The 2009 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 11-15, 2009 St. Louis, USA

Discrete Event Systems based Formation Control Framework to Coordinate Multiple Nonholonomic Mobile Robots

Gayan W. Gamage, George K. I. Mann and Raymond G. Gosine

Abstract— **This paper describes a leader-follower based formation control framework to coordinate multiple nonholonomic mobile robots. The proposed strategy deploys a control theoretic bottom-up approach where, continuous controllers are coordinated by a supervisory controlled discrete event system. All the mobile robots are required to navigate in an obstacle populated environment. And the followers keep a predetermined geometric formation with the leader while being adaptable to the constraints imposed by obstacles on the environment. The low level control is achieved by a set of behavior based controller with a high-level discrete event system that manages the dynamic interaction with the external environment. The use of discrete event systems reflects a modular manageable system with the potential for scalability and reusability. The proposed system is implemented through simulation and the results are shown to verify its operation.**

Index Terms - **nonholonomic mobile robots, multi-robot formation, hybrid control, nonlinear control, discrete event systems (DES)**

I. INTRODUCTION

Multi robot systems research has drawn an unprecedented focus over the last few years. The collective nature of performing a task by many agents increases effectiveness, efficiency and fault tolerance. It has a wide spectrum of practical applications including cooperative transportation of large objects [1],[2], Exploration, Surveillance, formation control [3],[4],[5], cooperative attack and rendezvous, intelligent highways and air traffic control. In all of these applications the ability of multi-robot systems to robustly function in and interact with the complex environments defines the ultimate usability of the system in a real world. Although the problem of dynamic interaction had been addressed for single robots [6],[7], the multi-robot dynamic interaction with the external world still remains an open research area. The traditional control theory fails in the face of dynamic changes due to it's fixed single mode of operation. Thus it highlights the need of a higher-level coordination protocol to handle the switching of the single modes of control theoretic operations. In behavior based robotics[8] the subsumption architecture and schema theory provides a similar higher-level platform to design reactive behaviors to occupy dynamic changes in the environment through behavior arbitration or behavior combination. The supervisory control of discrete event systems (DES) [9],[10], is an alternative design paradigm especially catered to model the dynamic and synchronous changes of a system. The dynamic interactions are modelled as events, which are controllable and uncontrollable in nature, for e.g: in robot navigation, detecting an obstacle is an uncontrollable event where as avoiding the obstacle is a controllable event. The supervisory control in the discrete event system exploits this controllability feature of events, to enable or disable them in such a way that the system robustly interacts with the dynamic environment [11].

This paper addresses the problem of multi-robot navigation in an unstructured environment with a leader-follower strategy. A designated leader robot and a set of it's followers are required to navigate in an unstructured environment. In addition, the follower robots are required to keep a predetermined geometric formation with the leader while relaxing the formation constraints in the face of obstacles and walls. Feedback linearization [12],[13],[14] is used to build controllers for both the leader and follower types of mobile robots to occupy different elementary, (ex: obstacle avoidance) as well as secondary, (ex: formation control with obstacle avoidance) behaviors. And these controllers are coordinated by a supervisory controlled discrete event system. Earlier approaches to leader-follower formation control with navigation is reported in [15],[4]. The basic formation controller developed in [4] is used with modifications for single robot navigation and formation control in this context where as [4] uses it only for formation control. In addition a new set of feedback linearized controllers are developed for follower robot navigation. The use of DES is also exploited in this context to coordinate these controllers where as [4] uses only a gross formation switching system which lacks modelling ease, reusability and scalability.

Our contributions are two fold. Firstly formation-control strategy based feedback linearized controllers are developed for both leader and follower robots. These include controllers for elementary behaviors, (ex: obstacle avoidance) and controllers for executing combined-behaviors, (e.g: wall following with goal navigation). Some elementary behaviors for e.g: formation control, can be combined with wall following or obstacle avoidance by relaxing some formation-constraints. This in effect minimize the chattering effects of switching of the DES model. The second contribution being the use of supervisory control of discrete event systems to model the coordination control of

This research is supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) and Memorial University of Newfoundland, Canada

G. Gamage, G. Mann and R. Gosine are with the Faculty of Engineering & Applied Science, Memorial University of Newfoundland, A1B3X5, St. John's, NL, Canada. {gayan,gmann,rgosine}@engr.mun.ca

the above behavior based controllers.

The paper is organized as follows, Section II explains the development of low level controllers for the mobile robots. Section III explains the supervisory control of discrete event systems modelling for both leader and follower robot navigation with some simulation results followed by the conclusion in section IV.

