Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Rigid Formation Keeping and Formation Reconfiguration of Multi-Agent Systems

D.H.A. Maithripala * **S. Jayasuriya** **

* *Dept. of Mechanical Engineering, Texas A& M University, College Station, TX 77843-3123 USA (e-mail: asanka@tamu.edu)*
** *Dept. of Mechanical Engineering, Texas A& M University, College Station, TX 77843-3123 USA (e-mail: sjayasuriya@tamu.edu)*

Abstract: A motion planning algorithm is presented for formation control of coordinating multi agents engaged in rigid formation keeping and formation reconfiguration. The multi agent system is separated into geometrically equivalent subsystems for distributed control. The proposed motion planning algorithm generates reference trajectories for each of these subsystems in real-time for online, distributed and autonomous control. Deriving the constrained kinematics eliminates the need for nonlinear programming to account for the system constraints, making the approach amenable to real-time control. A control strategy accounts for actuator/operating constraints to give dynamically feasible reference trajectories. Explicit consideration of actuator and operating limitations and nonholonomic constraints in the design of the reference trajectories, thereby addressing the important issue of dynamic feasibility, is one of the main contributions of the proposed approach. The motion planning algorithm is verified through simulations for a team of multi-agents moving in and switching between formations in a scouting scenario.

Keywords: Real-time formation control, autonomous, multi-agent systems, constrained kinematics, distributed control.

1. INTRODUCTION

Formation control, which falls under the broader category of coordinated control of multi-agent systems, has received considerable attention over the last few years. In this paper we are interested in the special coordinated motion control where maintenance of a rigid geometric formation and switching between predefined formations are the primary considerations. Rigid formation keeping has applications in scouting (Balch et al. (1998)), formation flight (Giulietti et al. (2000)), cooperative sensing (Agre et al. (2000)) and box pushing (Lewis et al. (1997)). Rigid formation keeping alone can in general be too restrictive for applications in an environment with obstacles and therefore formation reconfiguration becomes important (Desai et al. (2001); Sugihara et al. (1996); Yamaguchi et al. (2001)).

There has been two preferred approaches to formation control; (1) formulate it as a constrained optimization problem, (2) formulate it in the framework of a tracking control problem. The main limiting characteristic of many existing motion planning algorithms utilizing the former approach is the computational complexity (Betts (1998)) where even those proposed for real-time path planning lead to the solution of an optimization problem using nonlinear programming (Milam et al. (2000); Faiz et al. (2001)). On the other hand, most approaches that formulate formation control as a tracking control problem assume that the reference trajectory for the group as a whole is known a priori rather than designed to include individual agent dynamics. For example in many leader following algorithms a feasible trajectory may be designed for only the leader and then each individual is forced to keep up with the leader trajectory as best as possible. This approach generally does not work well for a system having dynamic constraints at each agent level since we are likely attempting to track a trajectory of the system that may be incompatible with some local agent dynamics. (Wang et al. (1991); Desai et al. (1998)) give leader-follower approaches for formation control where the trajectory of the leader is assumed. The concept of virtual structure (VS), introduced in (Lewis et al. (1997)) allowing a group of agents to behave as if they were embedded in a rigid body, is used in (Hu et al. (2001); Young et al. (2001)) for the rigid formation keeping problem. A leader-follower approach is used in (Desai et al. (1999)) for formation changing of nonholonomic robots where the leader-robot is required to follow a given trajectory while the follower robots are responsible for changing the formation. The dynamic constraints of the individual agents are incorporated in the group behavior to a certain degree in (Hu et al. (2001); Young et al. (2001)) where the VS slows down or speeds up along its assumed path depending on how well the formation is maintained. In all the above, the resulting formation tracking error necessarily depends on a desired reference path/trajectory assumed rather than designed for the leader or the VS. An exception is the work presented in (Belta et al. (2001)), designing reference trajectories for the rigid formation keeping problem, that can theoretically result in zero tracking error for the mobile agents maintaining formation. However dynamic constraints are captured only to the extent that the designed reference trajectories will be smooth.

