Experimental Studies of Multi-robot Formation and Transforming

Lei Liu*, Yongji Wang*, Shuanghua Yang,**
Graham Watson*, Brian Ford*****

** Department of Control Science and Engineering, Huazhong University of Science and Technology*
Wuhan, Hubei, 430074, China (e-mail: liulei@smail.hust.edu.cn, wangyjch@mail.hust.edu.cn).
*** Computer Science Department, Loughborough University, Loughborough, Leicestershire, LE11 3TU*
UK (e-mail: s.h.yang@lboro.ac.uk).
**** Systems Engineering Innovation Centre, BAe Systems, Loughborough, Leicestershire, LE11 3TU*
UK (e-mail: b.ford@lboro.ac.uk).

Abstract: Focusing on multi-robot flocking, this paper develops a formation holding and transforming method by using the leader-follower strategy on the double or triple robot groups. With this method the state and role of each robot can be identified, and the most suitable topology of the multi-robot formation is decided in order to go through a gap or avoid an obstacle. Furthermore, the algorithms and strategies are implemented on the Koala Robots of SEIC/BAe systems, and the communication between each robot is based on the Internal Communication Engine (ICE), which is a popular middleware used for building distributed communication environments. Finally, the experimental results demonstrate the efficiency of the proposed method.

1. INTRODUCTION

Robotics systems are widely used in modern manufacturing and industry to improve efficiency, accuracy and working environments. In the military domain robotic systems are also being adopted to perform dangerous and hazardous missions. With this increasing utilisation, a single robot can't meet all aspects of our needs because of its inherent limitations. As a solution, multi-robot systems are being investigated for many applications that are difficult or time consuming for a single robot. In many situations multi-robot systems can be used to improve sensing performance, robustness and reliability.

While robot control is considered to be a well-understood problem area, most of the current successful cases in multi-robot coordination do not rely on the results available within the Control Theory and Dynamical Systems literature. When operating in unstructured or dynamic environments with many different sources of uncertainty, it is very difficult, even impossible, to build mathematic model or design controllers. Thus, existing approaches to robot formation control generally fall into three categories: Behavior-based approach (Balch, et al., 1998; Stroupe, et al., 2003) desires to prescribe specific behavioral characteristics to each robot, such as collision avoidance and formation keeping, and the actions of the robots are derived by weighting the relative importance of them. However, it is difficult to analyse the approach by theoretical formalization and consequently it is not easy to guarantee the convergence of the robots into a desired formation. Secondly, the virtual structure approach (Shao, et al., 2006) considers each robot to behave as a particle in a single virtual rigid structure and to follow desired trajectories. The approach is not assigned to each single robot but to the entire formation as a whole. Thus the behavior of the robot formation is predictable and consequently the control of the robot formation is straight forward. Nevertheless this approach is used for large number of robots, which already requires a large inter-robot communication bandwidth for real world flocking control. The third is the leader-follower approach (Desai, et al., 2001; Das, et al., 2002; Tanner, et al., 2004; Consolini, et al., 2007), with a robot of the formation, designated as the leader, moving along a predefined trajectory whilst the other robots (the followers) maintain a desired posture (distance and orientation) relative to the leader. The main criticism of the leader-follower approach is that the formation does not tolerate leader faults and exhibits poor disturbance rejection features. In spite of these deficiencies the leader-follower approach is particularly appreciated because of its simplicity and scalability.

Our work is based on the leader-follower approach described in the thesis of Desai (Desai, 1998). We combine it with our flocking topology to conquer the deficiencies of the leader-follower approaches above and to extend it to cover a wider range of applications. The results of the simulations demonstrate the efficiency of our control strategy. The rest of the paper is organized as follows: Sect. 2 is the holistic structure of our research, Sect. 3 proposes the strategy and algorithms on flocking control and topology assignment, Sect. 4 introduces experimental tools, including robot and simulator, and Sects. 5 and 6 are the results and conclusions of the paper.

2. FRAMEWORK OF THE CASE

Multi-robot flocking control is widely referred to in many areas. However, most of the literature focuses on one problem area, such as formation control or obstacle avoidance. The results are also face domains specific, rather than for a common system.

In this paper we try to design an integrated multi-robot flocking system. The key is that the algorithms and programs are common to everyone in the flock. Therefore, though a robot can act different roles in the flock (for example a leader or a follower), they are all equal and interchangeable. In this way, theoretically, the multi-robot system is more dependable, which is necessary to the real world applications, such as the military environment.

