formation composed by three robots is shown. As mentioned above, the number of robots is not a constraint and then, we focus the analysis on the different situations the environment may present. The scenario in which the simulations have been executed shows several U-shaped obstacles, in which a classical formation control based only on reactive techniques would drive to block the motion of one or more followers, and thus to the formation.

In order to define the experiments, the following points are considered:

- The three strategies proposed for different situations are able to cope with any situation that may happen during navigation.
- The communication between robots and the shared information enhance the perception of every single robot.
- The whole navigation is able to navigate to the goal using smooth trajectories, even when transitions between strategies appear.

The experiments have been done in Player/Stage platform in which the kinematics, dynamics and sensor capabilities of the real robots in the laboratory, three Pioneers 3-AT with laser-range finding sensors, can be successfully simulated in a virtual world. For communications among robots, a real time wireless multihop protocol with multicast capabilities [16] has been used. Although they are simulations, the implementation is the same that the one used on real robots, so that the next step is easier to do.

In fig. 8, a trajectory tracked by the each robot in the formation is shown. It is depicted how the shape of the formation is adapted to go through narrow zones. No blocking motions are produced due to potential trap situations that could appear without planning strategies in complex scenarios as the one presented. The video attached to the paper [15] is the execution of this trajectory.

To give an idea of the dynamics of the simulation, in fig. 9 linear speeds of all the robots are represented. It is

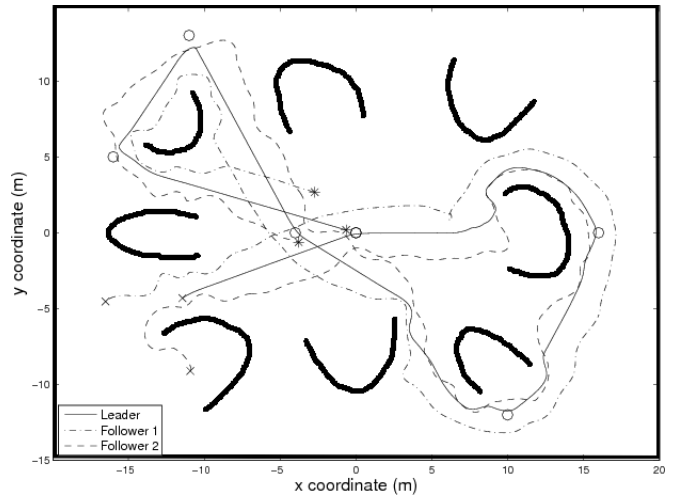
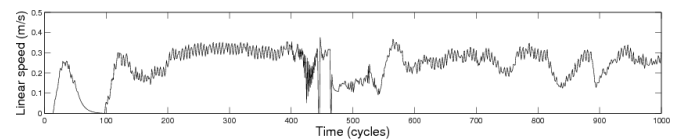


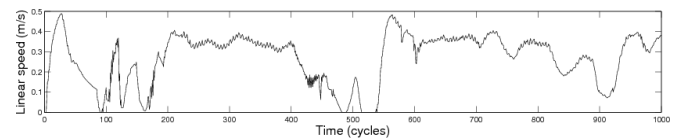
Fig. 8. Formation trajectory in a test scenario. They start from 'x' localizations and navigate through the waypoints marked as circles towards the goal(*).

clear the relationship existing among speeds of robots, due to the virtual links in the formation. In addition, there are no discontinuities in the graphics that could be associated with transitions between navigation strategies.

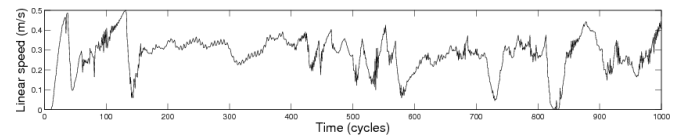
In TABLE I, there are some statistics based on the usage of the navigation strategies by the followers. The test performed are six different paths composed by ten goals computed randomly in the scenario. In each cell it is recorded the number of times each strategy has been used during the simulation by any follower. It is clear, according to the results, that most of the situations where the formation can be found can be solved without the path planning strategy. Although the scenario presents many potential U traps for the robots, obstacle avoidance strategy prevents the followers



(a) Formation leader linear speed profile.



(b) Follower 1 linear speed profile.



(c) Follower 2 linear speed profile.

Fig. 9. These graphics shows how robot velocities are interrelated by means of the spring-damper systems. At the beginning, while follower 2 is avoiding an U-trap obstacle, the other robots slow down, even stop, to wait for him.

Test	SD	OA	PP	Total
Random 1	3305	1031	4	4340
Random 2	5225	2001	26	7252
Random 3	5579	2720	65	8364
Random 4	4736	1556	2	6294
Random 5	5543	2888	5	8436
Random 6	6490	3685	29	10204
Video	2717	1405	82	4204
Total	33595	15286	213	49094
Rate	68.43%	31.14%	0.43%	100%

TABLE I
NAVIGATION STRATEGY STATISTICS

to enter in these traps. The *PP* strategy is only needed in few situations. It is computationally expensive, so it is not justified to use it for all the robots and all the situations as an alternative to the other strategies. The records obtained in the test shown in the video attached and in fig. 8 presents a little higher use of *PP* strategy but just because the starting position has been forced to show this capability.