II. Control Algorithms

The lead robot is capable of dragging the formation along with him subjected to constraints imposed by obstacles and walls and other robots on the environment. The formation positions for followers are defined by a relative distance and a relative bearing from the lead robot. Follower robot dynamics and leader dynamics are treated in isolation when building low level continuous controllers. The i^{th} robot can be described by,

$$\dot{x}_i = v_i \cos\theta_i \quad \dot{y}_i = v_i \sin\theta_i \quad \dot{\theta}_i = \omega_i \tag{1}$$

where $(x_i, y_i, \theta_i) \in SE(2)$ while v_i and ω_i are the linear and angular velocities of the i^{th} unicycle robot respectively. The approximate linearization of the above system at any point is clearly not controllable [16]. Hence, a linear controller cannot achieve posture stabilization, not even locally with approximate linearization [14]. But the accessibility rank condition [16] of the above system being globally satisfied proves that the system is controllable in a nonlinear sense. Hence nonlinear feedback-linearization [12] can be applied to (1). Since the decoupling matrix of (1) is singular, it can be made non-singular through changing the output measurement (x_i, y_i) to an offset from the current measurement coordinates. It results in,

$$\begin{pmatrix} x_i^{nw} \\ y_i^{nw} \end{pmatrix} = \begin{pmatrix} \cos\theta_i & -\sin\theta_i \\ \sin\theta_i & \cos\theta_i \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} + \begin{pmatrix} x_i \\ y_i \end{pmatrix} \tag{2}$$

a and b are offsets from the *origin* of the robot-coordinate system in X_R and Y_R directions respectively. And (x_i, y_i) are the current output measurement coordinates in the global-coordinate system while (x_i^{nw}, y_i^{nw}) are the newest output measurement coordinates in the global-coordinate system. Differentiation of the newer output coordinates with respect to time, results in a nonsingular dynamic system which is readily controllable.

$$\begin{pmatrix} \dot{x}_i^{nw} \\ \dot{y}_i^{nw} \\ \dot{\theta}_i \end{pmatrix} = \begin{pmatrix} \cos\theta_i & -a\sin\theta_i - b\cos\theta_i \\ \sin\theta_i & a\cos\theta_i - b\sin\theta_i \\ 0 & 1 \end{pmatrix} \begin{pmatrix} V_i \\ \omega_i \end{pmatrix} \tag{3}$$

A. Control Algorithms for leader robot

The lead-robot is capable of navigating along a given set of way points until the final goal is reached. Way point based navigation is achieved by a proportional feedback linearized controller considering the dynamics of (3). We also derive controllers for wall following, obstacle avoidance, goal navigation with obstacle avoidance for the leader robot.

1) Goal navigation: Once the desired sub-goal locations are given, input-output feedback linearization [12] of (3) gives,

$$\begin{pmatrix} V_l \\ \omega_l \end{pmatrix} = \begin{pmatrix} \cos\theta_l & -a\sin\theta_l \\ \sin\theta_l & a\cos\theta_l \end{pmatrix}^{-1} \begin{pmatrix} c_1(x_l^d - x_l) \\ c_2(y_l^d - y_l) \end{pmatrix} \tag{4}$$

Here, the lead-robot is to be driven to a desired sub goal location $(x_l^d, y_l^d) \in R^2$ without any desired θ_l. For simplicity, it is made $b = 0$ in (3) and taken a as the only offset, from where the current robot-position measurements are taken. $(x_l, y_l, \theta_l) \in SE(2)$ is the leader's current position in the newest measured coordinates with an offset of a. From here on, (x, y) refers to the measured coordinates with an offset of a. $v_l : (v_{max} \geq v_l \geq 0)$ is the linear and $\| \omega_l \| \leq W_{max}$ is the angular velocity of the leader robot while c_1 and c_2 > 0 are user defined constants. It is easily seen that, by applying control input as in (4) the system (3) converge exponentially to $(x_l^d, y_l^d) \in R^2$.

