

Formation Stabilization of Multiple Agents Using Decentralized Navigation Functions

Herbert G. Tanner and Amit Kumar
Mechanical Engineering Department
University of New Mexico
Albuquerque, NM 87131-0001

Abstract—We develop decentralized cooperative controllers, which are based on local navigation functions and yield (almost) global asymptotic stability of a group of mobile agents to a desired formation and simultaneous collision avoidance. The formation could be achieved anywhere in the free space; there are no pre-specified final positions for the agents and is rendered stable both in terms of shape and in terms of orientation. Shape and orientation stabilization is possible because the agents regulate relative positions rather than distances with respect to their network neighbors. Asymptotic stability is provable and guaranteed, once the parameters in the local navigation functions are tuned based on the geometry of the environment and the degree of the interconnection network. Feedback controllers steer the agents away from stationary point-obstacles and into the desired formation using information that can be obtained within their sensing neighborhood and through communication with their network neighbors. The methodology is tested in simulation where groups of three and four mobile agents come into formations of triangles and diamonds, navigating amongst obstacles.

I. INTRODUCTION

In this paper we solve the problem of decentralized formation stabilization with collision avoidance, in a way that the resulting formation shape and orientation can be engineered and regulated. The closed loop multi-agent behaviors can be designed, and arbitrary formation arrangements can be achieved with each agent using information that can either measure locally using its sensors or obtain through communication from a *limited set of network neighbors*. Our approach is based on navigation functions [1], which are special types of artificial potential field functions [2] that can be designed to have no local minima and steer the system to a desired destination from almost all initial conditions (except for a set of measure zero). We show that we can tune the parameters of any agent's potential field based on the geometric parameters of the workspace, the graph properties of the communication network, and the sensing characteristics of the agents so that the desired inter-agent relative position configurations are globally asymptotically stable.

This problem has been solved in a centralized formulation: Beard et al. [3] have coordinated formation satellites introducing the concept of the virtual structure. In [4] Tanner et al. have used a navigation function to steer a group of mobile manipulators in an environment with obstacles. Egerstedt [5] introduced the notion of *formation feedback* as a means to regulate agent motion so that global formation constraints

are respected. The problem with centralized approaches is that they are not robust in real implementation, but most importantly centralization involves computational and analytical complexity that does not allow the control architecture to scale with the size of the multi-agent group.

Thus, decentralized solutions are sought. One of the most popular ideas for decentralized cooperative control are behavioral (or reactive) models [6]. These are easily implementable [7], [8] but generally fail to yield any formal guarantees of convergence. The “emerging” formation configurations depend primarily on initial conditions. This is also the case in flocking cooperative control schemes [9], [10], [11] although the asymptotic synchronization of agents velocities is formally proven. In most schemes that use nearest interaction rules [12], [13], [14] the resulting relative positions or the final rendezvous point are unpredictable. Other decentralized approaches rely on ideas based on Internet protocols [15]. Voronoi diagram-based, decentralized nonsmooth cooperative control techniques [16] have shown to globally optimize certain objective functions, yet still, the final arrangement of agents cannot be foretold. A decentralized cooperative control methodology that does guarantee the convergence of agents in particular relative positions while navigating amongst sphere-like obstacles is [17]. However, the approach to decentralization is similar to that of [18]: each agent essentially carries a copy of some centralized coordination scheme, thus still requiring full knowledge of the system and environment state.

As observed in [19], [20], navigation functions, $V(q)$, that solve the problem for static case, are actually Lyapunov functions. The traditional control input $u = -\nabla V$ is not the only input capable of rendering $\dot{V} < 0$; in fact there exist many such input vectors. Any control scheme which makes $\dot{V} = -\nabla V \cdot \dot{q} < 0$ also solves the problem. Our paper is a step forward in the same direction and provides a control input for each agent that decreases the global navigation function and stabilizes a group of mobile agents into a desired formation. In this paper we construct a decentralized version of the navigation-function-based cooperative control algorithm introduced recently in [21] and achieve *global formation stabilization with collision avoidance using only local information*.

The next sections of the paper are organized as follows. In Section II we present the problem statement, using the concept of formation graph. In Section III we present the local potential function and its components. Section IV we show that the

direction of local potential function is a direction along which some common Lyapunov function decreases monotonically. In Section V we present our simulation results. Section VI, closes the paper with some conclusions and ideas for further work.