Our work can be separated into three tasks: Formation Control, Obstacle Avoidance and Navigation. The Formation Control task includes 3 functions: Initialize, Establish and Transform, which are the basic functions of multi-robot flocking. The task of Obstacle Avoidance, including collision avoidance, has the highest priority. Navigation reflects the performance of the whole system. The overall framework of the system is shown in Fig.1.

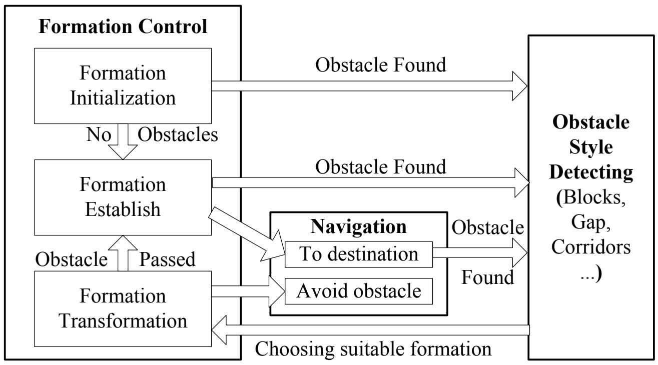


Fig.1. Overall framework of the system

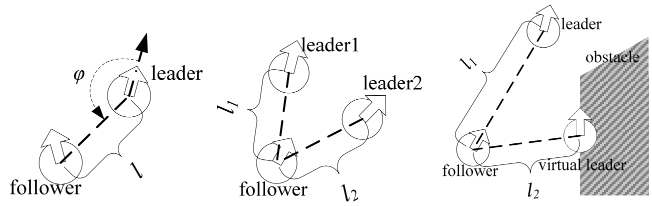
Normally the initial position of the robot is always available before flocking. Lemay, etc. [2004] proposed an approach that allows the group to determine the most appropriate assignment of positions in the formation autonomously by using directional visual perception to localize robots. Considering the complexity and efficiency of the vision process, especially for a multi-robot system, we chose a laser range and odometer to localize each robot and detect the environment with information from adjacent robots and obstacles.

3. FORMATION AND TRANSFORMING STRATEGY

The aim is to design a philosophy which is based on the leader-follower approach to organize and reorganize the formation, and keep a stable formation shape of multiple mobile robots. While this philosophy is similar in spirit to a behavior-based control paradigm, we differ by proposing more formal, control-theoretic approach in developing the basic components and their composition, considering the expansibility to a large number of robots and a 3D formation.

3.1 Local Leader-Follower Structure

On formation control the problems can be simplified to control the relative position and orientation of the robots in a group while allowing the group to move as a whole. As mentioned before, the leader-follower formation control plays an important role in multi-robot flocking research for its practicability and easy realization. Under this structure the relationship can be described by both distance and orientation between the leaders and followers. As shown in Fig.2 (a), the aim of the controller is to maintain the desired relative bearing (φ), and a desired separation (l), between the leader and the follower, which is called l-φ control strategy (Desai, *et al.*, 2001; Desai, 1998). Some literature defines the relative angle φ in the follower frame (Consolini, *et al.*, 2007).



(a) l-φ Strategy (b) l-l strategy (c) obstacle avoidance

Fig.2. Control Strategies of Multi-robot Flocking

In other scenarios, the follower has to keep the correct distance from two other robots to maintain stability and avoid collision. In this situation, the l-l strategy is designed to control the formation (shown in Fig.2 (b)). This control strategy is also used widely for obstacle avoidance where the obstacle is a virtual leader (Das, *et al.*, 2002), as shown in Fig.2 (c).

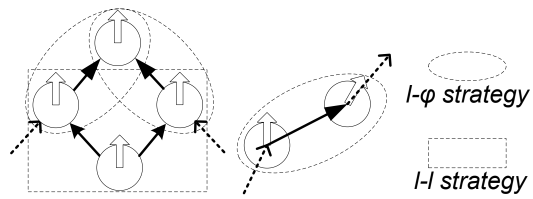


Fig.3. Structure Analysis of the Multi-robot Formation

In our work we combine the l-φ and l-l control strategies in flocking to maintain and transform the formations. In multi-robot formation control two typical shapes act as the basic elements of most complex formations: the diamond style and the line style (Balch, et al., 1998). As shown in Fig.3, in the diamond style of the formation, the robots in both wings need to keep their correct relative position and orientation to the leader, and the ones behind only consult the single robot in front. In this case, we chose l-φ strategy to control this relationship. However, the robot in the middle should pay attention to the both robots ahead to avoid colliding into other robots, thus the l-l strategy is necessary in order to restrict its movement. Furthermore, in the line style formation, the robots travel as a queue. So the l-φ strategy is suitable for this situation.

