# Integration of Planning and Control in Robotic Formations

**V.T. Ngo, A.D. Nguyen,** and **Q.P. Ha**
ARC Centre of Excellence for Autonomous Systems,
Faculty of Engineering,
University of Technology, Sydney,
PO Box 123 Broadway NSW 2007 AUSTRALIA

**Abstract**

This paper is devoted to planning and control of a pattern formation of mobile robots when moving between goal points in a known and static environment. Path planning is performed for a reference point in the formation using the modified A* search, coupled with a proposed smoothing technique to generate a feasible trajectory with nonholonomic constraints of mobile robots taken into account. Based on this reference trajectory and the predefined formation configuration in curvilinear coordinates, each robot in the formation computes its trajectory. Formation motion control is then integrated in the proposed framework to derive velocity profiles for robots in the group, taking into account differential geometry of the trajectories. Obstacle avoidance is guaranteed by varying the coordinates of those robots that are likely in collision with obstacles relative to the reference one. Simulation results are presented to illustrate the validity of the proposed framework.

Key words – robotic formation, modified A* search, curvilinear coordinates, obstacle avoidance.

## 1. Introduction

The problem of control and coordination of multi-robot systems has received a considerable interest recently as various applications can be performed faster and more efficiently with multiple robots than with a single robot. In many cases, multi-robot systems are much more robust and fault tolerant and can be easily expanded to a large scale. Some typical applications are moving large objects [Donald *et al.*, 2000] , exploration [Fox *et al.*, 2000], surveillance [Feddema and Schoenwald, 2001], search and rescue [Jennings *et al.*, 1997]. In this context, the control of robotic formations is particularly important in such applications as mine sweeping [Balch and Arkin, 1998; Healey, 2001], military scout and agricultural coverage tasks , where sensor assets are limited as it allows each robot in the formation to concentrate its sensing capability on a portion of the environment, while other robots in the formation cover the rest.

Research in robotic formations has focused on issues like formation generation [Arai *et al.*, 1989; Yamaguchia and Arai, 1994], maintenance of a formation shape [Balch and Arkin, 1998; Desai *et al.*, 2001], controlling and changing formations [Das *et al.*, 2002; Desai *et al.*, 1999; Nguyen *et al.*, 2004]. Generally, there are three broad approaches to the robotic formation problem available in the literature. They include the combined reactive behaviours [Balch and Arkin, 1998], leader–follower strategies [Desai *et al*., 1999; Desai *et al*., 2001], and virtual structures [Jongusuk and Mita, 2001; Lewis and Tan, 1997]. A comprehensive review of robotic formation is given in [Erkin *et al.*, 2003].

Our research objective is to integrate the path planning and control in moving towards a framework for the control and coordination of a group of mobile robots. Toward this goal this paper proposes to combine path planning and trajectory generation for the control of multiple robots in a given pattern, stressing on the formation dynamic behaviour, in terms of velocity profiles, particularly when turning**.** In this work, inspired by [Barfoot and Clark, 2004], we assume the availability of a grid cell map of an environment and use a reference point in the formation as the starting point to plan the path for the formation. This reference may be the center of the formation, one particular robot in the group, or any other point. Smoothing techniques are then applied to acquire the shorter, less turns, and appropriate turning radius path. Based on this path, the velocity profile is obtained for the reference point. The coordination between the robots in maintaining the formation is guaranteed using the motion planning as proposed in [Barfoot and Clark, 2004]. Changing the formation shape to fulfil a specific task or to deal with obstacle collision is acquired by planning the offsets from the reference trajectory in a curvilinear coordinates for each robot.

The rationale for the integration of planning and control in robotic formations is stated in [Ngo *et al.*, 2005], where a generic architecture is proposed for robotic formations moving in a static environment. In this work, the coordination of the mobile robot group is implemented in curvilinear coordinates, which allows for maintaining formation shape with possibilities to adjust the formation width or to change the formation shape with some concession made when the formation turns**.** The idea behind the proposed mechanisms for planning and control of a robotic formation can be illustrated in a flowchart shown in Figure 1. After a reference point has been chosen, the modified A* search is performed to find a path for the whole formation. If a path is found in this step, it is optimal subject to the defined heuristics and the path for the reference point is always safe. The path found is smoothed out to reduce the number of turns and to satisfy the dynamic and kinematic constraints of mobile robots. A reference trajectory is then generated for the reference point. Based on the formation configuration, and the reference trajectory, offsets for each robot in curvilinear coordinates are computed and the trajectories for all the robots in the formation are then obtained. Next, each robot performs its motion according with its planned velocity profile until the goal is reached.