II. PROBLEM FORMULATION

We consider a homogeneous group of N mobile agents, each with dynamics given by

$$\dot{q}_i = u_i, \quad q_i \in \mathbb{R}^n, \quad i = 1, \dots, N \quad (1)$$

where q_i and u_i are the state and control input of agent i , respectively. In the remaining, q and u will denote the stack vectors of q_i and u_i . The agents are treated as autonomous point-robots.

The objective here is to construct local potential fields that will enable the N agents to stabilize with respect to their group-mates in configurations that make a particular formation, while avoiding collisions between themselves and with obstacles in the environment. The desired formation is specified in terms of a labeled directed graphs.

Definition II.1 (Formation graph) *The formation graph, $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathcal{L}\}$, is a directed labeled graph consisting of:*

- a set of vertices (nodes), $\mathcal{V} = \{v_1, \dots, v_N\}$, indexed by the mobile agents in the group,
- a set of edges, $\mathcal{E} = \{(v_i, v_j) \in \mathcal{V} \times \mathcal{V}\}$, containing ordered pairs of nodes that represent inter-agent position specifications, and
- a set of labels, $\mathcal{L} = \{\gamma_{ij} \mid \gamma_{ij} = \|q_i - q_j - c_{ij}\|^2, c_{ij} \in \mathbb{R}^n, (v_i, v_j) \in \mathcal{E}\}$, indexed by the edges in \mathcal{E} .

All the nodes that are (network) adjacent to node i belong to the neighborhood set of agent i , \mathcal{N}_i . The orientation of the graph is merely used to define the relative position specifications $q_i - q_j - c_{ij}$ and is not meant to denote information flow. The latter is considered bidirectional between two nodes that are adjacent.

Throughout the paper, the use of the Euclidean norm is implied, i.e. $\|\cdot\| \equiv \|\cdot\|_2$. Specifying the formation specifications as edge labels in the form

$$\|q_i - q_j - c_{ij}\|^2 = 0, \quad \forall (v_i, v_j) \in \mathcal{E},$$

not only specifies inter-agent distances, but also their *relative orientation*. The workspace,

$$W = \{q \mid \|q\| \leq R\} \subset \mathbb{R}^n,$$

common for all agents, is assumed to be populated by a set of points $p_j, j = 1, \dots, s$ that represent obstacles.

The objective is to construct an artificial potential function φ_i , that depends only on information that is available to agent i and can steer the latter into a desired relative position with respect to a specific subset of agents with which it can communicate. This local potential function will generate the agent's input as follows:

$$u_i = -K \frac{\partial \varphi_i}{\partial q_i} = -K \nabla_i \varphi_i.$$

What we want to achieve is to have the closed loop system of all agents generate trajectories that converge to configurations in the free space where the desired formation is reached. These trajectories should have arbitrary initial conditions (except for points in a set of measure zero) and be collision free.

III. LOCAL NAVIGATION FUNCTIONS

The main idea in our approach is to build individual navigation functions that drive the agents along trajectories that decrease a common Lyapunov function. This common Lyapunov function could be in the form of a centralized navigation function. A centralized, navigation-function-based approach that recently appeared in literature [21] is our starting point. We then investigate ways to break it up into local components, each associated with an agent in the group, serving as local navigation functions for each of the agents.

In [21] it was shown that centralized formation global stabilization with obstacle avoidance is possible using the following navigation function, defined on a compact connected analytic manifold with boundary, $F \subset \mathbb{R}^{nN}$:

$$\varphi(q) \triangleq \frac{\gamma_d(q)}{e^{\beta(q)^{1/k}}}, \quad (2)$$

where:

- $\gamma_d(q) : F \rightarrow \mathbb{R}_+$ is a positive semi-definite scalar function, vanishing only when the agents are in the desired formation configuration,
- $\beta(q) : F \rightarrow [0, 1]$ function that vanishes only when agents are in contact with the obstacles or with one another.
- k is a (positive) tuning parameter.

The ‘‘goal function’’, $\gamma_d(q)$ is defined as:

$$\gamma_d(q) \triangleq \sum \gamma_{ij}(q_i, q_j), \quad \forall (i, j) \in \mathcal{E} \quad (3)$$

and it is essentially the sum of all edge label (specification) functions. It attains the value of zero only when all the formation specifications are met (i.e., all label functions are zero).