Therefore most complex formations can be constructed based on these two local structures. The scalability of the formation is another important part to be considered, but is always simplified by many existing works. Therefore, we need to analyse the topology designed in our multi-robot formation control, which can be extended to any large scale and even into 3D formations.

3.2 Formation Topology Analysis

Even though most of the literature (Balch, et al., 1998; Desai, et al., 2001; Fredslund, et al., 2002; Consolini, et al., 2007) claims that their results and algorithms can be recursively extended to any (N) robots, the topology designed to define each robot in the formation of multi-robots acts as the bottleneck, since they always identify the robots in the formation simply as a queue. If the scale and shape of the formation become large and complex, and the relationship between the adjacent robots cannot be reflected by the identification, it will be difficult to establish such a formation.

Here we design a simple but efficient formation topological method to identify the robots in the formation. In this method we can give a particular identification to each robot, which represents the role and state of the robot in the formation. After that, by the ID, the robots can recognize each other in the environment by communication.

The topological method is inherited from the Geometric Space Theory. In the N-dimension space, the robot is identified as an N-dimension vector (m_1, m_2, …, m_N), with the leader (0, 0, …, 0). In this way, m_t (t = 1, 2, …, N) shows not only the ID number of the robot, but also the position and motion relative to others. In Fig.4, the topology of multiple mobile vehicles in a 2D environment is shown. The leader is (0, 0), the robots on the left edge are identified as (m, 0), with the ones on the right are (0, m) (m = 1, 2, …, N). Therefore the robot (m, 0) is the follower of (m-1, 0), and vice versa. For the robot in the middle, it satisfies the *l-l* control strategy. So the robot (m, n) should follow the robots (m-1, n) and (m, n-1) with anticipated separation. With the expected *l* and *φ*, the formation can be extended to any N robots and any formation.

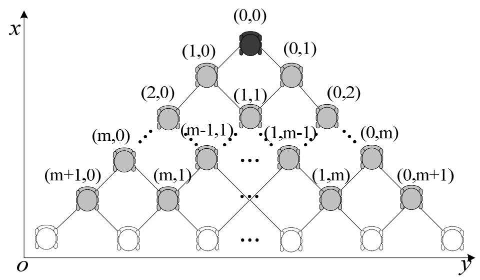


Fig.4. Topology Definition of 2D Formation

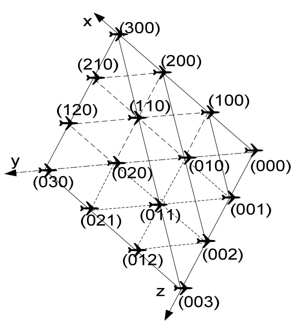


Fig.5. Topology Definition of 3D Formation

Furthermore we try to establish the 3D environment by using this method as shown in Fig.5, which has been used in the unmanned aerial vehicles (UAVs) (Tanner, 2007), such as helicopters (Saripalli, et al., 2002), unmanned spacecraft (Bikdash, et al., 2004) and missiles (Hughes, 2002). The formation of UAVs should keep a solid style, thus, each member should be defined by a triple vector with the whole formation as a solid matrix. Most of the elements in the formation are shown in Fig.5. The ones in the middle such as (m, m, m) may relate to (m, 0, 0), (0, m, 0) and (0, 0, m). In this situation it would be almost impossible to be identified as a queue by previous methods.

For a complex formation the topologic definition is not unique. It relies on the style of the formation.

4. EXPERIMENTAL PLATFORMS

Our experiment is based on the Koala ground robots in the Systems Engineering Innovation Centre (SEIC) of BAe Systems, which form the Configurable Systems Engineering Research Tool (ConSERT). The Koala is a midsize robot designed for research and development on unmanned vehicles, as shown in Fig.6. It rides on 6 wheels for indoor operation by different drive, and has been equipped with stereo pan-tilt cameras, compact laser rangefinders and wheel counters.