978-1-1234-7890-2/08/$20.00 © 2008 IFAC

10.3182/20080706-5-KR-1001.1144

In order to put in perspective some interesting work in the literature let us now consider a scenario where three robots are maintaining an equally spaced line formation. If all three agents are restricted to have the same speed, they *must* have common velocity directions to maintain this line formation. This corresponds to pure translational motion of the formation line. In fact multi-agents constrained to have the same speed can have only one of two stable formations; parallel motion characterized by common velocity directions of agents (with arbitrary relative spacing) or circular motion characterized by circular orbits of the agents about a common fixed point (Justh et al. (2004); Paley et al. (2005)). For the three agent line formation, a rotation of the line formation, no matter how small, will necessarily demand the agents to have differential speeds. The amount of dynamically feasible rate of rotation of the formation line will be a nonlinear function of the allowable differential speeds and the spacing between agents and will generally be considerably less than the allowable rate of turn of the individual agents. This simple example shows that the dynamic constraints that limit the maneuverability of a single agent can have a magnified effect in limiting the maneuverability of a formation as a whole. Parallel motion of a formation, while escaping this fact, will in general be too restrictive to be useful in practical applications. This is particularly so for example in the box pushing or formation flight scenarios. The issue of real-time trajectory generation under actuator and operating constraints is addressed for a constrained system in (Milam et al. (2000); Faiz et al. (2001)) and in the multi-agent formation control setting in (Dunbar et al. (2004)). However these methods end up solving a constrained optimization problem using nonlinear programming to generate feasible trajectories.

The critical role played by dynamic constraints in formation control problems that do not allow flexibility in their formation constraints has been overlooked in most approaches to formation control. Approaches that do consider dynamic constraints do so by solving a constrained optimization problem using nonlinear programming. One of the main goals of this paper is to advocate a change in paradigm in the approach to formation control that would address the key issues of dynamic feasibility and computational complexity. Dynamic feasibility is especially critical for formation control problems that have little flexibility in their formation constraints. The approach we propose in here is to embed the configuration and dynamic constraints of formation control into the design of reference trajectories to be used simultaneously by the tracking controllers of the individual agents. Theoretically (in the absence of model uncertainty, and external disturbances) this can result in zero tracking error. Explicit consideration of actuator and operating limitations in designing formation trajectories that can ideally result in zero formation error in the tracking control stage and real-time trajectory generation are the key contributions we make. In particular the actuator constraints we consider include lower bounds (with strictly positive bounds) for the individual robot speeds which we believe is essential in aircraft/UAV applications. Real-time motion planning, being model independent, explicit consideration of actuator/operating constraints, distributed/autonomous control and scalability are the main attributes of the motion planning algorithm presented in this paper.

2. APPROACH TO FORMATION CONTROL

This section presents a motion planning algorithm for the formation keeping and formation reconfiguration problems. The proposed scheme is computationally attractive for distributed, online and real-time control.

2.1 Formation Guidance

The proposed approach is on formation guidance as opposed to formation tracking. Formation guidance can be defined as the generation (or design) of reference trajectories to be used as the input for the formation agents' relative state tracking control law. Formation tracking control refers to design techniques and associated stability/performance results for these relative state tracking control laws. Explicitly incorporating the dynamic model, including all dynamic constraints of the agents, in the design of the reference trajectories will ensure zero tracking error in the tracking control stage, at least in theory. We say at least in theory, since this is with idealized assumptions of zero model uncertainty and zero disturbance. In actual implementation, model uncertainty and disturbance rejection will need to be accounted through feedback in the formation tracking stage.

2.2 Agent dynamics

We propose to design reference trajectories that capture the essential agent dynamics and constraints through a simplified dynamic model. For example, the dynamic capabilities of a four wheeled robot having many degrees of freedom and controls can be captured approximately but reasonably well through the much simpler unicycle model. The Unicycle model essentially captures the no slip condition of the wheeled robot while appropriate constraints on its higher level controls of speed and steer can effectively capture the wheeled robot's actuator and operating constraints. The accuracy with which the dynamic models of the individual agents are captured in the design of the reference trajectories will determine the degree of tracking error at the tracking control stage and ultimately in the degree of error in the formation.

2.3 Configuration of a formation

Consider N agents restricted to the plane making up a virtual structure (VS) with O_c being an arbitrary point on this VS (the centroid of the N agents at time t_0 for example). An orthogonal local coordinate frame B is assumed fixed at O_c and we make no distinction between the VS and this B frame. Let $(b_{i,1}, b_{i,2})$ denote the place holder for the i'th agent in this B frame. When $b_{i,1}, b_{i,2}$ are constant the VS will be rigid and when $b_{i,1}, b_{i,2}$ are time varying the VS too will be time varying making the formalism applicable for both the rigid formation keeping and the formation changing problems. Let (x, y) be local coordinates of O_c and ϕ the orientation of the B frame, with respect to an inertial frame I. Let (x_i, y_i) describe the position and θ_i the orientation of the i'th agent with respect to the frame I. Similarly suppose (x, y, θ) describes the position and orientation of a virtual agent at O_c.

