Cooperative navigation using environment compliant robot formations

P. Urcola, L. Riazuelo, M. T. Lázaro and L. Montano
Instituto de Investigación en Ingeniería de Aragón, University of Zaragoza (Spain)
Email: {urcola, riazuelo, mtlazaro, montano} @unizar.es

***Abstract*— This paper reports an autonomous cooperative navigation system for robot formations in realistic scenarios. The formation movement control is based on a virtual structure composed by spring-dampers elements, which allows the formation to comply with the environment shape. A different navigation strategy is applied to the leader of the formation and to the rest of robots of the team. The leader plans the trajectories by using a two-level path planner with obstacle avoidance capabilities. The motion of the follower robots is controlled by the virtual structure, which adapts to the environment while the leader is tracked, taking into account the kinodynamic constraints of the vehicles. The system is evaluated in experiments carried out in simulations, some of them made in a realistic and complex urban scenario, and with real robots.**

I. Introduction

In the last years, many efforts have been made in researching and developing applications of mobile multi-robot systems. One of the main issues of these works is the maintenance of some kind of group formation, more or less adaptable to the environment, while moving the team to reach a goal. This topic is one key point in some real outdoors applications, where robots move through large scenarios and cooperation between them is needed. Unknown zones exploration, surveillance, connectivity maintenance or guiding are examples of applications in which cooperating robot teams improve the mission results. Concretely, in URUS project [1], an evacuation in an emergency situation in an urban environment mission is defined. When a fire is detected, robots guide confused people through safe ways to a safe area.

For all these kind of tasks, a robot team needs some flexibility to fit the environment as they are navigating and finding different obstacles such as narrow streets, intersections or curves.

There are many papers considering formation maintenance. In [2], a virtual structure is defined to model the shape of the formation. They make use of graphs to represent this structure where robots are placed in the vertices and links represent the relative positions between them. This model is quite rigid because it does not consider neither changes in formation topology nor flexible edges. Many works tried to model formation control using physic analogies, because it is easier to analyse the behaviour using mathematical methods like Lyapunov functions [3] to study system stability. Concretely, virtual potential fields are applied to model the influence of the location of each robot in the movement of the others. In [4], potentials are introduced to achieve this kind of interactions between robots in very populated groups. Derived from potential fields, virtual spring-based systems have been proposed. These kind of approximations use the Hook's Law to compute forces between robots giving some flexibility to the structure and then smoothness to the movement. Works like [5], [6] and recently [7] use springs and dampers to model the relative control among robots inside the formation. But no adaptation to the environment is considered.

One approximation for obstacle avoidance in formation is to compute a configuration space for the whole formation as explained in [8]. This approach is not adequate for our purposes because a complete map is needed and we pretend to work in unknown scenarios too. Another solution proposed in [9] for providing flexibility is changing the interconnections between robots to modify the shape of the formation while moving. The problem of this solution resides in how to decide the best formation shape depending on the environment.

To have a complete navigation system, kinematic and dynamic constraints and obstacle avoidance must be taken into account. In this way, works cited before do not consider at least one of these aspects.

In [10], potential fields are utilised to fuse formation control and obstacle avoidance techniques. This work and [11] are closer to our proposal. Both use the idea of forces between robots to keep formation and forces from the environment to avoid obstacles, but neither robot dynamic constraints nor complex and realistic environments are considered.

In [12], a multi-robot navigation function that includes kinematic and dynamic constraints is presented. This approach computes one single function for all the formation, considering all robot constraints. Our proposal has a decentralised view of the formation, using robot to robot relationships.

We propose here the control of the movement of robot formations by considering both kinematic and dynamic constraints for the robots and navigation in realistic scenarios with obstacles, where formations have to comply to the environment shape, while maintaining the formation topology.