2) Formation controller: The lead-robot's obstacle avoidance and wall following controllers are dependent on the basic leader-follower formation controller developed in [4]. The state output measurements (x_s, y_s) of the follower robot are taken from an offset of a from it's origin along the X_R in the robot coordinate system as in (3). The Robot system

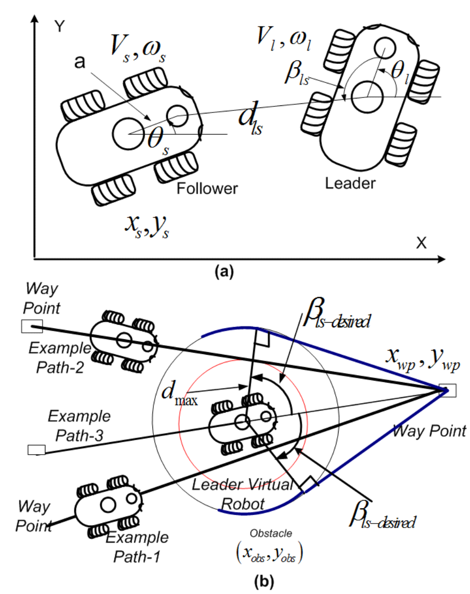

Fig. 1. (a) basic leader follower formation and (b) leader obstacle avoidance with sub-goal navigation

in Fig. 1 (a) is transformed to a new set of coordinates, where the lead-robot state is considered as an exogenous input to the system. Follower formation can be described by the relative distance d_{ls} and the relative bearing β_{ls} from the leader robot and the difference of relative orientation $\theta_l - \theta_s$. The kinematics are given as,

$$\dot{z}_{ls} = G_1(z_{ls}, \theta_{ls})\mathbf{u_s} + F_1(z_{ls})\mathbf{u_l}, \qquad \dot{\theta}_{ls} = \omega_l - \omega_s \tag{5}$$

where $z_{ls} = [d_{ls} \quad \beta_{ls}]^T$ is the system output and $\theta_{ls} = \theta_l - \theta_s$ is the relative orientation between the leader and follower.

$u_l = [v_l \quad \omega_l]$ is the exogenous input by the leader robot to the system while $u_s = [v_s \quad \omega_s]$ is the follower's driving inputs.

$$G_1 = \begin{pmatrix} \cos\gamma_{ls} & a\sin\gamma_{ls} \\ \frac{-\sin\gamma_{ls}}{d_{ls}} & \frac{a\cos\gamma_{ls}}{d_{ls}} \end{pmatrix}, \qquad F_1 = \begin{pmatrix} -\cos\beta_{ls} & 0 \\ \frac{\sin\beta_{ls}}{d_{ls}} & -1 \end{pmatrix}$$

where $\gamma_{ls} = \theta_{ls} + \beta_{ls}$ and a is the offset we described earlier and $\beta_{ls} = -\theta_l + \pi + atan2(y_l - y_s, x_l - x_s)$. By applying nonlinear feedback linearization the inputs of the follower robot are given by,

$$\mathbf{u_s} = G_1^{-1}(k(z_{ls}^d - z_{ls}) - F_1\mathbf{u_l}) \tag{6}$$

$k = [k_1 \quad k_2]^T > 0$ are the controller gains, while $z_{ls}^d = [d_{ls}^d \quad \beta_{ls}^d]^T$ are the desired relative distance and bearing of the follower robot from the leader robot. It has been proved in [4] the system outputs $[d_{ls} \quad \beta_{ls}]$ exponentially converges to the desired values and $\| \theta_{ls} \| \leq \delta$ for small $\delta \geq 0$ as $t \to \infty$.

3) Formation controller for obstacle avoidance: Formation controller (6) can be used to avoid obstacles while navigating to a goal location. The obstacle is considered as a virtual lead-robot (fig.1 (b)), whose heading is in the direction of the next sub goal $\alpha_{head} = atan2(y_{wp} - y_{obs}, x_{wp} - x_{obs})$. Also the obstacle has been extended to a circle of radius d_{max}(maximum turning radius of the robot). The formation controller (6) can be applied to avoid obstacles by taking (fig.1 (a)) follower as the actual leader-robot and the leader in (fig.1 (a)) as the obstacle for this context. Once a real lead-robot approaches the obstacle boundary of d_{max}, the controller (6) is used to drive it with a desired $z_{ls}^d = [d_{max} \quad \beta_{ls}^d]^T$ where,

$$\beta_{ls}^d = \cos^{-1}\frac{d_{max}}{\sqrt{((y_{wp} - y_{obs})^2 + (x_{wp} - x_{obs})^2)}} \tag{7}$$

The control law makes the robot keep a constant d_{max} distance from the obstacle and once the robot arrives at β_{ls}^d, it can safely return to goal navigation. For static obstacle avoidance, the virtual leader's exogenous inputs to the system is taken as zero. And If the obstacle has a motion, the exogenous inputs can be estimated by a decentralized state estimation as in [4]. Otherwise they are taken as zero. The controller (6) has a singularity when the initial relative bearing is $\pm\pi$. Hence, when the robot arrives as in Example path 3 of (fig.1 (b)), a hysteresis is added to move the robot away from the $\pm\pi$ bearing in either of the directions.