2.4 Separation into N subsystems

Distributed control is proposed as the control architecture for the motion planning algorithm and for this purpose we decouple the problem into N subproblems. From a geometric control point of view, this means the configuration and dynamic constraints defining the formation control problem needs to be separated into N geometrically similar sets of constraints. Consider the i'th subsystem made up of the i'th agent, the virtual agent at O_c and the B frame where the vectors $\mathbf{r}_i = (x_i, y_i)$, $\mathbf{r} = (x, y)$ are in the inertial frame I and the vector $\mathbf{b}_i = (b_{i,1}, b_{i,2})$ is in the B frame, as shown in Fig.1. This i'th subsystem has local coordinates $q_i = (x, y, \theta, x_i, y_i, \theta_i, \phi, b_{i,1}, b_{i,2})$.

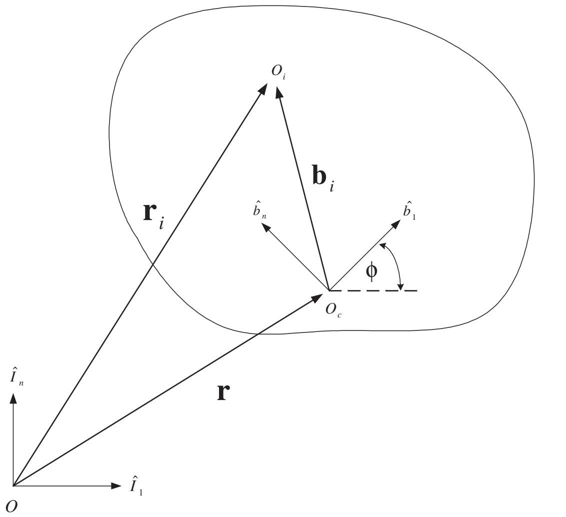

Fig. 1. Configuration of the i'th subsystem

2.5 Constrained Kinematics

Next we derive the constrained kinematics of the i'th subsystem which allows us to develop a real-time motion planning algorithm for the two formation control problems we consider.

The subsystem has the following formation constraints;

$$\begin{aligned} x_i &= x + b_{i,1}cos\phi - b_{i,2}sin\phi \\ y_i &= y + b_{i,1}sin\phi + b_{i,2}cos\phi \end{aligned} \tag{1}$$

Consider the following which is the same as condition Eq.(1); provided that Eq.(1) holds at some time instant (for example, t=0);

$$\begin{aligned} \dot{x}_i &= \dot{x} - \dot{\phi}(b_{i,1}sin\phi + b_{i,2}cos\phi) + \dot{b}_{i,1}cos\phi - \dot{b}_{i,2}sin\phi \\ \dot{y}_i &= \dot{y} + \dot{\phi}(b_{i,1}cos\phi - b_{i,2}sin\phi) + \dot{b}_{i,1}sin\phi + \dot{b}_{i,2}cos\phi \end{aligned} \tag{2}$$

Assume the following virtual controls for the VS;

$$\begin{aligned} \dot{\phi} &= \Omega \\ \dot{b}_{i,1} &= U_{i,1} \\ \dot{b}_{i,2} &= U_{i,2} \end{aligned} \tag{3}$$

We assume that the dynamics of the i'th agent can be captured reasonably well through the nonholonomic constraints of a unicycle making it applicable to wheeled robots (having no slip constraints) and UAVs alike. We also assume the same dynamics for the virtual agent at O_c. These nonholonomic constraints have the following equivalent control forms for the actual and the virtual agent;

$$\begin{aligned} \dot{x} &= V\cos\theta & \dot{x}_i &= V_i\cos\theta_i \\ \dot{y} &= V\sin\theta & \dot{y}_i &= V_i\sin\theta_i \\ \dot{\theta} &= \omega & \dot{\theta}_i &= \omega_i \end{aligned} \tag{4}$$

Agent dynamic constraints due to actuator limitations are explicitly captured through constraints on the kinematic controls V, ω, V_i, ω_i of the above equivalent control form as follows;

$$\begin{aligned} V^{min} \leq V_i, V &\leq V^{max} \\ -\frac{\omega^{max}}{V^{max}}V_i \leq \quad \omega_i &\leq \frac{\omega^{max}}{V^{max}}V_i \\ -\frac{\omega^{max}}{V^{max}}V \leq \quad \omega &\leq \frac{\omega^{max}}{V^{max}}V \end{aligned} \tag{5}$$

The above constraints when $V^{min} > 0$, forces all the agent trajectories to be smooth (C^1) and also in particular allows us to capture the stall speed constraint of fixed winged aircraft. As a preliminary step we will only consider explicit actuator constraints involving the velocities while acknowledging the importance of actuator constraints involving accelerations. Although V, ω are virtual controls of a virtual agent at O_c, we consider the same bounds for these as we do for the i'th agent. This allows us to replace the virtual agent with an actual agent if needed. Being virtual controls we do not consider limits on $\Omega, U_{i,1}, U_{i,2}$.