Fig.6. Koala Mobile Robots

Different from the conventional controllers, most of the midsize robots have equipped a laptop as the “brain”. The so-called “thinking” is organized by the developer’s programs. Between the robot and the programs is the Application Program Interface (API). Here, SEIC chose ORCA (Open Robot Controller Architecture), which is an open-source project and applies Component-Based Software Engineering principles to robotics (Makarenko, et al., 2006), as the API of the Koala robot. ORCA has provided the means for defining and developing the components which can be pieced together to form any complexity of robotic system. Users and developers can develop new programs to imitate the components and even use the existing components directly, because some basic functions are contained in the existing components such as laser and odometer utilization.

Before downloading the programs onto the Koala robots, we can test the program on the Player/Stage multi-robot simulator. The Player/Stage Simulator was developed by University of Southern California in 1999 to provide a simulation environment for multi-robot systems. Since then, it has been widely adopted, developed and extended by researchers.

Player and Stage play different roles. Player is a multi-threaded robot device server which allows the developer to access the hardware devices. Stage is the software to simulate multiple robots in a 2D environment with supporting tools and libraries. In Stage we can create any environment by drawing different maps.

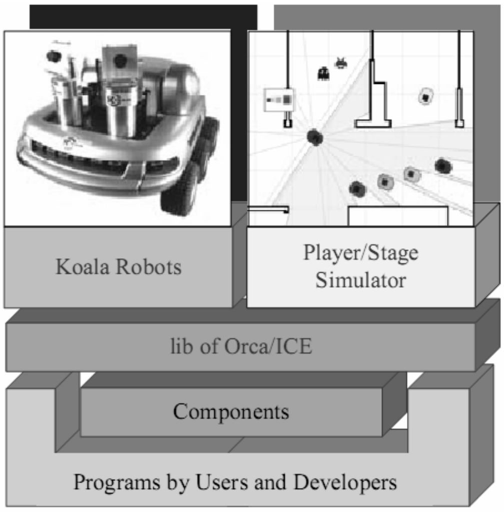


Fig.7. Configuration of the Robot Control Platform

Fig.7 shows the configuration of the multi-robot control and simulation. On the top of the figure there are two independent experimental platforms, the Koala Robots and the Player/Stage Simulator. User's programs are always tested on the simulator before being downloaded to the robots. The library files supported by ORCA are between the components and the platforms. They provide the API to developers. The programs developed by users are in the bottom of the configuration. They are based on the components structure of the ORCA.

Moreover, communication between individual component is provided by the Internal Communication Engine (ICE). ICE is an open-source and object-oriented middleware platform developed by ZeroC (http://www.zeroc.com/). The middleware is an important step forward towards making distributed computing available to application developers. The middleware platform takes care of the majority of networking chores, so it is much easier to build distributed applications with the middleware, without having to be proficient in networking (Henning, et al., 2007).

To enhance the catholicity, ORCA is built upon the ICE system. Thus, we can read the sensors and control the robot from anywhere on the network. Considering that most developers place primary importance on the results rather than the paths, ORCA makes the communication transparent to the developers.

5. EXPERIMENTAL RESULTS

Some typical experiments are designed to demonstrate our formation and transforming strategy. The simulations are based on the Player/Stage Simulator. In the experiment we used six robots in the formation only, which should be enough to illustrate our algorithm and strategy.

In the following figures the small blocks symbolize the mobile robots. The laser sensor reading has been indicated in the figures by the shaded sectors, which cover 180 degrees area in the front of the robot. The areas with the black outlines are the obstacles. To express the moving trend clearly the trail of the robot is shown in some of the figures, as the small pane following the robot.

To realize the complete self-configuration, the robots need to self-localize autonomously. However, GPS is forbidden in our environment. Thus we use a consulting wall and laser sensor to control the robots' relative positions, as in Fig.8. The only problem is how to separate the characters of the wall and neighbour robots. With the high accuracy laser sensor it is possible to detect the profile of the target. Thus, with the different profiles of the wall (plane) and adjacent robot (surface), the robot is able to distinguish between them.