Our proposal takes into account communication issues in the mobile ad-hoc network formed by the robot team. We use

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

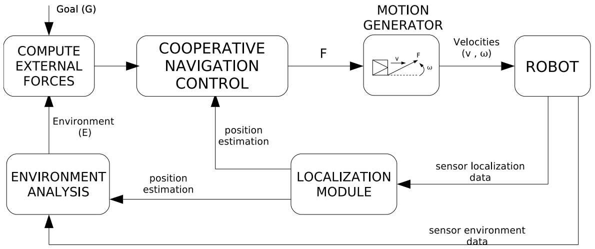


Fig. 1. General architecture for movement control

the real time protocol over wireless ad-hoc networks defined in [13] for data interchange in cooperative control. This ensures that all the data shared by the robots are available for every component in the team.

In section II the system overview is described. In section III we present the formation control schema and in IV the environment modelling for obstacle avoidance. Section V points out the motion generator of the robots and how to fuse formation control and obstacle avoidance with path planning. To conclude, we present in section VI the results obtained in simulated and real experiments and in VII the conclusions we extract from this work.

II. System Overview

The formation has only one leader. This robot plans the trajectory to the goal and tracks this path guiding the follower robots. Path planning and obstacle avoidance algorithms are applied to complete this task. The rest of team members follows the leader controlled by a model based on a virtual structure. Fig. 1 represents the general architecture for movement control. Each block represents a module that is running inside robots.

Robots gather data from their on board sensors and, depending on the type of sensor, information is used for different purposes. Environment perception and analysis block takes as input laser scans data. Localisation module integrates odometry and gyroscope data from the robot to estimate the position of the robot. To improve this estimation, we use scan matching techniques [14] for indoor experiments and GPS for outdoor ones. The environment analysis module calculates the influence of obstacles on the robot computed as a force. This influence and the goal attraction produce the external forces that will affect the robot movement. Formation structure forces are generated by a virtual spring-damper structure (see section III) which enforces the relative position and orientation between robots. These forces are computed in the cooperative navigation control module.

Once a unique force is computed for each follower, it is used as an input to the motion generator module. It computes velocity commands for the robots considering kinematic and dynamic constraints. As we are using non-holonomic robots, this command consists in linear and angular velocities to send to the robot controller.

In the following sections we explain the techniques used in each module and their integration. One of the major contributions of the paper is related to the use of the environment information to control and adapt the motion of the formation to the scenario geometry.

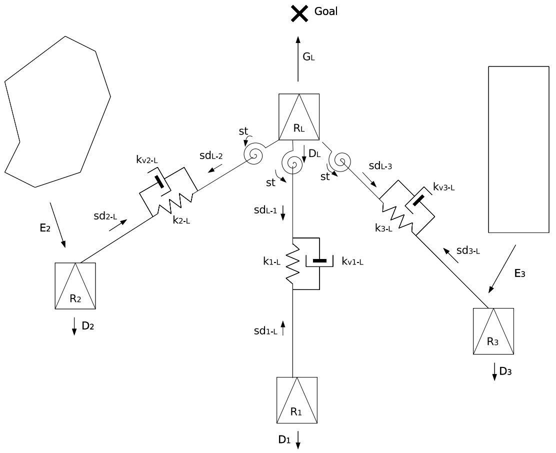


Fig. 2. Spring-damper structure for a robot team and forces involved in movement

III. Formation Structure and Dynamics

A. Structure model

In this section we present the model of a virtual structure for cooperative motion in robot formations. This model is based on a spring-damper analogy.

In order to incorporate obstacle avoidance capabilities it is necessary to build a structure that can be deformed in such a way the robots can avoid the obstacles while maintain the initial formation topology. Fig. 2 depicts an example of the virtual structure proposed. In it the robots are linked with linear spring-damper components, torsional springs or both of them. With the linear spring-damper link we achieve to maintain the distance between the robots and the torsional springs force the robots to maintain a given angle between them. This model provides the desired behaviour to the formation navigation, permitting changes in links in order to adopt different initial structures.