Let $f : z \to f(z)$ denote a function on z where $z \in \mathbb{R}^m \times \ldots \times \mathbb{R}^m$ and $f(z) \in \mathbb{R}^m$. Equation (1) can then be written in the compact form $f(q_i) = 0$ and consequently (2) will have the compact form $f(\dot{q}_i, q_i) = 0$. Equations Eq.(4) and Eq.(3) can be written as $\dot{q}_i = f(q_i, u_i)$, where $u_i = (V, V_i, \omega, \omega_i, \Omega, U_{i,1}, U_{i,2})$. Actuator constraints given explicitly in Eq.(5), are represented by $u_i \in \Pi_i$. The equations $f(\dot{q}_i, q_i) = 0$ and $\dot{q}_i = f(q_i, u_i)$ yield $F_i(q_i, u_i) = 0$ having the following coordinate form;

$$\begin{aligned} V_i\cos\theta_i &= V\cos\theta - \Omega(b_{i,1}\sin\phi + b_{i,2}\cos\phi) \\ &\quad + U_{i,1}\cos\phi - U_{i,2}\sin\phi \\ V_i\sin\theta_i &= V\sin\theta + \Omega(b_{i,1}\cos\phi - b_{i,2}\sin\phi) \\ &\quad + U_{i,1}\sin\phi + U_{i,2}\cos\phi \end{aligned} \tag{6}$$

Consider a partition of the controls $u_i = \{v_i, w_i\}$ where the dimension of w_i equals the number of equations in $F_i(q_i, u_i) = 0$. Then, *if* the jacobian matrix $\left|\frac{\partial F_i}{\partial w_i}\right| \neq 0$, the implicit function theorem assures us that we can solve $F_i(q_i, u_i) = 0$ for the w_i's in terms of the v_i's. For purposes of control and consensus, we are interested only in the partition $v_i = \{V, \omega, \Omega, U_{i,1}, U_{i,2}\}$ and $w_i = \{V_i, \omega_i\}$ for which $\left|\frac{\partial F_i}{\partial w_i}\right| = 0$ and we are forced to look beyond the kinematics of the problem.

Equation (6) directly yields the following;

$$\begin{aligned} V\sin(\theta-\theta_i) &= (\Omega b_{i,2}-U_{i,1})\sin(\phi-\theta_i) \\ &\quad -(\Omega b_{i,1}+U_{i,2})\cos(\phi-\theta_i) \end{aligned} \tag{7}$$

Taking the derivative of Eq.(7) with respect to time once, along with Eq.(4), Eq.(3) and Eq.(6) gives us the following;

$$\begin{aligned} V_i &= \{V\cos\theta-\Omega(b_{i,1}\sin\phi+b_{i,2}\cos\phi)+U_{i,1}\cos\phi \\ &\quad -U_{i,2}\sin\phi\}\ /\ \{\cos\theta_i\}. \\ \omega_i &= \{V\omega\cos(\theta-\theta_i)+\dot{V}\sin(\theta-\theta_i) \\ &\quad -\left(\Omega^2 b_{i,1}+2\Omega U_{i,2}+\dot{\Omega}b_{i,2}-\dot{U}_{i,1}\right)\sin(\phi-\theta_i) \\ &\quad -\left(\Omega^2 b_{i,2}-2\Omega U_{i,1}-\dot{\Omega}b_{i,1}-\dot{U}_{i,2}\right)\cos(\phi-\theta_i)\} \\ &\quad /\ \{V\cos(\theta-\theta_i)-(\Omega b_{i,2}-U_{i,1})\cos(\phi-\theta_i) \\ &\quad -(\Omega b_{i,1}+U_{i,2})\sin(\phi-\theta_i)\}. \end{aligned} \tag{8}$$