The paper is organised as follows. In section 2, path planning and smoothing are presented. The trajectory generation is discussed in Section 3 and the coordination strategy to avoid obstacles and inter–robot collision in Section 4. Simulation results are provided in section 5. A conclusion and future work are given in Section 6.

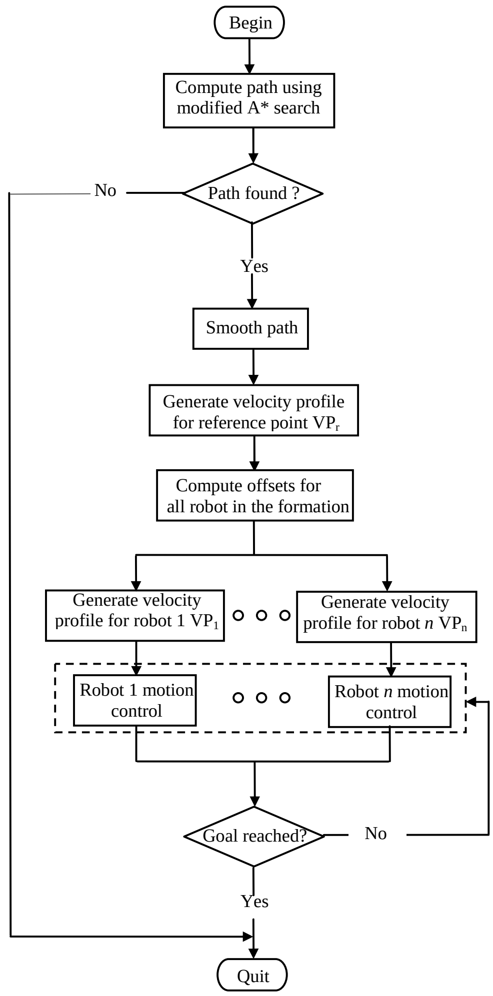



Figure 1. Flow chart of formation planning and control algorithm

## 2. Path Planing and Smoothing

### 2.1 Path Planning

A grid map is assumed to be available at this stage. Each cell is a node in the search process. The optimal path from the initial point to the goal point can be found using various standard graph search methods. Amongst them, the A* method is frequently employed to search the free space for an optimal path. As the A* method may be computationally-inefficient, the so-called modified A* search, can be used to lessen the computational burden involved. This method results in a "loose search", which is illustrated in Figure 2 for $k$=4. Accordingly, when a node is expanded, its children nodes are attached not adjacently but in $k$ cells away, where $k$ is planning step.

The vector approach is used to check the visibility of from the expanded node to its children nodes, i.e. to check whether a straight line from the expanded node to a child node intersects any obstacle. The vector approach is also used to check the visibility between the goal and the nodes with a minimum cost being opened. If the goal is visible then the search process is terminated and the safe path is found. A drawback of this method is that a safe path may not be found even there exists one. An alternation is to reduce the planning step $k$ if no path is found with a specific $k$. When $k$ is reduced to 1, the method becomes the traditional A* search.

In this paper, the modified A* search [Warren, 1993] is used to find the path for reference point in the formation. Assume that the center point in the formation is chosen as the reference point, and then even if the planning step $k$ is chosen to be equal or greater than half of the formation width, it does not guarantee that the formation can traverse to the goal without colliding with any obstacle due to the smoothing process, and the nature of generating a child node in the modified $A^*$ search method.

Here the cost function used to determine the optimal path is defined as $f = g + h$, where $g$ is the actual cost from the initial position to the current node, and $h$ is the heuristic cost from the current node to the goal defined as:

$$h = d(x,y) + c, \tag{1}$$

in which $d(x, y)$ is the distance from the current node to the goal and $c$ is the terrain cost of the current node. To guarantee a sufficient distance from the path to obstacles, those nodes within a set range of a known obstacle is assigned a higher terrain cost. Obviously, this parameter decreases with distance.

The vector approach is chosen here for checking the visibility between the node to be opened and the goal as the process involves a shorter searching time, can deal with the problem of the goal node not lying on "loose grid" [Warren, 1993], and helps eliminate the redundant way-points in the remainder of the path. This also helps reduce the post processing time to smooth the path, as will be detailed in Section 2.2.

As stated, this paper is dealing with only the case of known and static environments. However, it is recognised that normally in order to deploy a robotic formation, the environment should not be too cluttered and that the robots can only reach the goal with a predefined configuration provided the environment around the goal is obstacle–free. These observations together with the advantage of a reduced time for path searching and post processing with the modified A* search enable the online replanning requirement for robotic formations operating in dynamic environments.