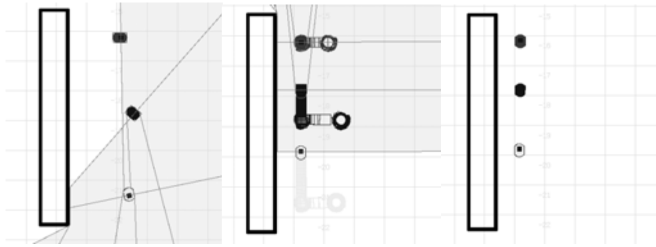


Fig.8. Position Initialization

After that the robots try to build a stable formation in a no-obstacles environment. As shown in Fig.9, six mobile robots start as a single line, which is initialized autonomously in Fig.8. The one on the top moves first and is identified using the ID (0, 0) as the leader of the formation. Thereafter, the one whose front laser reading changes will be the next one to fill in the formation and its ID will be confirmed simultaneously.

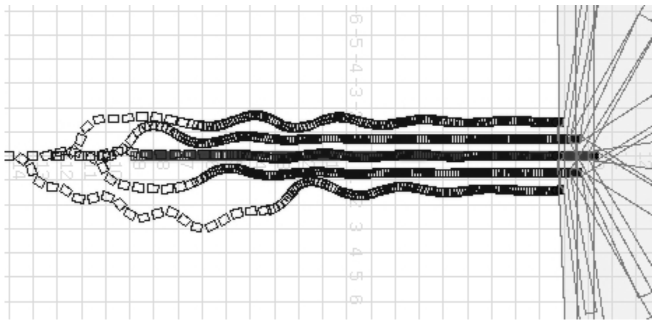


Fig.9. Establishing Formation without Obstacles

As mentioned before our formation strategy is based on the local leader/follower method. In Fig.9, the top of the triangle shape of formation is the leader robot, with two followers on both the left and right hand sides behind it. Each follower of the leader has another two followers behind it as well. In this way, to hold a stable style of the whole formation means to keep a stable relationship between all these leader/follower pairs as distance l and orientation φ.

Furthermore, when the leader arrives at the obstacles, it's necessary to transform the formation into an easy-passing shape to avoid them. The next three experiments are mainly on how to avoid different obstacles by transforming into special formations.

The first one is the narrow gap, which can be passed by two parallel robots at most. Thus the strategy is to change the formation back to the line style. Changing formation is equivalent to changing the topology of the multi-robot. In the six-robot group, the identification distributes as a similar triangle matrix:

$$\mathbf{M}_f = \begin{bmatrix} \mathbf{1} & \mathbf{1} & \mathbf{1} \\ \mathbf{1} & \mathbf{1} & \mathbf{0} \\ \mathbf{1} & \mathbf{0} & \mathbf{0} \end{bmatrix}.$$

In the 3×3 matrix, the number of rows and columns compose the ID of the robot in the formation. If the robot (m, n) exists in the formation, the corresponding element in the Matrix of row m and column n is **1**. While the flocking transforms to a line style formation, the matrix $\mathbf{M}_f$ will be transformed to a line vector:

$$\mathbf{V}_f = [00 \quad 10 \quad 20 \quad 01 \quad 11 \quad 02].$$

The ID in the vector $\mathbf{V}_f$ has to correspond to the element **1** in the matrix $\mathbf{M}_f$. It is a simple matrix transformation but efficient in multi-robot control.

As shown in Fig.10, when the leader detects the obstacle by a laser sensor it will tell everyone in the group to transform the formation a strategy according to the environment. After the whole formation runs through the gap, the formation should be transformed to the triangle shape again. However, the reforming of the triangle formation can not be decided by the leader alone but everyone in the formation, as the leader can not sense if the last robot has passed obstacle the obstacle.

The next experiment shown in Fig.11 considers the situation of an obstacle blocking the way of formation. We set a big obstacle in the way of the multi-robot formation. In this situation, transforming the formation is not enough. The flocking should try to plan a safe path to avoid the obstacle based on Laser based Navigation.

Our strategy for the leader is to find a path around, and then lead the formation past safely. In Fig.11, the flock transforms into a line formation first, and simultaneously the leader finds a suitable point of laser reading which will point to the way around. After passing the obstacle the flocking will resume back to a triangle formation again.

The third experiment is in a corridor environment. A corridor is a typical environment of complex obstacles for multi-robot navigation (Sarid, et al., 2007). Obviously it's impossible to go through the corridor while maintaining any rigid formations. The formation should be flexible and adapt to the curve of the corridor. Therefore, in Fig.12, every robot only follows the one before it until the leader, so the whole leader-follower structure is completely dynamic. Combining obstacle avoidance and laser navigation strategies, the flock can walk through the corridor without any collision.