The expression for ω_i in Eq.(8) is the same as Eq.(7) as long as Eq.(7) holds for some time instant. For the constrained i'th subsystem we choose $v_i = \{V, \omega, \Omega, U_{i,1}, U_{i,2}\}$ as the functions we have control over while $w_i = \{V_i, \omega_i\}$ will depend on these functions v_i. The functions V, ω, Ω of v_i will be the ones that couple the N subsystems. Equation (8) can be written in the compact form $w_i = f(q_i, \dot{v}_i, v_i)$ and we can rewrite $\dot{q}_i = f(q_i, u_i)$ as $\dot{q}_i = f(q_i, \dot{v}_i, v_i)$. We call $\dot{q}_i = f(q_i, \dot{v}_i, v_i)$ the *constrained kinematics* of the i'th subsystem since solutions to $\dot{q}_i = f(q_i, \dot{v}_i, v_i)$ satisfy both the formation constraints: $f(q_i) = 0$, and the velocity constraints: $\dot{q}_i = f(q_i, u_i)$. Now we can formally introduce what we mean by feasible solutions of the multi-agent system.

Feasible solutions: A system trajectory (the collection of trajectories for each of the N agents of the system) that satisfies the constrained kinematics: $\dot{q}_i = f(q_i, \dot{v}_i, v_i)$, and actuator/operating constraints: $u_i \in \Pi_i$, for all subsystems $i = 1, \cdots, N$ is defined a feasible solution.

Looking at Eq.(8) and Eq.(7) we see that V_i approaches V and ω_i approaches ω as $\Omega, U_{i,1}, U_{i,2}, \dot{\Omega}, \dot{U}_{i,1}, \dot{U}_{i,2}$ all approach zero (assuming $V \neq 0$). Hence the controls $V \in [V^{min}, V^{max}]$ with $V^{min} > 0$, $\omega \in [-\omega^{max}, \omega^{max}]$, $\Omega = U_{i,1} = U_{i,2} = 0$ satisfy $u_i \in \Pi_i$; $\forall i$, resulting in feasible solutions. These controls correspond to pure translational motion of the VS formation with V and ω being the only active controls. If we suppose that V is held constant, the kinematics of O_c of the VS are exactly that of the Dubins' car. A result due to Dubins then states that O_c of the VS is controllable (Dubins (1957)).

2.6 Distributed Control Strategy

Consider the following control law for $t = [t, t + \delta t]$;

$$\begin{aligned} \tau &= K_\omega(\omega^d-\omega)+\dot{\omega}^d \\ \dot{\Gamma} &= K_\Gamma(\Gamma_a-\Gamma)+\dot{\Gamma}_a \\ \Gamma_a &= K_\Omega(\Omega^d-\Omega)+\dot{\Omega}^d \\ \dot{f} &= K_f(f_a-f)+\dot{f}_a \\ f_a &= K_V(V^d-V)+\dot{V}^d \\ \dot{G}_{(i,j)} &= K_G\left(G_{a,(i,j)}-G_{i,j}\right)+\dot{G}_{a,(i,j)} \\ G_{a,(i,j)} &= K_U\left(U^d_{i,j}-U_{i,j}\right)+\dot{U}^d_{i,j} \end{aligned} \tag{9}$$

where $G_{i,j} = \dot{U}_{i,j},\ \tau = \dot{\omega},\ \Gamma = \dot{\Omega},\ f = \dot{V}$

We develop two sets of functions $V^d, \omega^d, \Omega^d, U^d_{i,j}\ for\ j = 1, 2$ for the control law given above in Eq.(9), resulting in two sets of controllers; one to drive the system towards feasibility and the other to achieve the team goal.

Controls for feasibility: Recall that $V \in [V^{min}, V^{max}]$, $\omega \in [-\omega^{max}, \omega^{max}]$, $\Omega = U_{i,1} = U_{i,2} = 0$ satisfy $u_i \in \Pi_i$ ensuring feasibility of solutions. Hence the control law Eq.(9) with

$$\begin{aligned} \omega^d &= 0 \\ \Omega^d &= 0 \\ V^d &= \frac{V^{min}+V^{max}}{2} \\ U^d_{i,j} &= 0 \end{aligned}$$

exponentially stabilizes $\omega, \Omega, U_{i,j}$ to zero and V to $(V^{min} + V^{max})/2$ driving the system towards feasibility.

Controls to achieve team goal: The control laws to achieve the team task or the team goal depends on the task and the application at hand. For example, in formation flying the team goal might be to form and maintain a rigid formation and possibly fly a trajectory specified by a set of waypoints. For the box pushing problem, the team goal might be to maintain formation and move towards a goal location. For scouting the goal can be moving as a rigid formation through a set of waypoints and to change formation to avoid obstacles in its path. Let us consider the example of a group of mobile robots maintaining and changing formation in a scouting task.