The ‘‘obstacle function’’ is given by

$$\beta(q) \triangleq \prod_{i,k} \beta_{ik} \prod_{i,j} b_{ij}, \quad i, j \in \{1, \dots, N\}, \quad k = 0, \dots, s.$$

in which each β_{it} is a function that models the proximity of a mobile agent with a stationary (point) obstacle, defined as:

$$\beta_{it} \triangleq \left(1 - \lambda \frac{(\|q_i - p_t\|^2 - d^2)^2}{(\|q_i - p_t\|^2 - d^2)^2 + 1} \right)^{\frac{\text{sign}(d - \|q_i - p_t\|) + 1}{2}}, \quad (4)$$

where λ is a tuning parameter and d expresses the ‘‘sensing’’ radius of the agents. The boundary of the workspace is represented by the function β_{i0} (for index $t = 0$). The (discontinuous) exponent in these functions has the effect of ‘‘flattening’’ the functions beyond a certain range d . In this way, the effect of obstacles is local: an obstacle at p_t will only affect the value of φ if an agent approaches it close enough.

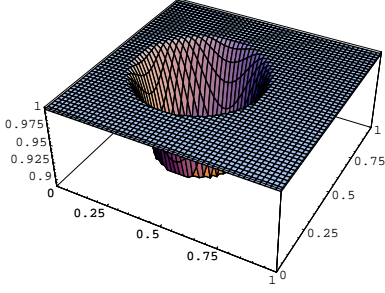


Fig. 1. The form of the obstacle functions β_{it} . Being constant beyond a distance d , they have a local effect on the navigation function.

By appropriately tuning the parameter λ the function β_{it} becomes differentiable at d ,

$$\lambda = \frac{1 + d^4}{d^4}. \quad (5)$$

so that the gradient of φ remains continuous everywhere.

The assumption that both the robots as well as the obstacles are represented by points is not as restrictive as it may seem, since it has been shown [4] that a large class of shapes can be mapped to single points through a series of transformations; this “point-world” topology can be regarded as a degenerate case of the “sphere-world” topology of [22].

Functions b_{ij} model inter-agent proximity. Their structure is similar to that of β_{it} :

$$b_{ij} \triangleq \left(1 - \lambda \frac{(\|q_i - q_j\|^2 - d^2)^2}{(\|q_i - q_j\|^2 - d^2)^2 + 1} \right)^{\frac{\text{sign}(d - \|q_i - q_j\|) + 1}{2}}. \quad (6)$$

In the remaining of this section we will describe how one can construct local potential functions such that if the agents move in the direction of the negated gradient of their respective local potential functions than the cost of the centralized potential function φ decreases monotonically.

We propose a local potential function for the i^{th} agent has the following form

$$\varphi_i \triangleq \frac{\gamma_{d_i}}{e^{\beta_i^{1/k}}} \quad (7)$$

where:

- $\gamma_{d_i} : F \rightarrow \mathbb{R}_+$ is a positive semi-definite scalar function, assuming the value of zero only when all formation specifications related to agent i , γ_{ij} , $j \in \mathcal{N}_i$ are zero (Section III-A).
- $\beta_i : F \rightarrow [0, 1]$ function that vanishes when agent i comes into contact with any of the s obstacles in the workspace or with another agent $j \in \{1, \dots, N\}$, $j \neq i$ (Section III-B).
- k is a (positive) tuning parameter.

A. Agent i^{th} goal function: γ_{d_i}

Function $\gamma_{d_i} : F \rightarrow \mathbb{R}_+$ encodes the control objective of the i^{th} agent, which is to minimize the label functions of all

edges adjacent to agent i .

$$\gamma_{d_i} \triangleq \sum_{l \in \mathcal{N}_i} \gamma_{il}. \quad (8)$$

When γ_{d_i} is minimized with respect to q_i , then agent i “would have done his part” with respect to reaching the desired formation. Ideally, if all formation specifications related to agent i are met, γ_{d_i} becomes zero.

B. Agent i^{th} obstacle function: β_i

Similar to the case of centralized navigation function in [21], β_i is made up as a product of several “obstacle functions.” Collisions can occur not only between an agent and an obstacle, but also between agents as well. For all obstacles $k \in \{1, \dots, s\}$ within a ball of radius d centered at q_i we define a function β_{ik} , as in (4); similarly, for all other agents $j \in \{1, \dots, N\} \setminus \{i\}$ in the same neighborhood, we consider a function b_{ij} as in (6). Then the obstacle function for agent i is defined as

$$\beta_i \triangleq \prod_k \beta_{ik} \prod_j b_{ij}. \quad (9)$$

IV. LOCAL MOTION FOR GLOBAL CONVERGENCE

In this section we will show that the negated gradient of the local potential function constructed in Section III, under certain conditions, is a direction along which the centralized navigation function (2). In the subsections that follow, we evaluate several terms that are central into showing that local controllers can decrease (2).