However, by using the leader/follower strategy, any leader's disturbance affects the followers' motion and the influence will be passed and amplified gradually to the last follower, which is even more observable in the line formation. Therefore, a tiny deflexion of the leader will cause a large movement in the followers. The common method to solve this problem is to choose suitable parameters for the motion controller.

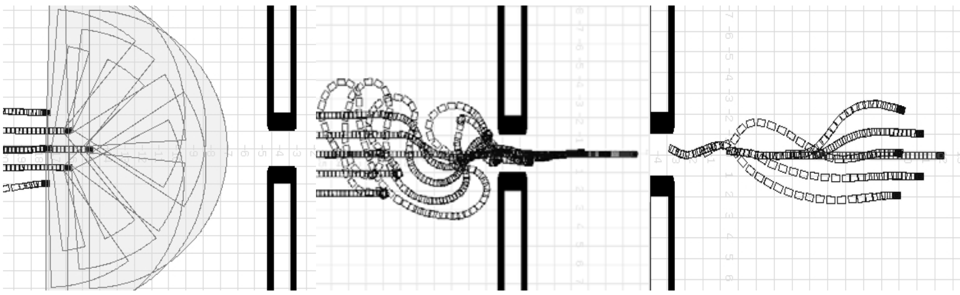


Fig.10. Walking Through a Gap

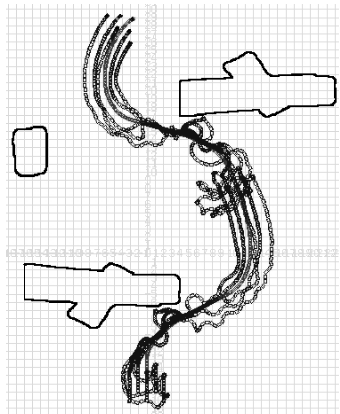


Fig.11 Avoiding Big Obstacles

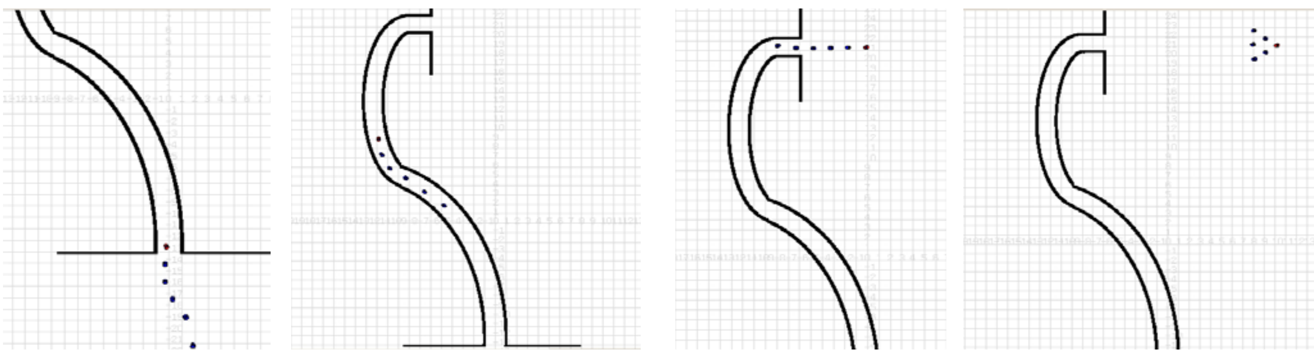


Fig.12. Walking Through a Corridor

6. DISCUSSION AND CONCLUSIONS

The paper is based on the full communication between the robots, so that everyone in the flock can share the information with each other. Considering if communication is limited (Fredslund, et al., 2002), the strategies in this paper would not keep the formation stable. Because the role of the robot is decided autonomously by its initial position and communication status, the formation might be split by more than one leader if the communication is incomplete.

Furthermore, the error from the odometer produced, for instance, by slide and skid, is ignored in our work. In the GPS absent environment, the odometer error will affect the flocking. In the future work, we will try to adjust the location information in relation to a certain obstacles such as the wall and the gap.