The main force is $\mathbf{G}_L$, applied to the robot leader ($\mathbf{R}_L$). This is the virtual force exerted by the goal on the leader to attract the robot. A first approximation of this force is computed as a function of the given leader's maximum desired velocity and so it is limited. In section V we present the path planning integration and how this force $\mathbf{G}_L$ is computed according to the plan.

Each spring-damper link between robots generates a force $\mathbf{SD}_i$. This force is defined as:

$$\mathbf{SD}_i = \sum_{j=1}^{N} \mathbf{sd}_{ij} a_{ij} + \sum_{j=1}^{N} \mathbf{st}_{ij} b_{ij} \tag{1}$$

with $a_{ij}, b_{ij} \in \{0, 1\}\ \forall i, j \in 1..N$. The elements $a_{ij} = 1$ of a selection matrix A represent the linear spring-damper links between robots, and the elements $b_{ij} = 1$ of a selection matrix B represent the torsional spring link between i , j robots. The force generated by the linear spring-damper link $\mathbf{sd}_{ij} = (sd_{ijx}\ , sd_{ijy})$ is computed as:

$$\mathbf{sd}_{ij} = k_s(d_{ij} - d_{0ij})\mathbf{d_u} + k_v\mathbf{v}_{ij} \tag{2}$$

The constants k_s and k_v are the spring and damping coefficients, respectively, d_{ij} and d_{0ij} are the distance between the robots and the rest distance, respectively, $\mathbf{d_u}$ is the unit vector linking i and j robots and $\mathbf{v}_{ij}$ is their relative velocity.

The force generated by the torsional spring link $\mathbf{st}_{ij} = (st_{ijx}\,, st_{ijy})$ is computed as:

$$\mathbf{st}_{ij} = \frac{k_{ro}(\theta_{ij} - \theta_{0ij})}{d_{ij}}\mathbf{n} \tag{3}$$

where k_{ro} is the torsional spring coefficient, d_{ij} is the distance between the robots $i\ ,\, j$, $\mathbf{n}$ is the normal vector on the movement plane and θ_{ij}, θ_{0ij} are the angle between robots $i\,,j$ and the rest angle between them, respectively. The constants, k_s, k_v, k_{ro} have been chosen to have a slightly overdamped behaviour.

A force $\mathbf{D}_i$ is introduced as a damping term due we want to simulate a real system. It is defined by:

$$\mathbf{D}_i = f_d\mathbf{v}_i \tag{4}$$

where f_d is the damping coefficient and $\mathbf{v}_i = (\dot{x}_i\,, \dot{y}_i)$ the velocity vector of the robot.

The obstacle avoidance capability is provided by means of external forces on each robot. This force $\mathbf{E}_i$ (see Fig. 2) is generated by the environment and it is applied to the slave robots in the formation. In the next section we explain the process of computing the value of this force, always bounded to a maximum value.

Summarising, the total force $\mathbf{F}_i$ applied on each follower robot i of the team is:

$$\mathbf{F}_i = \mathbf{SD}_i + \mathbf{D}_i + \mathbf{E}_i \tag{5}$$

It includes the influence of the spring-damper structure on each robot, the damping force, and the force generated by the environment that provides obstacle avoidance.

For the leader robot, the force $\mathbf{F}_L$ which provokes its motion is computed in a different way, as explained in the following subsection.

B. Leader motion planning

One of the major objectives of this work is to present a system for cooperative multi-robot motion control working in real scenarios in which the robots have to comply with environment obstacles. A lot of work has been made in single robot navigation and motion planning. We extend here the navigation under the previous hypotheses to multi-robot teams. The proposal is to use a single robot path planner for the formation leader to make the path computed this way optimal and free of collisions, local minima, and cyclic behaviours.

A two-level motion planning is used, one global and another local, for the leader robot motion control. The main difference between them is the working scale. While global planning admits low map resolutions, the local needs a highly detailed map to achieve the optimal feasible trajectory. Concretely we are using grid maps with a cell size of 0.5 m. and 0.05 m., for the global and local planners, respectively.