A. Gradient of φ_i with respect to q_i : $\nabla_i \varphi_i$

Agent i will be steered to a direction aligned with the negated gradient of the local navigation function φ_i , with respect to its own coordinates, q_i . This gradient is given by

$$\nabla_i \varphi_i = \frac{1}{e^{\beta_i^{1/k}}} \left[\nabla_i \gamma_{d_i} - \frac{\gamma_{d_i}}{k} \beta_i^{(\frac{1}{k}-1)} \nabla_i \beta_i \right], \quad (10)$$

One can easily verify that γ_d and γ_{d_i} , are related in the following way:

$$\gamma_d(q) = \frac{1}{2} \sum_{i=1}^N \gamma_{d_i}, \quad (11)$$

$$\nabla_i \gamma_d = \nabla_i \gamma_{d_i}$$

B. Gradient of φ with respect to q_i : $\nabla_i \varphi$

Naturally, the centralized navigation function (2) depends on the coordinates of agent i . A component of the gradient vector of φ contains the partial derivative of φ with respect to q_i . This component is given by:

$$\nabla_i \varphi = \frac{1}{e^{\beta^{1/k}}} \left[\nabla_i \gamma_d - \frac{\gamma_d}{k} \beta^{(\frac{1}{k}-1)} \nabla_i \beta \right]$$

Note now that the coordinates of q_i do not appear in the edge label functions of agents that are not linked to i in the formation graph. Thus,

$$\nabla_i \varphi = \frac{1}{e^{\beta^{1/k}}} \left[\nabla_i \gamma_{d_i} - \frac{\gamma_{d_i}}{k} \beta^{(\frac{1}{k}-1)} \nabla_i \beta \right] \quad (12)$$

In the following sections we bound one by one the components of the gradients of the centralized and decentralized navigation functions in order to investigate how the gradient vectors align.

C. The gradient of local goal functions is bounded

The next lemma provides a lower bound for the goal function of agent i , if it has not completely satisfied any of its formation specifications. If this is the case, then all edge label functions associated with this agent will be bounded away from zero by some constant δ^2 . The bound obtained depends on this constant, δ , the size of the workspace in which the agents move and the maximum degree of the formation graph.

Lemma IV.1 *If agent i^{th} edge label functions are bounded away from zero, i.e. $\{\gamma_{il} \mid \gamma_{il} > \delta^2, \forall l \in \mathcal{N}_i\}$, then the gradient of γ_{d_i} with respect to q_i satisfies*

$$\|\nabla_i \gamma_{d_i}\| \geq \frac{n_i \delta^2}{\mathcal{R}},$$

where \mathcal{R} is the radius of the workspace F (the radius of the largest ball containing F).

Proof: The i^{th} agent goal function is given by

$$\gamma_{d_i} = \sum_{l \in \mathcal{N}_i} \|q_i - q_l - c_{il}\|^2 = \sum_{l \in \mathcal{N}_i} \gamma_{il}.$$

Based on the above, the gradient of γ_{d_i} with respect to q_i is found to be

$$\nabla_i \gamma_{d_i} = 2 \sum_{l \in \mathcal{N}_i} (q_i - q_l - c_{il}) \quad (13)$$

Now, since all edge label functions γ_{il} are lower bounded, we can define δ to be so that $\delta^2 = \min \{\gamma_{il} : l \in \mathcal{N}_i\}$

Therefore, in the region of q_i we are considering, $\min \{\gamma_{d_i}\} \triangleq \delta_i = n_i \delta^2$, where $n_i \triangleq |\mathcal{N}_i|$, in other words n_i is the *degree* of node i .