The main contribution of the paper is in proposing a topology system to identify a large scale multi-robot group in a discretionary dimension space. This topology system is easy to implement and impacts in multi-robot control research. For a large number of robots flocking, robot's ID should include abundant information to identify the specific robot. When the ID is changed, the role of the robot can be changed simultaneously. With an increased scale of formation, the advantage of the topologic method is more apparent, as it has reduced calculating cost. Furthermore, we use ICE to build the communication environment.

Most of the algorithms and strategies here might be applicable in the unmanned battle field in the future. The chariots can organize the attacking or defending formations and avoid the enemy's firepower automatically. It can be also useful in the detection of objects in high risk environment.

ACKNOWLEDGMENTS

Lei Liu is financially funded by Chinese Scholarship Council for his exchange study in Loughborough University.

REFERENCES

Balch, T., R.C. Arkin (1998). Behavior-based Formation Control for Multi-robot Teams. In: *IEEE Transactions on Robotics and Automation*, **vol. 14**, **no. 6**, pp. 926-939.

Stroupe, A.W., T. Balch (2003). Behavior-Based Mapping and Tracking with Multi-Robot Teams Using Probabilistic Techniques. In: *Proceedings of the 2003 IEEE International Conference on Robotics and Automation (ICRA '03)*.

Shao, J., L. Wang and G. Xie (2006). Flexible Formation Control for Obstacle Avoidance Based on Numerical Flow Field. In: *Proceedings of the 45th IEEE conference on Decision & Control*, pp. 5986-5991.

Das, A.K., R. Fierro, V. Kumar, etc. (2002). A Vision-Based Formation Control Framework. In: *IEEE Transactions on Robotics and Automation*, **vol.18**, **no. 5**, pp. 813-825.

Consolini, L., F. Morbidi, D. Prattichizzo and M. Tosques (2007). A Geometric Characterization of Leader-Follower Formation Control. In: *IEEE international Conference on Robotics and Automation*, pp. 2397-2402.

Tanner, H.G., G.J. Pappas and V. Kumar (2004). Leader-to-Formation Stability. In: *IEEE Transactions on Robotics and Automation*, **vol. 20**, **no. 3**, pp. 443-455.

Desai, J.P., J.P. Ostrowski, and V. Kumar (2001). Modeling and control of formations of nonholonomic mobile robots. In: *IEEE Transactions on Robotics and Automation*, vol. 17, no. 6, pp. 905–908.

Desai, J.P. (1998). Motion Planning and Control of Cooperative Robotic Systems. *University of Pennsylvania, Ph.D. Thesis*.

Lemay, M., F. Michaud, D. Letourneau and J.M. Valin (2004). Autonomous Initialization of Robot Formations. In: *Proceedings of the 2004 IEEE International Conference on Robotics & Automation*, pp. 3018-3023.

Tanner, H.G. (2007). Switched UAV-UGV Cooperation Scheme for Target Detection. In: *Proceedings of the 2007 IEEE International Conference on Robotics and Automation*, pp. 3457-3462.

Saripalli, S., D.J. Naffin and G.S. Sukhtme (2002). Autonomous Flying Vehicle Research at the University of Southern California. In: *Proceedings of the First International Workshop on Multi-Robot Systems*, pp. 73-82.

Bikdash, M., F. Hadaegh, D. Scharf, and S. Ploen (2004). Three-dimensional analysis of basic formation initialization algorithms in deep space. In: *Proceedings of the 2004 American Control Conference*, pp. 3447-3453.

Hughes, E.J. (2002). Multi-Objective Evolutionary Guidance for Swarms. In: *Proceedings of the Evolutionary Computation on 2002, Proceedings of the 2002 Congress*, pp. 1127-1132.

Makarenko, A., A. Brooks and T. Kaupp (2006). Orca: Components for Robotics. In: *International Conference on Intelligent Robots and Systems (IROS'06) Workshop on Robotic Standardization*.

Fredslund, J., M.J. Matari (2002). A General Algorithm for Robot Formations Using Local Sensing and Minimal Communication. In: *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 837-846.

http://www.zeroc.com/

http://orca-robotics.sourceforge.net/orca/orca_doc_quickstart_explained.html

Henning, M., M. Spruiell (2007). Distributed Programming with Ice. Revision 3.2.1.

Sarid, S., A. Shapiro and Y. Gabriely (2007). MRBUG: A Competitive Multi-Robot Path Finding Algorithm. In: *Proceedings of 2007 IEEE International Conference on Robotics and Automation*, pp. 877-882.