### 2.2. Path Smoothing

The common kinematic model for a non-holonomic three-wheel robot is described as:

$$\begin{aligned} \dot{x}(t) &= v\cos\theta \\ \dot{y}(t) &= v\sin\theta \\ \dot{\theta}(t) &= \omega, \end{aligned} \tag{2}$$

where $x(t)$ and $y(t)$ denote the position of the centre point on the wheel axis, $\theta(t)$ represents the orientation, and inputs $v$ and $\omega = vK$ are the current translational and angular velocities respectively, K is the curvature of the trajectory the robot is following. The mobile robots have three degrees of freedom, i.e. two positional degrees of freedom $(x, y)$ and one orientational $\theta$, which are related to a non-holonomic constraint equation implied in the model:

$$-\dot{x}\sin\theta + \dot{y}\cos\theta = 0\,. \tag{3}$$

To ease with the navigation and subject to the vehicles' kinematic and dynamic constraints, the paths resulted from the search process need to be smoothed out, using the technique proposed as follows. As the A* algorithm searches the eight nodes around the current one, and then proceeds to the next node, it produces a zigzag path. This zigzag path is unacceptable for mobile robots especially when moving in a formation. An ideal differentially driven robot can turn on the spot (spin on wheels), but this does not

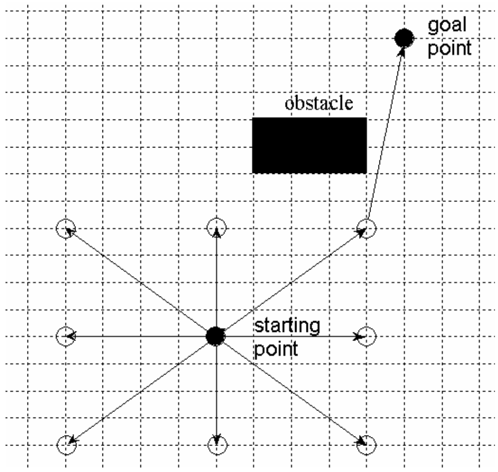



Figure 2. Modified A* search in 2D with planning step $k$=4

imply that the formation can turn on the spot while maintaining the geometric formation. Here the "line of sight" principle is applied to smooth out the path obtained from the A* search. This is summarised in Algorithm 1 below.

*Algorithm 1*

1. Choose the first path node as the start node and its successive node as the end node. Record the first path node as the new way-point.
2. Check the visibility between the start node and the end node.
3. If they are not visible to each other, record the immediate precedent of the end node as a new way-point, and choose this node as the start node with its successive as the end node. Go to step 2.
4. If they are visible to each other, then
   4.1 If the end node is not the goal, choose the node successive to the end node as the new end node. Go to step 2.
   4.2 If the end node is the goal, record it and exit.

The resulted way-points of the path after applying Algorithm 1 are shown in Figure 3 in comparison with the modified A* search path. As noted in [Barfoot and Clark, 2004], the square, rectangular, or even a wedge formation can only be maintained perfectly (including the position and orientation of the robots) in a straight line motion. When the formation turns, a concession must be made, i.e., the formation is maintained in curvilinear coordinates rather than in the rectilinear coordinate system. Algorithm 1 results in the path with smallest number of turns. As can be seen, the resulted path is smoother but still contains sharp corners. Technically, the robot can stop at those points and rotate on its wheels until it reaches the desired orientation and then moves on but this approach is time-consuming and energy-inefficient, especially for a formation, and as noted above, this turning on the spot does not maintain the formation perfectly. Consequently there is a need for further smoothing as suggested in the proposed second algorithm where the turning radius is taken into consideration. This is explained in the following.

Given a quadruple of the start point, the robot orientation at the start point, the end point and the required orientation at this point, there are 2 possible shorted paths for the start point to reach the end point with a fixed starting orientation and arbitrary ending orientation, and 4 possible shorted paths with fixed starting and ending orientations as illustrated in Figure 4. In Case 1, the path between any two successive way-points consists of an arc followed by a straight line. The robot orientation at the end point will be the orientation for the next way-point. In Case 2, the path between any two successive way-points consists of an arc, followed by a straight line, and then by an arc.

If the translational velocity of the robots along the path is assumed to remain constant then the minimum turning radius is determined by the maximum rotational velocity of the robot. Furthermore, if the orientation of the formation is required to be fixed only at the starting and goal positions while may be arbitrary at other way-points along the path, then Case 2 can be applied only for finding the path between way-points $(n-1)^{th}$ and $n^{th}$ (the goal), where $n$ is the way-point number for the path resulted from Algorithm 1. Case 1 is applied for the rest pairs of successive way-points. This is summarised in Algorithm 2 below.