1) Global Path Planning: The global planner computes the general trajectory of the formation as a list of waypoints to the goal. Some of the planners with these capabilities need a complete map of the environment to work. But this situation is some kind of unreal, because environments changes dynamically and pre-computed maps may become obsolete very quickly. For this reason, we decided to use E* planner [15]. This planner is able to work in unknown environments, can incorporate dynamically new obstacles to the map and can replan the trajectory when changes affect the current one.

2) Local Path Planning: The local path planner uses the hybrid system defined in [14] which implements a synchronous planner-reactor, unifying the advantages of deliberative and reactive systems for navigation.

The requirements for this path planner are different that for the global one. A moving map joined to the leader robot that keeps a local plan is used. The optimal planning method in this situation is harder to find and deeply depends on the application as commented in [16]. In this system a D* algorithm is used to compute the path in a dynamic environment. For obstacle avoidance issues, the system uses the Nearness Diagram (ND) method [17], which gives robot a very accurate trajectory tracking while avoiding static and dynamic obstacles.

In the formation scheme, the attractive force $\mathbf{G}_L$ (defined in section III-A) to be applied to the leader is computed from the solution provided every sampling time by the described planner. The force module is limited according to the dynamic constraints of the vehicle. The final force applied to the leader is,

$$\mathbf{F}_L = \mathbf{G}_L + \mathbf{SD}_L \tag{6}$$

where $\mathbf{SD}_L$ are the forces induced by the connected robots to the leader through the spring-damper structure. Note that the other forces computed for the follower robots are not longer used here, because their effects are involved in the planner-reactor force solution.

IV. Adaptation to the Environment

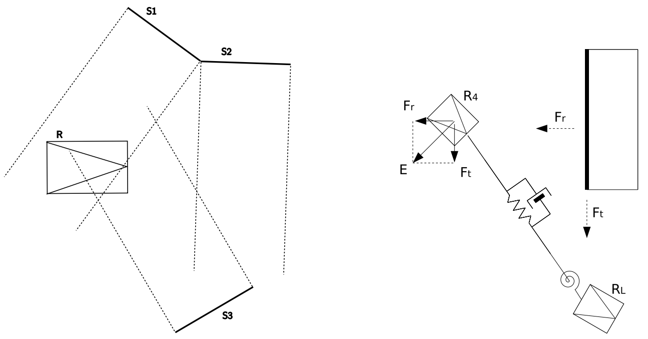
We describe in this section how the system processes the sensor information which perceive the environment, and the way it is modelled and integrated for the formation navigation in order to adapt it to the environment. That is how $\mathbf{E}_i$ in equation (5) is computed.

A. Sensor data processing

We are using robots equipped with laser rangefinder sensors that compute the distance to the obstacles in a 180 degrees field of view. For the purpose of obstacle avoidance of the robots in the formation, the sensor information has to be filtered and processed. The basic treatments are:

1) All the obstacles that are not close enough, given a security distance to the robot, are discarded.

2) The points inside this security zone are grouped in straight line segments using a split and merge algorithm. The parameters of the segmentation algorithm are tuned depending on the size of the obstacles expected in the environment.
3) An influence zone is computed for every segment. It is defined as the *infinite* rectangle generated while shifting the segment orthogonally to its own direction. All those segments that keep the robot inside their influence zone are considered (see Fig. 3(a)).



(a) Influence zones.

(b) Forces generated by segments.

Fig. 3. In (a), robot movement will be affected by segments s1 and s3 because it is inside their influence zones. Segment s2 will be discarded because robot is not inside its zone. In (b), the segments chosen generate repulsive forces to avoid obstacles and tangential forces to guide robots towards the leader.

This environment processing eliminates undesired behaviours due to obstacles that do not influence the robot motion.

B. Interaction with robots

Each of the obstacles that are influent in robot movement, generates a virtual force which consists in two different components, as shown in Fig. 3(b).