The control law given by Eq.(9) with

$$\begin{aligned} \omega^d &= K_{\beta-\theta}(\beta-\theta)+\dot{\beta} \\ \Omega^d &= K_{\theta-\phi}(\theta-\phi)+\omega \\ V^d &= \begin{cases} V^{min} & if\quad VS\ is\ turning \\ V^{max} & else \end{cases} \\ U^d_{i,j} &= K_b\left(b^d_{i,j}-b_{i,j}\right)+\dot{b}^d_{i,j} \end{aligned}$$

exponentially stabilizes $(\beta-\theta), (\theta-\phi), (b^d_{i,j}-b_{i,j}), (V^d - V)$ to zero where $b^d_{i,j}$ describes the desired VS formation and where $\beta = \arctan\left(\dfrac{y_f-y}{x_f-x}\right)$ with (x_f, y_f) being the desired waypoint of the VS. Then $(\beta-\theta)$ is the angle between the desired waypoint of the VS and its current heading and the objective is to orient the VS formation towards the desired waypoint. The objective of the choice of V^d is to slow down the VS when negotiating a turn and speed up when not. We assume V^d remains either V^{min} or V^{max} for the duration of $[t, t+\delta t]$ ensuring continuity of $\dot{V}^d$.

We propose a distributed control strategy that is implemented in a receding horizon framework. Each agent i in the system solves the constrained kinematics $\dot{q}_i = f(q_i, \dot{v}_i, v_i)$ for the time interval $[t, t+\delta t]$ using either the controls for feasibility or the controls to achieve the team goal, and this is repeated continuously from one time interval to the next. However for consensus, all the agents need to implement the same control law during $[t, t+\delta t]$

thereby generating identical functions $V, \omega, \Omega, U_{i,j}$ with respect to time in each of the subsystems. We assume δt to be fixed. The distributed control and communication strategy is shown in the form of a flow-chart in Fig.(2).

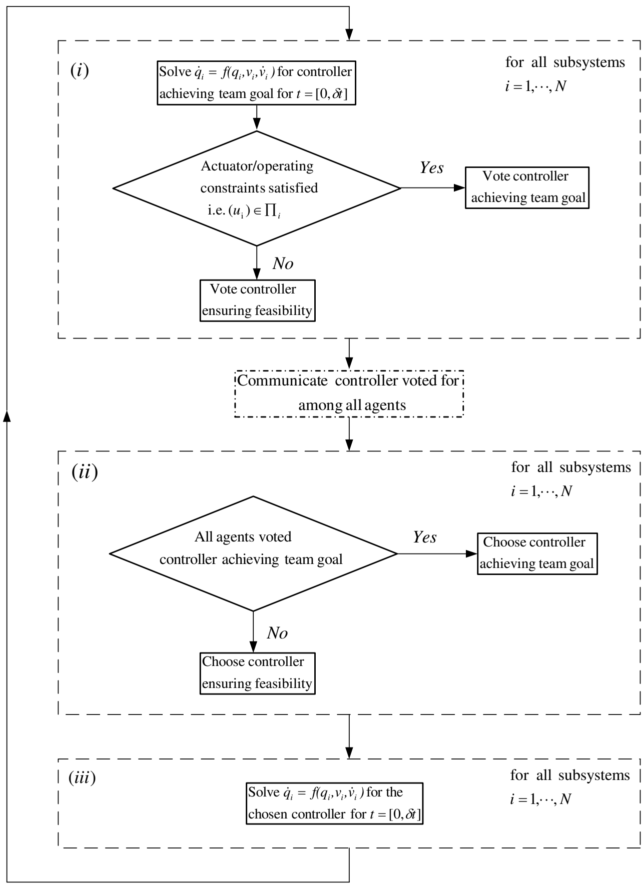

Fig. 2. Distributed control architecture

In each cycle, each of the N agents solves its corresponding constrained kinematics for the time interval $[t, t+\delta t]$ with controls to achieve the team goal first. If all the actuator/operating constraints were satisfied within this time interval, the corresponding agent votes for the controls achieving the team goal. If any of the actuator/operating constraints were violated within $[t, t+\delta t]$, the agent votes for the controls to ensure feasibility. Recall that for consensus, each of the subsystems need to implement the same type of controller during $[t, t+\delta t]$. The type of controller voted for by each agent is communicated amongst all other agents to come to a common decision on the type of controller to use on all the agents. If even one of the N agents had voted for controls for feasibility, then all of the N agents chooses controls for feasibility to solve the constrained dynamics for $[t, t+\delta t]$, to generate their trajectories. If on the other hand, all the N agents had voted for the controller achieving the team goal, then each of the agents computes its trajectory for the time interval $[t, t+\delta t]$ using controls to achieve team goal. Note that the VS trajectory is designed (identically) by each of the N agents in addition to their own trajectory. This is a necessary redundancy in computation in the proposed distributed control strategy.