*Algorithm 2*

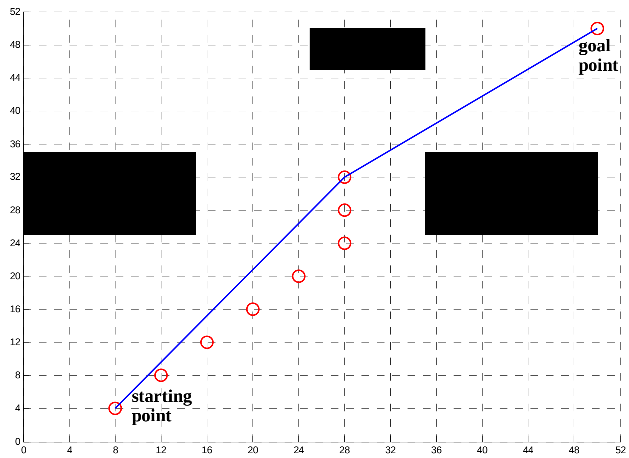
1. Record the first way-point (start position), given a predetermined turning radius. For any pair of successive way-points between the first and the $(n-1)^{th}$ way-points, perform the following step
   - 1.1 Calculate the two possible shorted paths using Case 1.
   - 1.2 If no possible paths exist or the existing paths collide with an obstacle, reduce the turning radius. Go to Step 1.1.
   - 1.3 If there exists at least one possible path without obstacle collision, choose the shortest one (if there are two). Record the necessary data and go to step 1.1.
2. For the path between the $(n-1)^{th}$ and $n^{th}$ way-points, perform the same steps as 1.1 to 1.3, except that calculate the four possible shorted paths using Case 2. Record the necessary data and exit.

## 3. Formation Motion Control

### 3.1. Reference Trajectory Generation

After running the modified A* search and Algorithm 1 and 2, the necessary data are available. For example, with a path between two successive way-points consisting of an arc followed by a straight line, the necessary data for trajectory generation include the length of the arc, the centre of the corresponding circle, the turning radius of the arc, its starting position on the circle, the length of the straight line, its starting position, and the orientation of the straight line.

Once these data are available, the robot position and orientation at a particular time, i.e. the reference trajectory $(x_r(t), y_r(t), \theta_r(t))$ can be easily calculated. One alternation is to obtain the reference trajectory in the form of velocity profile $(v_r(t), \omega_r(t))$ or $(v_r(t), K_r(t))$. As the smooth path obtained from the proposed algorithm has been checked as collision-free, the resulting trajectory is safe for the reference point of the formation.

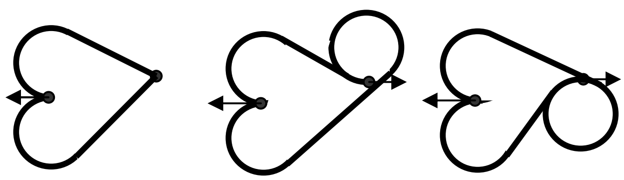



Way-points with modified A* search

Straight lines connecting smooth way–points obtained from Algorithm 1

Obstacle space

Figure 3. Smooth paths obtained from proposed Algorithms



(a) (b)

Figure 4. Possible shortest paths considering turning radius

(a). Case 1: Two options with fixed starting orientation and arbitrary ending orientation
(b). Case 2: Four options with fixed starting and ending orientation

### 3.2. Velocity profiles

Based on the reference point chosen and the predefined geometric formation, each individual robot $i$ in the group has predetermined offsets $[p_i, q_i]^T$ in the curvilinear coordinates relative to the reference point C as shown in Figure 5.

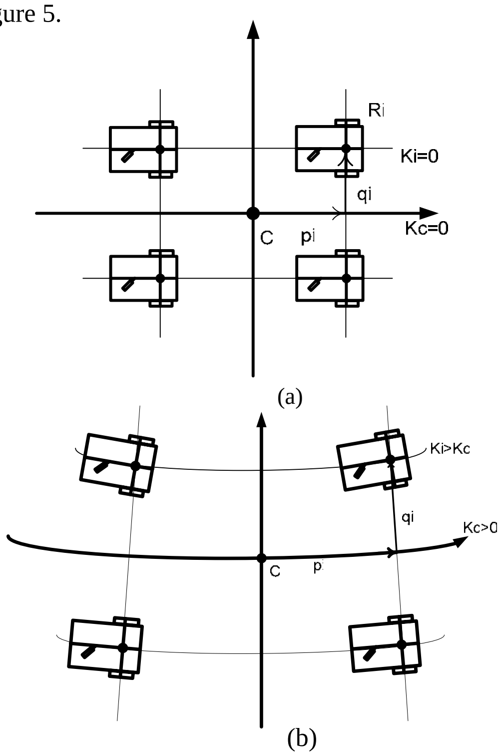



Figure 5. Square formation in a straight line motion (a) and while moving (b).