- The repulsive component $\mathbf{F_r}$ is the one that avoids obstacles. It is defined by equation (7), where d is the orthogonal distance from the robot centre to the straight line that holds the segment, k is a parameter to tune the repulsive force from each segment and $\mathbf{u_n}$ is a unit vector orthogonal to the segment direction; it is always pointing from the segment to outside.

$$\mathbf{F_r} = \frac{k}{d^2}\mathbf{u_n} \tag{7}$$

- The tangential component $\mathbf{F_t}$ (equation (8)) is used as a guide for the robots to follow the leader. It is parallel to the segment and it is pointing to the direction of the projection of the vector defined from the robot centre to the leader of the formation.

$$\mathbf{F_t} = \frac{k}{d^2}\mathbf{u_t} \tag{8}$$

The k parameter is tuned to start the obstacle influence when the robot is at a given distance from it. One special case happens when a segment direction is close to the leader orientation, as in case of corridors. In this situation, the tangential force would not guide but just accelerate the robot towards the leader, so it is not considered.

The total force $\mathbf{E_i}$ (equation (9)) that the environment induces in each robot i, is computed from all the N segments that influence each robot,

$$\mathbf{E_i} = \sum_{j=1}^{N}\mathbf{F_{ri}} + \sum_{j=1}^{N}\mathbf{F_{ti}} \tag{9}$$

V. Motion Generator and Formation Navigation

Once all the forces, $\mathbf{F}_L$ for the leader and $\mathbf{F}_i$ for the follower robots, have been computed they have to be applied using a Motion Generator (MG) for differential-drive mobile robots (those used in our experiments). The MG transforms these forces into linear and angular velocities according to the equation:

$$\dot{\mathbf{x}}_i = \mathbf{P}\mathbf{x}_i + \mathbf{Q}\mathbf{F}_i \tag{10}$$

where

$$\mathbf{P} = \frac{-2b}{mr}\begin{bmatrix}1 & 0\\ 0 & k_i d^2\end{bmatrix} \qquad \mathbf{Q} = \frac{1}{m}\begin{bmatrix}1 & 0\\ 0 & k_i h\end{bmatrix} \tag{11}$$

By solving this differential equation we can obtain the linear and angular velocities $\mathbf{x}_i = (v\ , \omega)$. This model takes into account the kinematic and dynamic constraints of the robot, so generating feasible trajectories for all the robots in the formation. The parameters r, d, h are geometric constants of the robots (wheel radius, distance between the robot centre and the wheel, and moment arm respectively) and m is the mass. More details about the model and how the parameters b (viscous friction) and k_i (inertial coefficient) can be tuned to obtain an overdamped behaviour available in [18]. The stability issues of the controlled system and its dynamic behaviour as a function of the parameters is also addressed there.

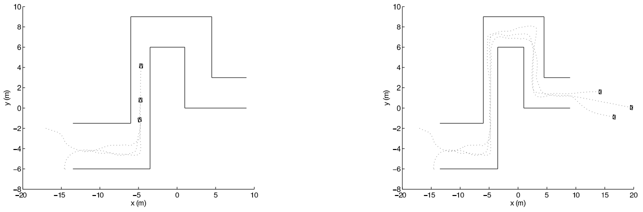
All the techniques explained so far are integrated in the whole system controller as shown in Fig. 1. The whole controller takes into account the kynodinamic constraints of the robots to compute feasible trajectories.

VI. Experimental Results

A. Simulations

We have addressed three experiments to evaluate the system performance, two in simulations using the Player/Stage platform [19], which takes into account robot dynamics, and one using the Pioneer 3AT real robots. In the first experiment, a scenario with three corridors is built. It allows to test the dynamic behaviour of the robots using the whole control system; the formation has to adapt its shape to narrow corridors while has to react to sudden changes in orientation. In Fig. 4 the robot trajectories exhibit a stable and smooth behaviour when the robots enter in another corridor. A short video of this experiment is attached electronically[1].