a) Clustered obstacle avoidance: Clustered obstacles can be identified as a set of overlapped obstacles as in (fig.2 (a)). While *obstacle-1* in (fig.2 (a)) is being avoided by the above strategy, the robot comes to $\mathbf{P_1}$ and identifies a second obstacle. To avoid *obstacle-2*, the robot keeps d_{max} (maximum turning radius of the robot) distance from the obstacle and tries to go in the *shortest path* to the way point with the above strategy. The problem arises, when the shortest path around the obstacle overlaps with the the previous visited obstacles. Then the heading of the virtual leader robot of *obstacle-2* is changed to as heading along the straight line connecting the two obstacles from *obstacle-1* to 2, which is $\theta_h = atan2(y_{obs}^2 - y_{obs}^1, x_{obs}^2 - x_{obs}^1)$. And drive the real-robot along the longest path of obstacle avoidance to just above $\mathbf{P_2}$, which is $\mathbf{P_3}$. Once $\mathbf{P_2}$ is passed we can switch to the earlier obstacle avoidance strategy. The β_{ls}^d for point $\mathbf{P_2}$ can be calculated as, $\beta_{ls}^d = \pi + atan2(y_{way}^2 - y_{obs}^2, x_{way}^2 - x_{obs}^2) - \theta_h$. θ_h is the orientation angle of the leader robot.

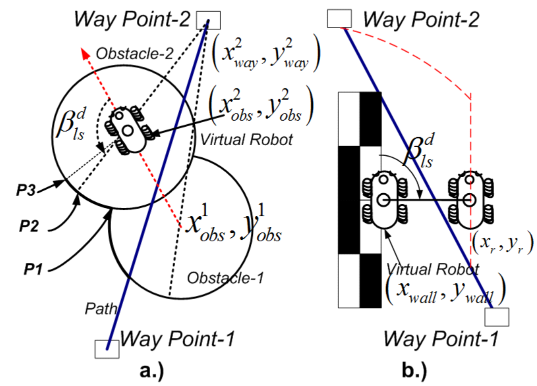

Fig. 2. (a) clustered obstacle avoidance (b) wall following

4) Formation controller for wall following: Once a wall is detected, if a virtual leader is made to slide along the wall with a fixed velocity, the real robot can be made to follow it with a constant relative distance of d_{wall}^{max} and an angle of β_{ls} as in (fig.2 (b)). Assuming the distance sensors are fixed on the front of the robot, if a wall is detected on the left distance sensors, the heading of the virtual leader robot is taken as pointing towards the right-most scans of the wall from the left-most. If an obstacle is detected on the right distance sensors, the vice versa. Moreover for a left most scan $\beta_{ls}^d = -\frac{\pi}{2}$ while for right most scans $\beta_{ls}^d = \frac{\pi}{2}$. The location on the wall where the shortest scan distance recorded initially is taken as the virtual robot's position (x_{wall}, y_{wall}). And if a wall is detected dead on the front of the real robot, the earlier obstacle avoidance is activated.

B. Control Algorithms for followers

The followers of the system keeps a tight formation with the leader by generating motion commands through (6). But once obstacles or walls are encountered all of the formation constraints can no longer be met at the same time. Hence keeping a desired relative bearing is relaxed while still keeping a desired distance from the leader.

1) Obstacle avoidance with formation control: Obstacle avoidance and wall following is achieved through a three-robot formation structure. One being the real leader, another the follower and the other being the obstacle or the wall. The kinematics for obstacle avoidance while keeping a desired distance (fig.3 (a)) with the leader is given as,

$$\dot{z}_{dll} = G(\gamma_{13}, \gamma_{23}, a)\mathbf{u_3} + F_1(\beta_{23})\mathbf{u_2} + F_2(\beta_{13})\mathbf{u_1} \tag{8}$$

where $z_{dll} = [d_{13} \quad d_{23}]^T$ is the system output. $u_1 = [v_1 \quad \omega_1]$ is one exogenous input by the real-leader robot to the system and if the obstacle's motion parameters can be estimated, $u_2 = [v_2 \quad \omega_2]$ can be used as another exogenous input to the system. $u_3 = [v_3 \quad \omega_3]$ is the real-follower's driving inputs. d_{13} & d_{23} are the relative distances from the real-leader and