Once the $[p_i \ q_i]^T$ coordinates of robot $i$ have been determined, then its translational velocity $v_i$ and the curvature $K_i$ are obtained as proposed by [Barfoot and Clark, 2004] as follow.

For convenience, the velocity profile of the reference point C as a function of time, $t$, can be rewritten as a function of distance, $d_r$:

$$v_r(d_r), \quad K_r(d_r),$$

where $d_r(t) = \int_0^t v_r(\tau)d\tau$ . (4)

The distance travelled by robot $i$ along the reference trajectory is

$$s_i(t) = d_r(t) + p_i(t), \tag{5}$$

where it is noted that $[p_i \ q_i]^T$ is function of time. The trajectory of robot $i$ is computed as the following

$$\begin{aligned} v_i(s_i) &= SQv_r(s_i) \\ \omega_i(s_i) &= v_i(s_i)K_i. \end{aligned} \tag{6}$$

where

$$\begin{aligned} S &= \operatorname{sgn}(1 - q_i K_r(s_i)) \\ Q &= \sqrt{\left(\frac{dq_i}{ds_i}\right)^2 + \left(1 - q_i K_r(s_i)\right)^2} \\ K_i &= \frac{S}{Q}\left( K_r(s_i) + \frac{\left(1 - q_i K_r(s_i)\right)\dfrac{d^2 q_i}{ds_i^2} + K_r(s_i)\left(\dfrac{dq_i}{ds_i}\right)^2}{Q^2} \right), \end{aligned} \tag{7}$$

in which $\frac{dq_i}{ds_i}$ and $\frac{d^2 q_i}{ds_i^2}$ are the first and second derivative of $q_i$ with respect to $s_i$, respectively. With this method, a square formation would look like Figure 5b while turning. When the $[p_i \ q_i]^T$ coordinates are constant, equation (6) is simply as

$$\begin{aligned} v_i(s_i) &= v_r(s_i)\left(1 - q_i K_r(s_i)\right) \\ \omega_i(s_i) &= v_i(s_i)\frac{K_r(s_i)}{\left(1 - q_i K_r(s_i)\right)}. \end{aligned} \tag{8}$$

## 4. Obstacle Avoidance

As noted previously in Section 2, the collision-free trajectory for the reference point does not always guarantee the safety for the whole formation. Our strategy is to change the trajectory of those robots which are likely to collide with obstacles. However, in some worse cases, e.g. the width of the path is too narrow to allow for more than one robot, the formation must be changed to a column. While avoiding the static obstacles, the robots also need to avoid collision with other mobile objects, e.g. other robots of the formation. When specific robots need to change their trajectories, the one with the highest priority is planned first. The trajectories for robots of lower priority are planned in accordance with of those of a higher priority.

With the formation planning method presented in section 3, if the formation is static, i.e., $[p_i \ q_i]^T$ coordinates of each robot are constant, the individual robot trajectories will not collide provided that the formation does not turn sharper than a threshold curvature . When the formation needs to be changed from one shape to another shape or narrow its width to accommodate the task, the $[p_i \ q_i]^T$ coordinates of each robot will be planned as functions of time or distance. It is noted from formulae (6) and (7) that, the shape or width of the formation can only be changed if the second derivative $\frac{d^2 q_i}{ds_i^2}$ exists, i.e., offset $q_i$ must be adequately smooth with respect to the corresponding trajectory during the transient from one configuration to another. In this paper, when the formation width needs to be narrower or larger by an amount $\Delta q_i = q_{i,f} - q_{i,0}$ over an incremental distance $\Delta s_i = s_{i,f} - s_{i,0}$, $q_i$ is chosen to take the form:

$$q_i(s_i) = \begin{cases} q_{i,0} & s_i \le s_{i,0} \\ q_{i,0} + \Delta q_i \left( \dfrac{s_i - s_{i,0}}{\Delta s_i} \right)^2 \left( 3 - 2\dfrac{s_i - s_{i,0}}{\Delta s_i} \right) & s_{i,0} < s_i \le s_{i,f} \\ q_{i,f} & s_i > s_{i,f} \end{cases}.$$

With this trajectory of $q_i$, the width of the formation will be narrower if $|q_{i,0}| > |q_{i,f}|$, larger if $|q_{i,0}| < |q_{i,f}|$, and become a column if $q_{i,f} = 0$. It is also noted that, when the formation changes to a column, two points having a same $p_i$ coordinate with respect to the reference point will apparently become the same point whose distance is $p_i$ from the reference point, which may be problematic. To overcome this undesired case for the motion planing method used, the $p_i$ coordinates of those robots need to be adjusted so that those robots will not collide with each other and the column shape can be formed. This is accomplished by decreasing or increasing the velocity of each robot in appropriation while they still follow the same trajectory.