From its definition, it follows that $\gamma_{d_i}(q_i)$, is a differentiable convex function of q_i . Therefore for all $x, y \in \text{domain of } \gamma_{d_i}(q_i)$, we have the following inequality

$$\gamma_{d_i}(x) \geq \gamma_{d_i}(y) + \nabla \gamma_{d_i}(y)^T (x - y),$$

which can be written in the form

$$\nabla \gamma_{d_i}(y)^T (y - x) \geq \gamma_{d_i}(y) - \gamma_{d_i}(x).$$

If ' x ' is the optimal solution, then $\gamma_{d_i}(y) - \gamma_{d_i}(x) > 0$, and the above yields:

$$\|\nabla \gamma_{d_i}(y)\| \|(y - x)\| \geq \nabla \gamma_{d_i}(y)^T (y - x) \geq \gamma_{d_i}(y) - \gamma_{d_i}(x)$$

Combining the left and right hand sides, we have:

$$\|\nabla \gamma_{d_i}(y)\| \geq \frac{\gamma_{d_i}(y) - \gamma_{d_i}(x)}{\|(y - x)\|}$$

With x being the (globally) optimal argument, we have that $\gamma_{d_i}(x) = 0$, which means that

$$\|\nabla \gamma_{d_i}(y)\| \geq \frac{\gamma_{d_i}(y)}{\|(y - x)\|},$$

and minimizing the right hand side, we have:

$$\|\nabla \gamma_{d_i}(y)\| \geq \min \left\{ \frac{\gamma_{d_i}(y)}{\|(y - x)\|} \right\} = \frac{n_i \delta^2}{\mathcal{R}} \quad (14)$$

■

D. The gradient of local obstacle functions is bounded

The next Lemma provides a bound on the norm of the gradient of the obstacle functions that model the proximity of agents to obstacles. The bound suggested by the Lemma depends on the radius d of the region around each of the agents in which they can identify obstacles and other agents.

Lemma IV.2 *The norm of the gradient of β_{it} with respect to q_i i.e., $\|\nabla_i \beta_{it}\|$ is upper bounded by*

$$\frac{8}{3\sqrt{3}} \frac{(1 + d^4)}{d^7}.$$

Proof: First consider β_{it} as a function of $z_{it} \triangleq \|q_i - p_t\|^2 - d^2$: $\beta_{it}(q_i) = \beta_{it}(z_{it}(q_i))$. Using the chain rule,

$$\nabla_i \beta_{it} = \frac{-2\lambda z_{it}}{(1 + z_{it}^2)^2} \nabla_q(z_{it}), \quad (15)$$

and thus,

$$\|\nabla_i \beta_{it}\| = \frac{-4\lambda z_{it}}{(1 + z_{it}^2)^2} \sqrt{z_{it} + d^2}. \quad (16)$$

Obviously, one has $\|\nabla_i \beta_{it}\| \leq (1 + z_{it}^2)^2 \|\nabla_i \beta_{it}\|$, and consequently,

$$\max \{\|\nabla_i \beta_{it}\|\} \leq \max \{(1 + z_{it}^2)^2 \|\nabla_i \beta_{it}\|\}. \quad (17)$$

Define the function

$$F(z_{it}) \triangleq \{(1 + z_{it}^2)^2\} \|\nabla_i \beta_{it}\|.$$

Interestingly, while $\|\nabla_i \beta_{it}\|$ is difficult to bound directly, function $F(z_{it})$ can be maximized easily: $\max \{F(z_{it})\}$ is obtained for $z_{it} = -\frac{2}{3}d^2$, (Figure 2) and therefore $\max \{F(z_{it})\} = \frac{8\lambda}{3\sqrt{3}d^3} = \frac{8}{3\sqrt{3}} \frac{(1 + d^4)}{d^7}$ and due to (17),

$$\max \{\|\nabla_i \beta_{it}\|\} \leq \frac{8}{3\sqrt{3}} \frac{(1 + d^4)}{d^7} \quad (18)$$

■

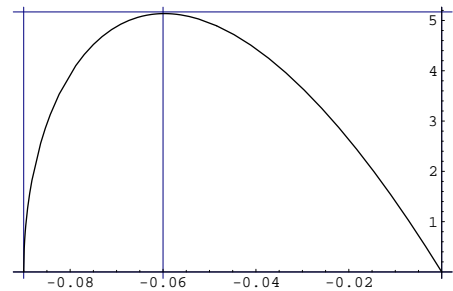


Fig. 2. The gradient of F has a unique maximum.