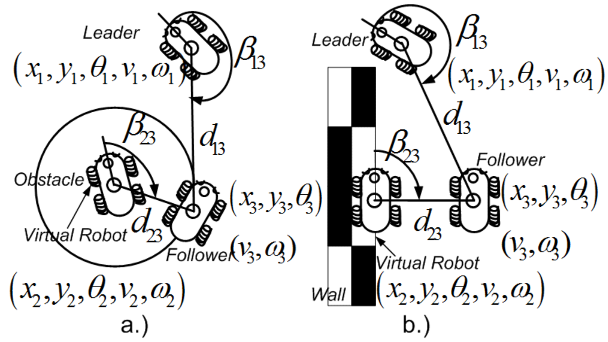

Fig. 3. (a) obstacle avoidance with formation control (b) wall following with formation control

the obstacle to the follower respectively.

$$G = \begin{pmatrix} \cos\gamma_{13} & a\sin\gamma_{13} \\ \cos\gamma_{23} & a\sin\gamma_{23} \end{pmatrix}, \qquad F_2 = \begin{pmatrix} 0 & 0 \\ -\cos\beta_{23} & 0 \end{pmatrix}$$

$$F_1 = \begin{pmatrix} -\cos\beta_{13} & 0 \\ 0 & 0 \end{pmatrix}, \quad where \quad \begin{aligned} \gamma_{13} &= \beta_{13} + \theta_{13} \\ \gamma_{23} &= \beta_{23} + \theta_{23} \end{aligned}$$

Also $\theta_{13} = \theta_1 - \theta_3$, $\theta_{23} = \theta_2 - \theta_3$. β_{13} and β_{23} can be calculated as in the basic formation controller in (5). Through nonlinear feedback linearizarion, motion commands for the follower is,

$$\mathbf{u_3} = G^{-1}(c(z^d_{dll} - z_{dll}) - F_1\mathbf{u_l} - F_2\mathbf{u_2}) \tag{9}$$

$c = [c_1 \quad c_2]^T > 0$ being controller gains, while $z^d_{dll} = [d^d_{13} \quad d^d_{23}]^T$ are the desired relative distances from the leader and the obstacle. It is seen that the closed loop system is stable and converges to z^d_{dll} arbitrarily fast.

2) Wall following with formation control: Wall following with formation (fig.3 (b) is performed similar to the previous case of obstacle avoidance with the addition of keeping β_{23} at a desired value. Depending on the heading of the wall it can be $\pm\frac{\pi}{2}$. Also we make ω_2 of the virtual leader zero, such that it can only slide along the heading of the wall with a v_2 only. Kinetics of the system is,

$$\dot{z}_{wll} = G(z_{wll}, \beta_{23}, \gamma_{13}, \gamma_{23}, a, d_{23})\mathbf{u} + F(\beta_{13})\mathbf{u_1} \tag{10}$$

$$G, \begin{pmatrix} \cos\gamma_{13} & a\sin\gamma_{13} & 0 \\ \cos\gamma_{23} & a\sin\gamma_{23} & -\cos\beta_{23} \\ \frac{-\sin\gamma_{23}}{d_{23}} & \frac{a\cos\gamma_{23}}{d_{23}} & \frac{\sin\beta_{23}}{d_{23}} \end{pmatrix} F, \begin{pmatrix} -\cos\beta_{13} & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$$

Also $z_{wll} = [d_{13} \quad d_{23} \quad \beta_{23}]^T$ is the system output. $u = [v_1 \quad \omega_1]$ is the only exogenous input by the real-leader. And $u = [v_3 \quad \omega_3 \quad v_2]^T$ are follower's inputs followed by the virtual leader's linear velocity. Through feedback linearization,

$$\mathbf{u} = G^{-1}(c(z^d_{wll} - z_{wll}) - F\mathbf{u_1}) \tag{11}$$

$c = [c_1 \quad c_2]^T > 0$ being controller gains, while $z^d_{wll} = [d^d_{13} \quad d^d_{23} \quad \beta^d_{23}]^T$ are the desired settings of the system. The location on the wall where, the initial wall detection scans received the shortest distance is taken as the virtual robot's starting pose. And by subsequent usage of (11) we derive motion commands for the virtual leader on the wall (v_2) and the follower $[v_3 \quad \omega_3]^T$ to follow the wall. Also the virtual leader is stopped where the wall ends, to switch to some other navigation task. It is seen that the closed loop system (11) is stable and converges to z^d_{wll} arbitrarily fast.