[1] Also available at: http://robots.unizar.es/videos/videos.html



(a)

(b)

Fig. 4. Narrow corridor simulation. (a) Robots enter into the narrow corridor adapting the shape of formation. (b) Robots get back to the original shape of the formation when they get out of the corridor

The second experiment tries to show the global behaviour in a large scenario which has different kind of obstacles and corridors simulating one of the real experiments to carry out in the URUS project. The whole campus scenario is approximately 100m. × 100m. sized and the mission consists on an evacuation reacting to an emergency (i.e., a fire).

Fig. 5 represents four snapshots in different instants of the simulation experiment carried out in a zone (50m. × 50m.) of the scenario. The trajectories planned for the leader and the ones corresponding to the follower robots are depicted. No collision is produced in any moment of the simulation due to reactive navigation system.

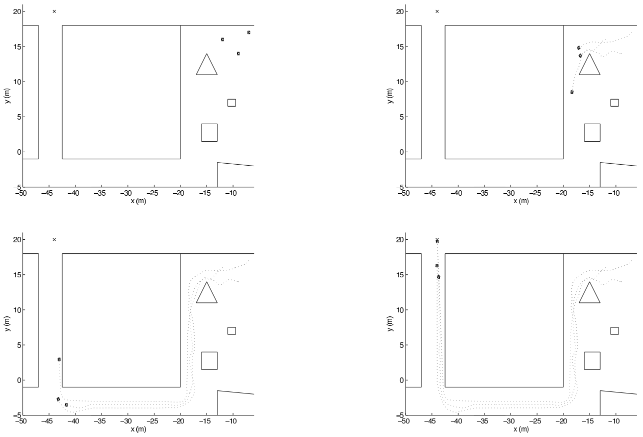


Fig. 5. From top-left to bottom-right, four snapshots of the zone of the scenario where the evacuation experiment will take place. The robots reach a safe place (a X in the figures) from the square where the fire has started

In the third experiment we test the whole system with real robots. The scenario consists in a wide corridor with a narrow zone in the middle, where only one robot at a time can navigate through. The dynamic behaviour and the capability to adapt the shape of the formation to the real environment constraints are analysed. Fig. 8 shows three snapshots of the experiment.

Fig. 6 shows the commanded and real linear velocities for one of the follower robots. It can be seen how real velocities are close to the commanded ones. The conclusion is that the controller of the system is well tuned to control the real robots, that is the robot dynamics have been well modelled.

Fig. 7 presents the linear and angular velocities during the experiment. The velocity graphics of real experiment show how the leader motion is propagated to the follower robots. In the linear velocities figure can be seen that the leader (Robot 1) starts its movement at the beginning of the experiment but, the followers keep stopped more than 20 cycles until the commanded velocities computed from the spring forces exceed the minimum value that the Pioneer robots need.

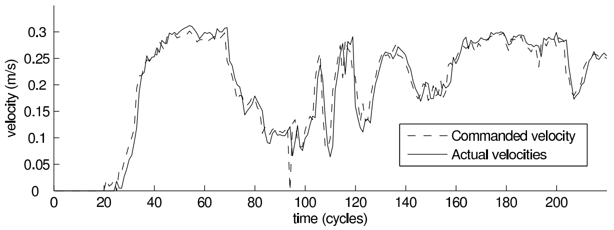
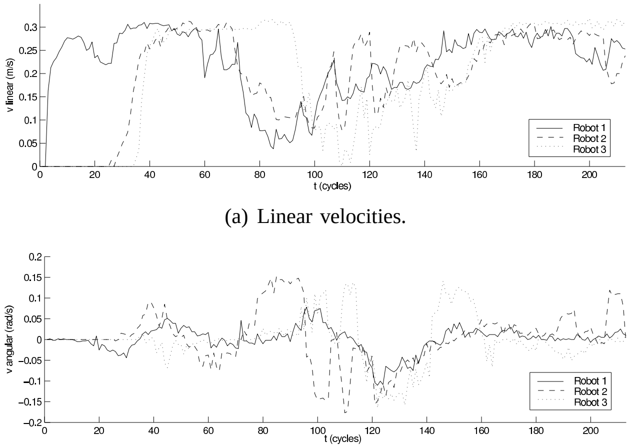


Fig. 6. Robot commanded and real velocities for one of the followers



(a) Linear velocities.