## 5. Simulation results

In this section, the proposed framework is illustrated in two examples where the formation of a wedge type needs to change its configuration to avoid obstacles. The reference point is chosen to coincide with robot $R_1$ and this robot is designated as the leader as the motions of the other two robots ($R_2$ and $R_3$) are based on the trajectory of this leader with $R_2$ has higher priority than $R_3$.

Figure 6 shows that three robots traversing in a desired wedge have to form into a column when moving through a narrow corridor. The $x(t)$ and $y(t)$ trajectories of the three robots are shown respectively in Figure7 and Figure 8.

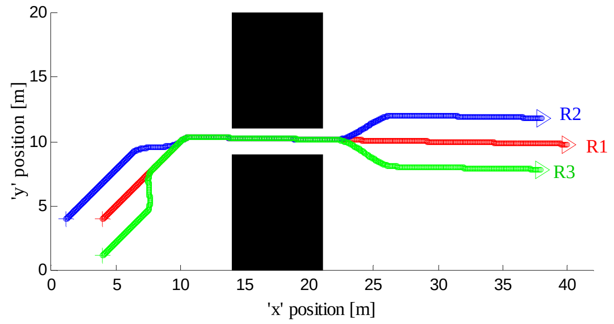



Figure 6. Paths for three robots moving in a wedge, then a column and then back to a wedge.

Typical snapshots of the formation are recorded over time as shown in Figure 9. The results demonstrate the capability of the three robots in moving from an initial position to reach the goal while maintaining the formation shape (a wedge) and changing it accordingly to an environment using the proposed architecture.

In the second scenario, as can be seen in Figure 10, the formation just needs to narrow its width to go through a larger corridor. Again, Figure 11 and Figure 12 show respectively the time trajectories $x(t)$ and $y(t)$ for the three robots, while the formation snapshots over time are presented in Figure 13. The results indicate the proposed framework can handle well situations when the formation shape does not need to be changed, but only needs to adjust the distance between the robots to meet the requirement.

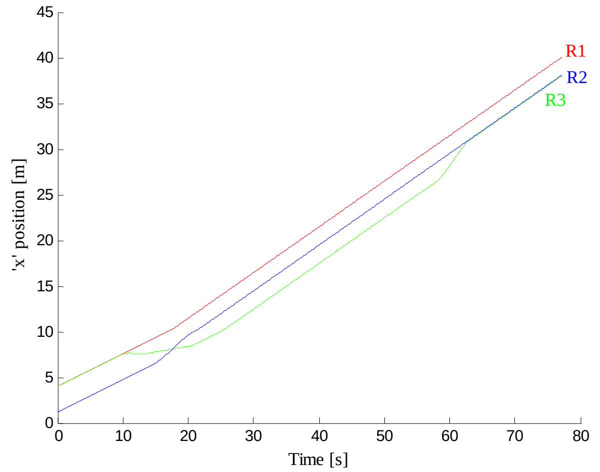



Figure 7. Time trajectories $x(t)$ for three robots in the first example.

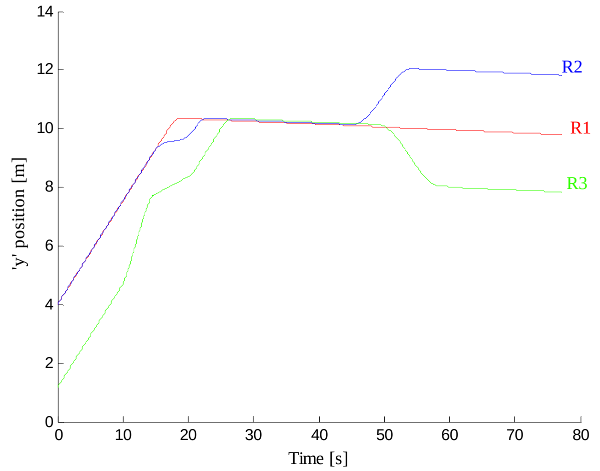



Figure 8. Time trajectories $y(t)$ for three robots in the first example.