III. Discrete Event Systems Modelling

The action coordinations of the leader and follower robots are formulated by discrete event systems with supervisory control. Continuous dynamics models developed above are taken as controllable action events of the DES models. And any other constraints specified will be handled by modelled supervisors. The primitive DES systems for a leader robot and for follower robots can be described by (fig.4). We also

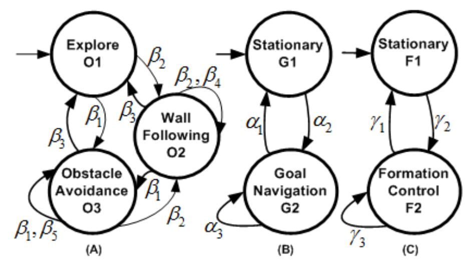

Fig. 4. DES models for primitive behaviors. (A) obstacle avoidance and wall following, (B) goal navigation behavior, (C) formation control behavior

assume that the robot obstacle avoidance and wall following can not be active at the same time. In such cases where there is a wall and obstacle both to tackle, the precedence is given to obstacle avoidance while the contact point of walls are considered as obstacles.

A. DES model for obstacle avoidance & wall following

Set of states $Q : \{Explore - O_1, Wall\ following - O_2, Obstacle\ avoidance - O_3\}$
Set of events $\Sigma : \{detect\ obstacle - \beta_1, detect\ wall - \beta_2, detect\ freespace - \beta_3, move(wallFollow) - \beta_4, move(obstacleAvoidance) - \beta_5\}$
supervisory controllable events $\Sigma_c = \{\beta_4, \beta_5\}$

B. DES model for Goal navigation behavior

Set of states $Q : \{Stationary - G_1, Goal\ navigation - G_2\}$
Set of events $\Sigma : \{goal\ reached - \alpha_1, goal\ computed - \alpha_2, move(to\ goal) - \alpha_3\}$
supervisory controllable events $\Sigma_c = \{\alpha_2, \alpha_3\}$

C. DES model for Formation control behavior

Set of states $Q : \{Stationary - F_1, Formation\ control - F_2\}$
Set of events $\Sigma : \{leader\ lost - \gamma_1, leader\ detected - \gamma_2, keep\ formation - \gamma_3\}$
supervisory controllable events $\Sigma_c = \{\gamma_3\}$

D. Leader-Robot Navigation

The leader robot is to be navigated to the end goal via pre-calculated sub goals (through the use of A* algorithm in a voronoi decomposed map) while avoiding obstacles and following walls. In order to develop the holistic leader-robot navigation system, the primitive obstacle avoidance and goal navigation DES models in (fig.4) above are combined together using parallel composition. For the DES model in (fig.5), the controllable events are $\{\alpha_2, \alpha_3, \beta_4, \beta_5\}$ and the supervisor developed to enable or disable these controllable events is given below. '1' stands for enabling, '0' stands for disabling and 'x' stands for not caring the given controllable event. In states O_2G_1 & O_3G_1, the event goal computed-α_2 is disabled since the goal computation happens when there is no obstacles or walls near the robot. Wall avoidance-β_4 is enabled in both O_2G_1 & O_2G_2 and the continuous dynamics of wall following procedure described in section II.A.4 is applied to follow the walls. since β_4 is enabled in O_2G_2, the event move to goal-α_3 is disabled to make sure that no two controllable events exist in a single state. For this system we have not defined a combined control methodology for wall following and goal navigation. Hence O_2G_2 degenerates to a control of wall following only. In state O_3G_1, the event of pure obstacle avoidance-β_5 is handled by a reactive obstacle avoidance procedure as in [6]. But in O_3G_2 both events α_3 , β_5 are enabled and a new event is introduced to combine both goal navigation with obstacle avoidance described by the dynamics model given in section II.A.3.

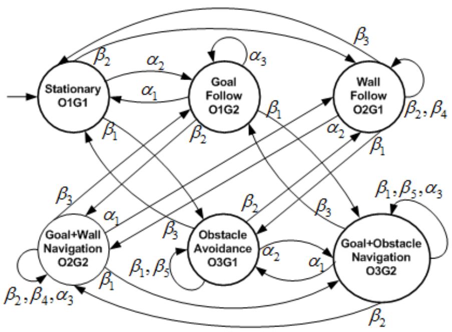

Fig. 5. Leader robot DES model

State	O_1G_1	O_1G_2	O_2G_1	O_2G_2	O_3G_1	O_3G_2
Σ_c	xxxx	xxxx	0x1x	x01x	0xx1	x1x1