(b) Angular velocities.

Fig. 7. These are the velocity graphics from real experiments. It can be seen how the robot formation structure propagates the leader movement to the followers in presence of obstacles.

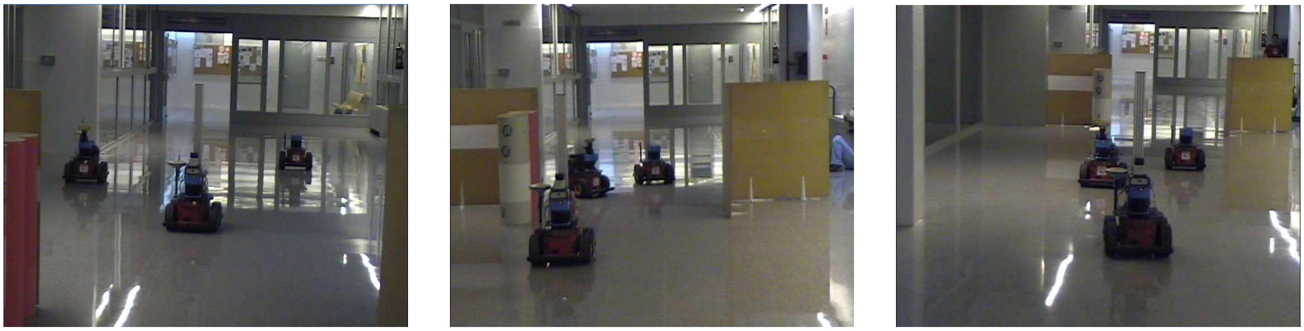
Between cycles 70 and 100, Robot 2 slows down to let Robot 3 go through the narrow zone. This behaviour is propagated to the leader, which consequently slows down too. Relevant deviations between cycles 70 and 160 can be found out in angular velocity graphic (Fig. 7(b)). These perturbations are caused by the influence that the narrow zone (see Fig. 8) induces on the formation structure.

In the control of real robots, some additional issues have to be considered. The relative localisation of the robots has to be computed and transmitted to all the robots, as necessary data to obtain the motion commands. But it has to be made with real-time constraints. A real-time wireless multi-hop protocol [13] is used for this proposal.

Under real conditions, our navigation system fits the time constraints of the real robot control cycle, which is strongly related to the time the sensors needs to gather data (~ 300 ms).

VII. Conclusions

We have developed a navigation system for robot formations that is able to adapt the shape of the structure to the



(a) Initial configuration.

(b) Passing through a narrow zone.

(c) Robots recover original relative positions.

Fig. 8. Screenshots from real experiments. We can see the formation in its initial configuration (a), how they adapt the formation structure when the environment requires it (b) and the way they get back to the original shape when no obstacles are around.

environment. The main characteristics of the system are:

- The controller of the leader uses a two-level path planning and obstacle avoidance hybrid system to drive the formation within the free space.
- The motion control of the robots in the formation is based on a virtual spring-damper structure, which enforces the relative position and orientation of the robots.
- The same model allows to adapt the shape and motion of the formation to the environment geometrical constraints.

This motion control system is flexible enough to cope with the different situations that can be found in real environments. The simulation and real experiments show that the system adapts to the environment, computing feasible and smooth trajectories compatible with the kinodynamic robot constraints.

Development of techniques to avoid problems coming from the local reactivity provided by the virtual model to the follower robots, to improve the environment model and the cooperative planning to deal with more general, dynamic and no polygonal scenarios, is an ongoing work. Improvements in robot localisation by using SLAM techniques for indoor and outdoor environments will be integrated in the system.

Another important work is to do the real experiments of the URUS project in real scenarios. The techniques presented in this paper have to be integrated with cooperative perception systems to build a robot network system for applications in urban environments.