Let s_i be the total number of obstacles within the sensor radius of i^{th} agent. Now β_i as defined in (9) can be compactly written as

$$\beta_i = \prod_{t=0}^{s_i} \beta_{it} = \beta_{it} \cdot \bar{\beta}_{it}$$

where $\bar{\beta}_{it}$ is the ‘‘omitted product’’ of β_{it} [22]. Note that obstacles outside this region correspond to functions that are identically equal to 1; thus, generally $\prod_{t=0}^{s_i} \beta_{it} = \prod_{t=0}^s \beta_{it}$. Then we can write:

$$\nabla_i \beta_i = \nabla_i \left\{ \prod_{t=0}^s \beta_{it} \right\} = \nabla_i \{ \bar{\beta}_{ij} \beta_{ij} \} = \sum_{j=0}^s \{ \bar{\beta}_{ij} (\nabla_i \beta_{ij}) \}. \quad (19)$$

Thus, maximizing the left and right hand sides:

$$\begin{aligned} \max \{ \|\nabla_i \beta_i\| \} &= \max \left\| \sum_{j=0}^s \{ \bar{\beta}_{ij} (\nabla_i \beta_{ij}) \} \right\| \\ &= (s+1) \left\{ \frac{8}{3\sqrt{3}} \frac{(1+d^4)}{d^7} \right\} \end{aligned} \quad (20)$$

since $\max \bar{\beta}_{it} = 1$.

The same bound can be used for the obstacle function of the centralized navigation function, β . The function can similarly be expressed in terms of the omitted product of i as $\beta = \beta_i \bar{\beta}_i$ so that its gradient is written

$$\nabla_i \beta = \beta_i \nabla_i \bar{\beta}_i + \bar{\beta}_i \nabla_i \beta_i$$

and because $\bar{\beta}_i$ does not depend on q_i ,

$$\nabla_i \beta = \bar{\beta}_i \nabla_i \beta_i = \bar{\beta}_i \left[\sum_{j=0}^s \{ \bar{\beta}_{ij} (\nabla_i \beta_{ij}) \} \right].$$

Maximizing the left and right hand sides,

$$\begin{aligned} \max \{ \|\nabla_i \beta\| \} &= \max \left\| \bar{\beta}_i \left[\sum_{j=0}^s \{ \bar{\beta}_{ij} (\nabla_i \beta_{ij}) \} \right] \right\| \\ &= (s+1) \left\{ \frac{8}{3\sqrt{3}} \frac{(1+d^4)}{d^7} \right\} \end{aligned} \quad (21)$$

Our main result is captured in the following Proposition. It states that an appropriate value of the parameter k in the local navigation functions can ensure that the local gradients are roughly aligned with the components of the gradient of the centralized navigation function. This implies that the centralized navigation function φ will be decreasing along the trajectories of $\dot{x}_i = -K \nabla_i \varphi_i$.

Proposition IV.3 *There exists a sufficiently large value for k so that the trajectories of the system $q_i = -K \nabla_i \varphi_i$ monotonically decrease the navigation function in (2).*

Proof: To determine whether the components of $\nabla \varphi$ are aligned to the corresponding $\nabla_i \varphi_i$, we are considering their inner product. If the inner product can be made (strictly) positive, then the centralized function will be monotonically decreasing, implying that the system will eventually reach its

desired configuration. The inner product $\langle \nabla_i \varphi, \nabla_i \varphi_i \rangle$, is given by the expression

$$\frac{\left[\nabla_i \gamma_{d_i} - \frac{\gamma_{d_i}}{k} \beta_i^{(\frac{1}{k}-1)} \nabla_i \beta_i \right]^T \left[\nabla_i \gamma_{d_i} - \frac{\gamma_{d_i}}{k} \beta_i^{(\frac{1}{k}-1)} \nabla_i \beta_i \right]}{e^{(\beta_i^{1/k} + \beta_i^{1/k})}}$$

Now define the functions x_i , a_i and b_i as follows:

$$\begin{aligned} x_i &= \nabla_i \gamma_{d_i}, \\ a_i &= \gamma_{d_i} \beta_i^{(\frac{1}{k}-1)} \nabla_i \beta_i, \\ b_i &= \gamma_{d_i} \beta_i^{(\frac{1}{k}-1)} \nabla_i \beta_i. \end{aligned}$$

Substituting, we have

$$\begin{aligned} \langle \nabla_i \varphi, \nabla_i \varphi_i \rangle &= \left(x_i - \frac{a_i}{k} \right)^T \left(x_i - \frac{b_i}{k} \right) \\ &= x_i^T x_i - \frac{1}{k} x_i^T (a_i + b_i) - \frac{1}{k^2} a_i^T b_i. \end{aligned}$$

In order to show that the inner product is positive it is sufficient to show that

$$\|x_i\|^2 > \frac{1}{k} \underbrace{[x_i^T (a_i + b_i) - \frac{1}{k} (a_i^T b_i)]}_{\{A\}}. \quad (22)$$