The computations shown in the blocks (i),(ii) and (iii) of the flow-chart of Fig.2 are performed by each of the N-agents in parallel and as such increasing the number of agents in the system has minimal effect on the overall communication/computation time thus making the approach scalable. Communication amongst the agents need not be continuous and has to occur only once in each cycle of the receding horizon control strategy. The main drawback of this strategy however is that it requires synchronized control and communication among all its agents. We note that the distributed receding horizon control architecture is not technically decentralized, since a globally synchronous implementation requires centralized clock keeping.

3. SIMULATION RESULTS

The proposed motion planning algorithm was simulated on a multi-agent system in a scouting scenario. Here a group of multi-agents are required to maintain formation and move along a path defined by a set of predetermined waypoints. The multi-agents are also required to change formation to avoid obstacles or when a change in the terrain or the task makes it beneficial to maintain a different formation. Actuator constraints of the individual agents were assumed to be $V^{min} = 0.2m/s, V^{max} = 1m/s$ and $\omega^{max} = 2.5rad/s$. Figure.(3) shows simulation results for six agents moving through a given set of waypoints while maintaining and changing between predetermined formations. The simulation results are shown in the form of a series of superimposed snap shots of the coordinated multi agent motion. The controls V_i, ω_i corresponding to higher level controls of "speed" and "steer" for each of the six agents for the above results are shown in Fig.(4). Smooth V_i, ω_i implies the existence of functions f_i, τ_i such that $f_i = m_i \dot{V}_i$ and $\tau_i = J_i \dot{\omega}_i$ giving the dynamics of the i'th agent, where m_i, J_i are its mass and inertia. In general, the computation time of the algorithm was an order of magnitude less than the real-time over which the algorithm was implemented.

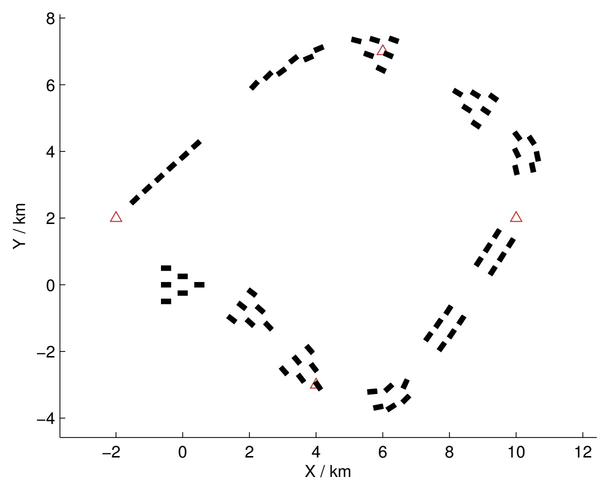

Fig. 3. Formation keeping and reconfiguration motion for six mobile agents

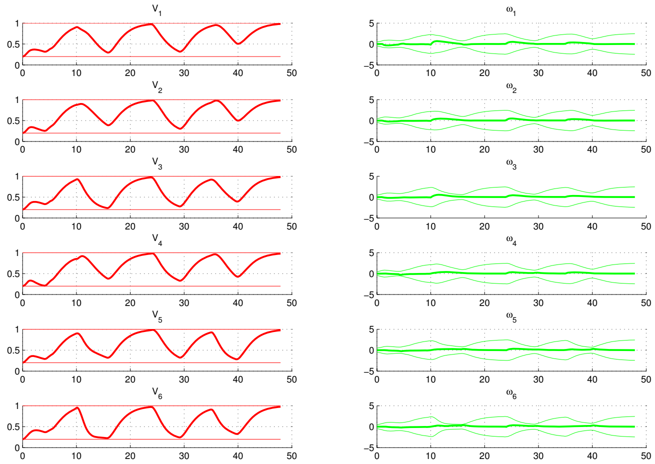

Fig. 4. "Speed" and "Steer" controls for each of the six agents for the coordinated motion shown in Fig.3

4. CONCLUSION

In this paper we have presented a real-time motion planning algorithm for the rigid formation keeping and formation reconfiguration problems. The idea was to design feasible trajectories in terms of maintaining formation with zero error while satisfying individual agent dynamic constraints and capabilities. Existing tracking controllers can then be used on the actual models of the agents to track these trajectories.