REFERENCES

[1] "URUS:Ubiquitous networking Robots in Urban Settings," EU Project IST-1-045062, Web site: http://www-iri.upc.es/groups/urus/.

[2] M. A. Lewis and K.-H. Tan, "High precision formation control of mobile robots using virtual structures," *Auton. Robots*, vol. 4, no. 4, pp. 387–403, 1997.

[3] H. G. Tanner and A. Kumar, "Formation stabilization of multiple agents using decentralized navigation functions," in *Proceedings of Robotics: Science and Systems*, Cambridge, USA, June 2005.

[4] N. E. Leonard and E. Fiorelli, "Virtual leaders, artificial potentials and coordinated control of groups," in *Decision and Control, 2001. Proceedings of the 40th IEEE Conference on*, vol. 3, 4-7 Dec. 2001, pp. 2968–2973vol.3.

[5] J. H. Reif and H. Wang, "Social potential fields: A distributed behavioral control for autonomous robot," *Robotics And Autonomous Systems*, vol. 27, pp. 171–194, 1999.

[6] 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*, 12-15 Dec. 2005, pp. 5089–5094.

[7] E. Z. MacArthur and C. 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*, 2007, pp. 479–484.

[8] P. Ogren and N. E. Leonard, "Obstacle avoidance in formation," in *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on*, vol. 2, 14-19 Sept. 2003, pp. 2492–2497vol.2.

[9] J. P. Desai, J. P. Ostrowski, and V. Kumar, "Modeling and control of formations of nonholonomic mobile robots," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 6, pp. 905–908, December 2001.

[10] P. V. Fazenda and P. U. Lima, "Non-holonomic robot formations with obstacle compliant geometry," Conference Paper, September 2007, 6th IFAC Symposium on Intelligent Autonomous Vehicles.

[11] S. Mastellone, D. M. Stipanovic, C. R. Graunke, K. A. Intlekofer, and M. W. Spong, "Formation control and collision avoidance for multi-agent non-holonomic systems. theory and experiments," *The International Journal of Robotics Research*, vol. 27, pp. 107–126, January 2008.

[12] S. Loizou and K. Kyriakopoulos, "Navigation of multiple kinematically constrained robots," *IEEE Transactions on Robotics*, vol. 24, no. 1, pp. 221–231, Feb. 2008.

[13] D. Tardioli and J. L. Villarroel, "Real time communications over 802.11: Rt-wmp," in *Proc. IEEE Internatonal Conference on Mobile Adhoc and Sensor Systems MASS 2007*, 2007, pp. 1–11.

[14] J. Minguez, F. Lamiraux, and L. Montesano, "Metric-based scan matching algorithms for mobile robot displacement estimation," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, Barcelona, Spain, April 2005, pp. 3557–3563.

[15] R. Philippsen and R. Siegwart, "An interpolated dynamic navigation function," in *Proceedings of the IEEE International Conference on Robotics and Automation*, no. 3782-3789, Barcelona, Spain, 2005.

[16] D. Ferguson, M. Likhachev, and A. Stentz, "A guide to heuristic-based path planning," in *Proceedings of the International Workshop on Planning under Uncertainty for Autonomous Systems, International Conference on Automated Planning and Scheduling (ICAPS)*, June 2005.

[17] J. Minguez and L. Montano, "Nearness diagram (nd) navigation: Collision avoidance in troublesome scenarios," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 1, pp. 45–59, February 2004.

[18] J. R. Asensio and L. Montano, "A kinematic and dynamic model-based motion controller for mobile robots," in *15th Triennial World Congress*, 21-26 July 2002. [Online]. Available: http://webdiis.unizar.es/ lmontano/Dynamic-IFAC02.pdf

[19] B. P. Gerkey, R. T. Vaughan, and A. Howard, "The player/stage project: Tools for multi-robot and distributed sensor systems," in *ICAR*, 2003, pp. 317–323.