Expression A can be bounded as follows:

$$\begin{aligned} \{A\} &< \left[\frac{(\|x_i\|^2 + \|a_i + b_i\|^2)}{2} - \frac{1}{k} (a_i^T b_i) \right] \\ &= \frac{1}{2} \underbrace{\left[\|x_i\|^2 + \|a_i\|^2 + \|b_i\|^2 + 2 \frac{(k-1)}{k} a_i^T b_i \right]}_{\{B\}} \end{aligned}$$

Assuming $k > \frac{1}{2}$, $\{B\} < \frac{1}{2} [\|x_i\|^2 + (\|a_i\| + \|b_i\|)^2]$

Hence for the inner product to be positive, we have the following condition

$$\|x_i\|^2 > \frac{1}{2k} [\|x_i\|^2 + (\|a_i\| + \|b_i\|)^2]$$

which provides us with a condition on k ,

$$k > \frac{1}{2} \left[1 + \frac{(\|a_i\| + \|b_i\|)^2}{\|x_i\|^2} \right]$$

In view of (11), (20) and (21), one can write that

$$\begin{aligned} \|a_i\| &\leq \gamma_d \|\nabla_i \beta\| \leq \frac{N}{2} \max\{\gamma_{d_i}\} \|\nabla_i \beta\|, \\ \|b_i\| &\leq \gamma_{d_i} \|\nabla_i \beta_i\| \leq \frac{N}{2} \max\{\gamma_{d_i}\} \|\nabla_i \beta\|. \end{aligned}$$

Thus, it is sufficient to have

$$k > \frac{1}{2} + \frac{N^2 \max\{\gamma_{d_i}\}^2 \|\nabla_i \beta\|^2}{2 \|x_i\|^2}$$

and using (21) and (14),

$$k > \frac{1}{2} + \frac{2N^2 \mathcal{R}^4 (s+1)^2 \left\{ \frac{8}{3\sqrt{3}} \frac{(1+d^4)}{d^7} \right\}}{n_i^2 \delta^4} \quad (23)$$

If (23) holds then

$$\dot{\varphi} = \sum_{i=1}^N (\nabla_i \varphi)^T \dot{q} = -K \sum_{i=1}^N \left\langle \frac{\partial \varphi}{\partial q_i}, \frac{\partial \varphi_i}{\partial q_i} \right\rangle < 0$$

■

V. SIMULATION RESULTS

In this section, we present simulation studies in which a team of three agents form a triangle, and a team of four agents form a diamond amidst obstacles arranged in a Π formation. In all the simulation cases presented here the sensor radius of all the agents is $d = 0.2$. An obstacle is not visible to a particular agent if it is outside its sensor radius.

In Figures 3, 5 and 7, the solid (double) arrows connecting the agents' final positions indicate the formation graph edges. Dotted lines denote inter-agent distances which are not taken into account explicitly in the agents goal functions.

A. Three agent formation

In this case, we coordinate a team of three agents into forming an equilateral triangle, pointing "north". The agents start at initial configurations denoted by $1'$, $2'$, and $3'$ and reach their final configuration denoted by 1, 2, and 3 in Figure 3. The corresponding inter-agent distance is plotted in Figure 4, in which we can verify the obstacle avoidance capabilities of the agents. Note that due to the finite sensing radius, agents come close before engaging into collision avoidance maneuvers.

In Figure 4, it can be seen that agent 2 comes close to agent 3, although still no collision occurs. We increase the collision avoidance properties of the fields by decreasing the value of the tuning parameter, k from 2 to 1.2, thus making the effect of proximity functions (β or b) stronger. In the simulation run depicted in Figure 5, we clearly see that under the influence of the new field, agent 2 performs a collision avoidance maneuver as soon as it starts coming close to agent 3. The corresponding inter-agent distance plot Figure 6 confirms this statement.

B. Four agent formation

In these simulation scenarios, the size of the group is increased to four. The objective here is to coordinate the team into forming a diamond, with agent 1 pointing "north." The agents start at initial configurations denoted by $1'$, $2'$, $3'$, and $4'$ and reach their final configuration denoted by 1, 2, 3, and 4 in Figure 7. In this case the initial conditions are chosen so that the agents start the motion in the neighborhood of a Π formation of point-obstacles. The agent trajectories depicted in Figure 7 demonstrate the ability of the group to stabilize to the desired formation configuration while avoiding collisions with nearby obstacles. The corresponding inter-agent distances are plotted in Figure 8, verifying that although the paths of the agents cross during maneuvering, their distances remain always bounded away from zero.