The above two problems are intrinsically geometric problems in the configuration space-time and we intend to develop an intrinsic geometric formulation of the constraints, to arrive at the constrained dynamics of the multi-agent systems as opposed to the constrained kinematics we have presented in this paper. In this paper we have only considered explicit bounds on the higher level controls of "speed" and "steer" and hence bounded velocities. In deriving the constrained dynamics, we intend to explicitly consider bounded accelerations through bounds on "force" and "torque" in addition to the already considered bounds on velocities.

REFERENCES

Balch, T., and Arkin, R. C., "Behavior-Based Formation Control for MultirobotTeams," IEEE Trans. Robot. Automat., vol. 14, no. 6, 1998, pp. 926-939.

Giulietti F., Pollini L., and M. Innocenti, "Autonomous formation flight," IEEE Control Syst. Mag., vol. 20, 2000, pp. 34-44.

Agre J. and Clare L., "An integrated architecture for cooperative sensing networks," IEEE Computer, vol. 33, 2000, pp. 106-108.

Lewis M. A. and Tan K.H., "High precision formation control of mobile robots using virtual structures," Auton. Robots, vol. 4, 1997, pp. 387-403.

Desai J. P., Ostrowski J. P., and Kumar V., "Modeling and control of formations of nonholonomic mobile robots," IEEE Trans. Robot. Automat., vol. 17, 2001, pp. 905-908.

Sugihara K., Suzuki I., "Distributed algorithms for formation of geometric patterns with many mobile robots," J. Rob. Syst., vol. 13, no. 3, 1996, pp. 127-139.

Yamaguchi H., Arai T., and Beni G., "A distributed control scheme for multiple robotic vehicles to make group formations," Robot. Auton. Syst., vol. 36, no. 4, 2001, pp. 125-147.

Betts J.T., "Survey of Numerical Methods for Trajectory Optimization", AIAA J. Guid. Control Dyn., vol. 21, no. 2, 1998, pp. 193-207.

M. B. Milam, K. Mushambi, and R. M. Murray, "A computational approach to real-time trajectory generation for constrained mechanical systems," *Proc. IEEE Conf. Decision and Control*, Sydney, Australia, 2000, pp. 845-851.

N. Faiz, S. K. Agrawal, and R. M. Murray, "Trajectory planning of differentially flat systems with dynamics and inequalities," AIAA J. Guid. Control Dyn., vol. 24, 2001, pp. 219-227.

Wang P. K. C., "Navigation strategies for multiple autonomous robots moving in formation," J. Robot. Syst., vol. 8, no. 2, 1991, pp. 177-195.

Desai J. P., Ostrowski J. P., and Kumar V., "Controlling formations of multiple mobile robots," Proc. of IEEE Int. Conf. Robotics and Automation, vol. 4, Leuven, Belgium, 1998, pp. 2864-2869.

Egerstedt M. and Hu X., "Formation constrained multi-agent control," *IEEE Int. Conf. on Robotics and Automation*, Seoul, Korea, 2001, pp. 3961-3967.

Young B. J., Beard R. W., and Kelsey J. M., "A control scheme for improving multi-vehicle formation maneuvers," *Proc. of the American Control Conference, 2001.*

Desai J.P., Kumar V., and Ostrowski J. P., "Control of changes in formation for a team of mobile robots," *Proc. of IEEE Int. Conf. Robot. Automat.*, Detroit, MI, 1999, pp. 1556-1561.

Belta C. and Kumar V., "Motion generation for formations of robots: a geometric approach", *Proc. of IEEE Int. Conf. Robot. Automat.*, Seoul, Korea, 2001.

E. W. Justh and P. S. Krishnaprasad, "Equilibria and steering laws for planar formations," Systems and Control Letters, vol. 52, no. 1, 2004, pp. 25-38.

D. Paley, N. E. Leonard, and R. Sepulchre, "Oscillator models and collective motion: Splay state stabilization of self-propelled particles," *Proc. of 44th IEEE Conf. Decision and Control*, 2005, pp. 3935-3940.

W. B. Dunbar and R. M. Murray, "Receding Horizon Control of Multi-Vehicle Formations: A Distributed Implementation," *Proc. of 43rd IEEE Conf. on Decision and Control*, 2004, pp. 1995-2002.

Dubins L. E., "On curves of minimal length with a constraint on average curvature and with prescribed initial and terminal positions and tangents," Amer. J. Math., vol. 79, 1957, pp. 497-516.