VIII. CONCLUSIONS

A cooperative navigation system for a small team of robots in unknown scenarios has been proposed. The system applies three navigation strategies, each associated to a situation defined by three defined robot-environment relationships. In order to identify these situations, a grid-based model representing the perceived obstacles and a technique to check on the model the kind of obstacles the robot observes, have been developed. The environment information is shared by all the robots, which cooperatively enlarge their field of view, improving the planned motion. The system is implemented in a decentralized way and the robots are related in such a way that the number of robots is not a main issue. It is only considered by the leader navigation algorithm and for communication issues which, thanks to the real time protocol, are time bounded.

The strategies have been evaluated by means of simulations in a scenario with obstacles that could lead the formation to be blocked. However, the system correctly identifies the different situations choosing the corresponding navigation strategy, smoothly switching between them and leading the robots towards the goal without stopping or breaking the formation.

Some improvements can be taken into account to make the system more adaptable to the environment. When the formation is forced to suddenly change the direction in a stretch zone, the maneuver can be very difficult and some undesired behaviours can appear. If this situation is detected,

some specific strategies can be applied to accomplish the task in a better way, for example, changing the configuration of the formation or even the roles of the robots.

REFERENCES

- [1] Tove Gustavi and Xiaoming Hu. Observer-Based Leader-Following Formation Control Using Onboard Sensor Information. *IEEE Transactions on Robotics*, 24(6):1457–1462, December 2008.
- [2] Herbert G. Tanner and Amit Kumar. Formation stabilization of multiple agents using decentralized navigation functions. In *Proceedings of Robotics: Science and Systems*, Cambridge, USA, June 2005.
- [3] N. Gulec and M. Unel. A novel coordination scheme applied to nonholonomic mobile robots. In *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC '05. 44th IEEE Conference on*, pages 5089–5094, 12–15 Dec. 2005.
- [4] Erica Z. MacArthur and Carl D. Crane. Compliant formation control of a multi-vehicle system. In *Proceedings of the 2007 IEEE International Symposium on Computational Intelligence in Robotics and Automation*, pages 479–484, 2007.
- [5] Thomas Kepfstedt, Masakazu Mukai, Masayuki Fujita, and Vladimir Sawodny. A Networked Formation Control for Groups of Mobile Robots using Mixed Integer Programming. *Proceedings of the 2006 IEEE International Conference on Control Applications, Munich, Germany*, pages 579–584, October 2006.
- [6] Gal A. Kaminka, Ruti Schechter-Glick, and Vladimir Sadov. Using Sensor Morphology for Multirobot Formations. *IEEE Transactions on Robotics*, 24(2):271–282, April 2008.
- [7] P. Ogren and N. E. Leonard. Obstacle avoidance in formation. In *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on*, volume 2, pages 2492–2497 vol.2, 14–19 Sept. 2003.
- [8] Pedro V. Fazenda and Pedro U. Lima. Non-holonomic robot formations with obstacle compliant geometry. Conference Paper, September 2007. 6th IFAC Symposium on Intelligent Autonomous Vehicles.
- [9] Silvia Mastellone, Dusan M. Stipanovic, Christopher R. Graunke, Koji A. Intlekofer, and Mark W. Spong. Formation control and collision avoidance for multi-agent non-holonomic systems. theory and experiments. *The International Journal of Robotics Research*, 27:107–126, January 2008.
- [10] S.G. Loizou and K.J. Kyriakopoulos. Navigation of multiple kinematically constrained robots. *IEEE Transactions on Robotics*, 24(1):221–231, Feb. 2008.
- [11] P. Urcola, L. Riazuelo, M.T. Lazaro, and L. Montano. Cooperative navigation using environment compliant robot formations. *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pages 2789–2794, Sept. 2008.
- [12] J. R. Asensio and L. Montano. A Kinematic and Dynamic Model-Based Motion Controller For Mobile Robots. *International Federation of Automatic Control 15th Triennial World Congress, Barcelona, Spain*, 21 July 2002.
- [13] J. Minguez and L. Montano. Sensor-based motion generation in unknown, dynamic and troublesome scenarios. *Robotics and Autonomous Systems, Elsevier Science.*, Vol. 52, No. 4.:290–311, 2005.
- [14] J. Minguez and L. Montano. Nearness diagram (nd) navigation: Collision avoidance in troublesome scenarios. *IEEE Transactions on Robotics and Automation*, 20(1):45–59, February 2004.
- [15] Cooperative robot team navigation video. Submitted with the paper also. <http://webdiis.unizar.es/~urcola/videos/iros09.avi>.
- [16] Danilo Tardioli and Jose Luis Villarroel. Adding multicast capabilities to wireless multi-hop token-passing protocols: Extending the rt-wmp. In *14th IEEE International Conference on Emerging Technologies and Factory Automation*, 2009.