TABLE I

$\Sigma_c = \{\alpha_2, \alpha_3, \beta_4, \beta_5\}$

1) Leader robot simulations: Through the use of a path planning algorithm, we find a way point based path for a given map and the robot is driven along these way points as shown in (fig.6). Green dotted lines are path segments. They sometimes overlap with walls and obstacles. Hence the wall following and obstacle avoidance procedures explained above were used along with the respective DES model. Through the many experiments run, it's clear that the low level controllers, managed by the supervisory control of discrete event system is successful in navigating the leader robot to the final goal. It is also observed that the chattering effect is minimized due to the introduction of combined behavior controllers ex: goal navigation with obstacle avoidance. In the simulations, we only focussed on static obstacle avoidance. And those static obstacles were avoided in the shortest path possible, to the next way point. Cluttered obstacles were also successfully evaded without any significant chattering effect. The wall following procedure leads to a more systematic way of following the wall, again minimizing the chattering effects which would not have been possible with most of the earlier wall following strategies. All the simulations were carried out in the matlab environment.

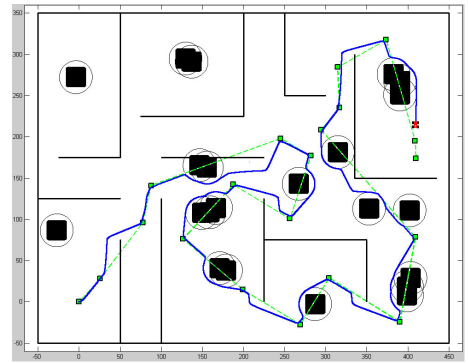

Fig. 6. Leader robot simulation in an office layout with walls and obstacles: Blue path is the actual robot path, green squares are way points, black squares marked with boundaries of black circles are obstacles while black lines are walls

E. Multiple follower-Robots Coordination

Through the parallel composition of the elementary discrete event systems of obstacle avoidance and formation control, a new complex DES model is built as shown in (fig.7). The followers of the lead robot follow their leader while avoiding obstacles and following walls. The supervisor to control the follower DES model is given as below,

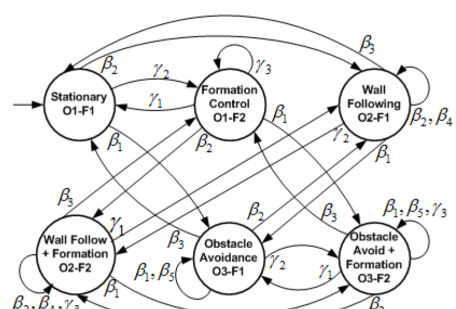

Fig. 7. Follower robots DES model

In state O_1F_2, the event formation control-γ_3 is enabled since it's a state of pure formation control and the leader based formation dynamics developed in (6) is used to follow the leader in a given geometric formation. The event wall follow-β_4 is enabled in O_2F_1 to follow the walls when and the

State	O_1F_1	O_1F_2	O_2F_1	O_2F_2	O_3F_1	O_3F_2
Σ_c	xxx	1xx	x1x	11x	xx1	1x1

TABLE II
$\Sigma_c = \{\gamma_3, \beta_4, \beta_5\}$

robot is near a wall and the communication to the leader robot is lost. Hence the individual wall following procedure of section II.A.4 is applied to follow the walls. But in O_2F_2, the state where both the wall following and formation control becomes active events γ_3 & β_4 are enabled to introduce a new event which incorporates both wall following and formation keeping actions and the dynamics of section II.B.2 are used to handle that event. In state O_3F_1 the event obstacle avoidance-β_5 is triggered and the reactive obstacle avoidance procedure in [6] is again applied to avoid only the obstacles. In O_3F_2, both obstacle avoidance and formation control actions become active and the supervisor enables both 'formation keep' and 'obstacle avoidance' events and introduce a new event, which combines both the actions in to one continuous dynamics model given in section II.B.1 above.