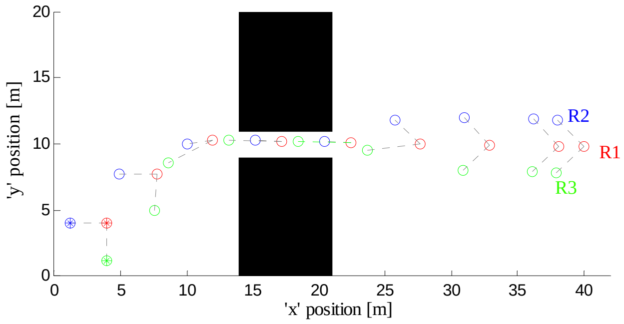



Figure 9. Snapshots over time for the formation in the first example

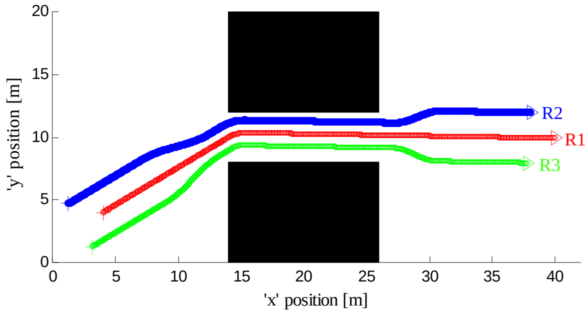



Figure 10. Paths for three robots in a wedge when passing a corridor with only F2 adjusting its trajectory.

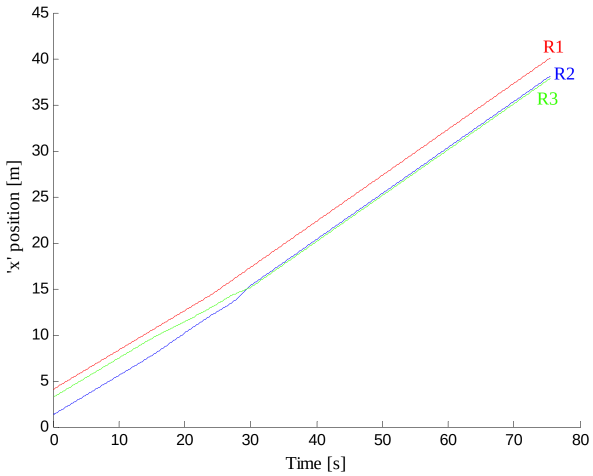



Figure 11. Time responses $x(t)$ for three robots with only F2 changing its trajectory

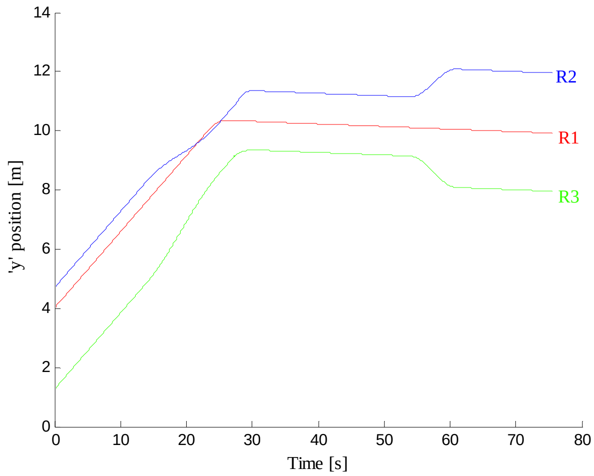



Figure 12. Time responses $y(t)$ for three robots with only F2 changing its trajectory

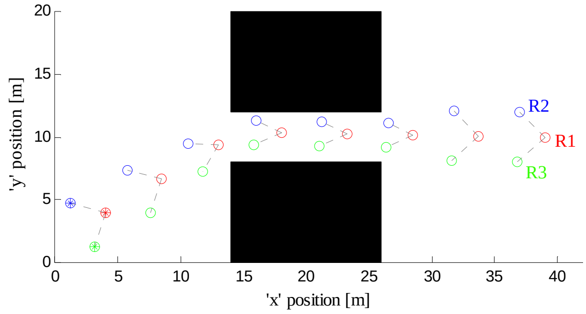



Figure 13. Formation snapshots over time in the second example

## 6. Conclusion and future work

We have presented an efficient framework for planning and control a robotic formation moving in a static environment. The contribution of this paper includes (i) fast and feasible path finding, which is achieved by using the modified A* search and the vector approach coupled with the two proposed smoothing algorithms, taking into account the kinematic and dynamic constraints of mobile robots, and (ii) the maintenance and changing of formations, which is done in curvilinear coordinates to accomplish the required tasks while formation safety is concerned. Illustrative simulation for a three-robot wedge was performed for two scenarios. The results obtained together with the advantage in fast path searching and post processing suggest a possibility of extending the work toward a generic architecture for robotic formation control in dynamic environments. This will be the topic of our future work.

## Acknowledgement

This work is supported by the Vietnam Ministry of Education and Training and also by the ARC Centre of Excellence programme, funded by the Australian Research Council (ARC) and the New South Wales State Government.

## References


[Arai *et al.*, 1989] T. Arai, H. Ogata, and T. Suzuki, "Collision avoidance among multiple robots using virtual impedance," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Tsukuba, Japan, pp. 479-485, 1989.

[Balch and Arkin, 1998] T. Balch and R. Arkin, "Behavior- based formation control for multi-robotic teams," *IEEE Transaction on Robotics and Automation*, vol. 4, pp. 926-934, 1998.

[Barfoot and Clark, 2004] T. D. Barfoot and C. M. Clark, "Motion planning for formations of mobile robots," *Journal of Robotics and Autonomous Systems*, vol. 46, pp. 65-78, 2004.

[Das *et al.*, 2002] A. K. Das, R. Fierro, V. Kumar, J. P. Ostrowski, J. Spletzer, and C. J. Taylor, "A vision-based formation control framework," *IEEE Transaction on Robotics and Automation*, vol. 18, 2002.

[Desai *et al.*, 1999] J. P. Desai, J. P. Ostrawski, and V. Kumar, "Control of changes in formation for a team of mobile robots," *Proceedings of the IEEE International Conference on Robotics and Automation*, Detroit, MI, pp. 1556-1561, 1999.

[Desai *et al.*, 2001] J. P. Desai, J. P. Ostrawski, and V. Kumar, "Modelling and control of formation of nonholonomic mobile robots," *IEEE Transaction on Robotics and Automation*, vol. 17, pp. 905-908, 2001.

[Donald *et al.*, 2000] B. Donald, L. Gariepy, and D. Rus, "Distributed manipulation of multiple objects using ropes," *Proceedings of the IEEE International Conference on Robotics and Automation*, San Francisco, CA, pp. 450-457, 2000.

[Erkin *et al.*, 2003.] B. Erkin, S. Onur, and S. Erol, "A review: Pattern formation and adaptation in multi-robot systems," Robotics Institute - Carnegie Mellon University, Pittsburgh CMU-RI-TR-03-43, 2003.

[Feddema and Schoenwald, 2001] J. Feddema and D. Schoenwald, "Decentralized control of cooperative robotic vehicles," *Proceedings of the SPIE*, Orlando, Florida, pp. 1-11, 2001.

[Fox *et al.*, 2000] D. Fox, W. Burgard, H. Kruppa, and S. Thrun, "A probabilistic approach to collaborative multi-robot localization," *Autonomous Robots*, vol. 8, pp. 325-344, 2000.

[Healey, 2001] A. J. Healey, "Application of formation control for multi-vehicle robotic minesweeping," *Proceedings of the The 40th IEEE Conference on Decision and Control*, Orlando, Florida, USA, pp. 1497-1502, 2001.

[Jennings *et al.*, 1997] J. S. Jennings, G. Whelan, and W. F. Evans, "Cooperative search and rescue with a team of mobile robots," *Proceedings of the IEEE International Conference of Advanced Robotics*, Monterey, CA, pp. 192-200, 1997.

[Jongusuk and Mita, 2001] J. Jongusuk and T. Mita, "Tracking control of multiple mobile robots: A case study of inter - robot collision free problem," *Proceedings of the IEEE International Conference on Robotics and Automation*, Seoul, Korea, pp. 2885-2890, 2001.

[Lewis and Tan, 1997] M. A. Lewis and K. H. Tan, "High precision formation control of mobile robot using virtual structures," *Autonomous Robots*, vol. 4, pp. 387-403, 1997.

[Ngo *et al.*, 2005] V. T. Ngo, A. D. Nguyen, and Q. P. Ha, "Toward a generic architecture for robotic formations: Planning and control," *Proceedings of the Sixth International Conference on Intelligent Technologies*, Phuket, Thailand, 2005.

[Nguyen *et al.*, 2004] A. D. Nguyen, Q. P. Ha, S. Huang, and H. Trinh, "Observer-based decentralised approach to robotic formation control," *Proceedings of the 2004 Australian Conference on Robotics and Automation*, Canberra, Australia, pp. 1-8, 2004.

[Warren, 1993] C. W. Warren, "Fast path planning using modified a* method," *Proc. IEEE International Conference on Robotics and Automation*, Atlanta, Georgia, USA, pp. 662-667, 1993.

[Yamaguchia and Arai, 1994] H. Yamaguchia and T. Arai, "Distributed and autonomous control method for generating shape of multiple mobile robot group," *Proc. IEEE Int. Conf. on Robotics and Automation*, San Diego, CA, USA, pp. 800-807, 1994.