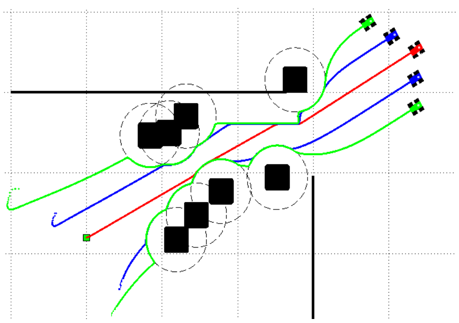

Fig. 8. Multi Robot Simulation: Red robot is the leader, blue and green robots are it's followers

1) simulations of the complete system: The leader robot is navigated as in the previous simulation, while the followers were coordinated and run to geometric formations by the DES model for the follower robots with the respective low level controllers. The system was tested with different geometric shapes of wedge, diamond, horizontal lines and triangular shapes with arbitrary starting points for the followers. Obstacle avoidance and wall following while keeping a desired distance to the leader was also tested for different shapes cited above (fig.8). We observe that as long as the leader robot does not make sudden rotations, the system manages to avoid obstacles or follow the walls effectively. The constraint ω_{max} makes sure these sudden manoeuvres for the leader does not occur. Also, once the followers sense any sudden maneuvers, they depend on their own controls till the leader stabilizes. Chattering effect is again minimized due to the introduction of combined behaviors. For these simulations, the exogenous inputs needed by the followers from their leader, are communicated every time to make their controls work.

IV. CONCLUSIONS

The contributions of this paper is the formulation of a leader-follower based control theoretic framework to coordinate multiple nonholonomic mobile robots. Low level continuous feedback-linearized control algorithms were developed to handle elementary and combined behaviors for robot navigation. And a higher level supervisory controlled discrete event systems provides a modular framework to coordinate the actions of the robots in a dynamic environment. The proposed system is implemented through simulations to validate its usability. It is seen from simulations, that the use of low level controllers for combined behaviors minimize the chattering effect. Also the underlying supervisory controlled discrete event system provides ease of modelling, scalability, modularity and reusability. Thus new behaviors can be added without much of a hassle. The followers depend on accurate measurements of leader's pose, velocity and on its own sensed information like odometry. Obtaining such accurate measurements becomes really challenging in a real world robot application, where the delay and uncertainty is prevalent. The simulations suggests the controllers can withstand small noise variations but are susceptible for large ones. Nevertheless the system remains robust under small noises and has the potential to scalability and reusability.

REFERENCES

[1] D. Rus, B. Donald, and J.Jennings, "Moving furniture with a team of autonomous robots," *in Proc. IEEE/RSJ IEEE. Int. Conf. Intelligent Robots and Systems*, pp. 556–561, August 1995.

[2] M. Mataric, M. Nilsson, and K. Simsarian, "Cooperative multi-robot box pushing," *in Proc. IEEE/RSJ IEEE. Int. Conf. Intelligent Robots and Systems*, pp. 235–242, August 1995.

[3] T. Balch and R. Arkin, "Behaviour-based formation control for multi-robot systems," *IEEE T. on Robotics and Automation*, vol. 14, no. 6, pp. 926–939, August 1998.

[4] A. Das, R. Fierro, V. Kumar, J. Ostrowski, J. Spletzerm, and C. Taylor, "A vision-based formation control framework," *IEEE T. on Robotics and Automation*, vol. 18, no. 5, pp. 813–825, October 2002.

[5] W. Ren, "Consensus based formation control strategies for multi-vehicle systems," *American Control Conference*, June 2006.

[6] M. Egerstedt and X. Hu, "A hybrid control approach to action coordination for mobile robots," *Automatica*, vol. 38, no. 1, pp. 125–130, 2002.

[7] D. Chwa, "sliding-mode tracking control of nonholonomic wheeled mobile robots in polar coordinates," *IEEE Trans. Contr. Syst. Technol.*, vol. 12, no. 4, pp. 637–644, 2004.

[8] R. C. Arkin, *Behavior based robotics*. MIT Press, 1998.

[9] C. G. Cassandras and S. Lafortune, *Introduction to discrete event systems*. Springer, 1999.

[10] W. M. Wonham and P. J. Ramadge, "Modular supervisory control of discrete-event systems," *Mathematics of Control, Signals and Systems*, vol. 1, pp. 13–30, Feb 1988.

[11] R. Bajcsy, "Discrete event systems for autonomous mobile agents," *Robotics and Autonomous Systems*, vol. 12, pp. 187–198, 1994.

[12] A. Isidori, *Nonlinear Control Systems*. Springer, 1989.

[13] H. K. Khalil, *Nonlinear Systems*. Prentice Hall, 1996.

[14] R. Brockett, R. Millmann, and H. Sussmann, *Differential Geometric Control Theory*. Birkhauser, 1983.

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

[16] G. Oriolo, A. D. Luca, and M. Vendittelli, "Wmr control via dynamic feedback linearization: Design, implementation and experimental validation," *IEEE Trans. Contr. Syst. Technol.*, vol. 10, no. 6, pp. 835–852, 2002.