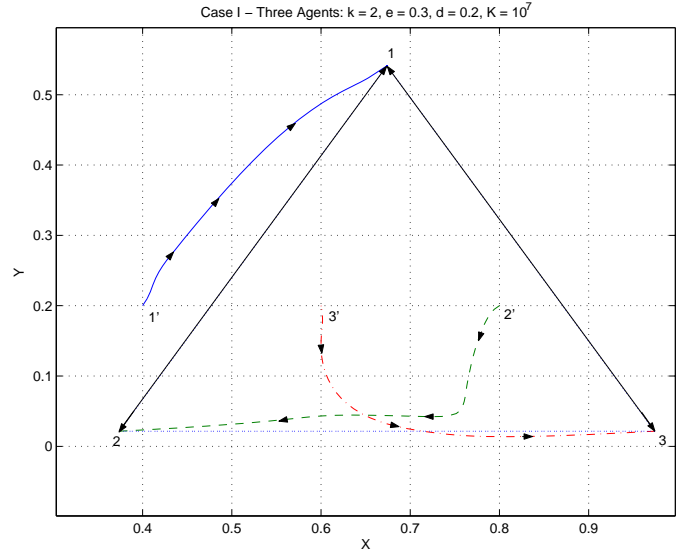


Fig. 3. Inter-agent collision avoidance and convergence to desired formation with $k = 2$.

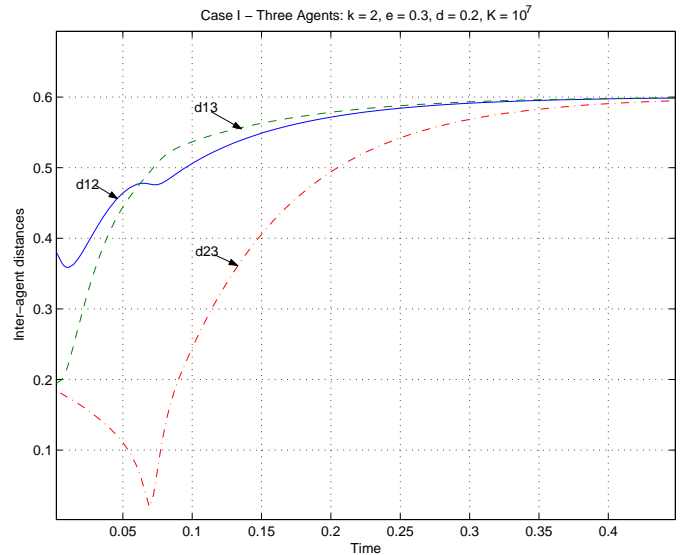


Fig. 4. Inter-agent distances with $k = 2$.

VI. CONCLUSION

A decentralized cooperative control scheme is presented that allows a team of mobile agents to asymptotically converge to a desired formation of particular shape and orientation from almost any initial conditions. Collision avoidance, both among agents and between agents and environment obstacles, is guaranteed. The formation can be reached at any position in the free workspace, because the technique does not "tie" the formation to a particular point in space. Agent controllers use state information from a limited set of specific neighboring agents and have access to environment data within a certain region around their location. Although the controllers are

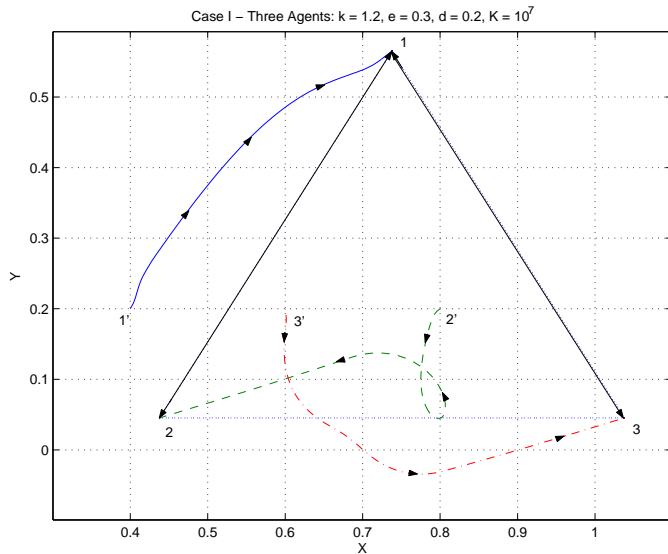


Fig. 5. Inter-agent Collision avoidance and convergence to desired formation with $k = 1.2$.

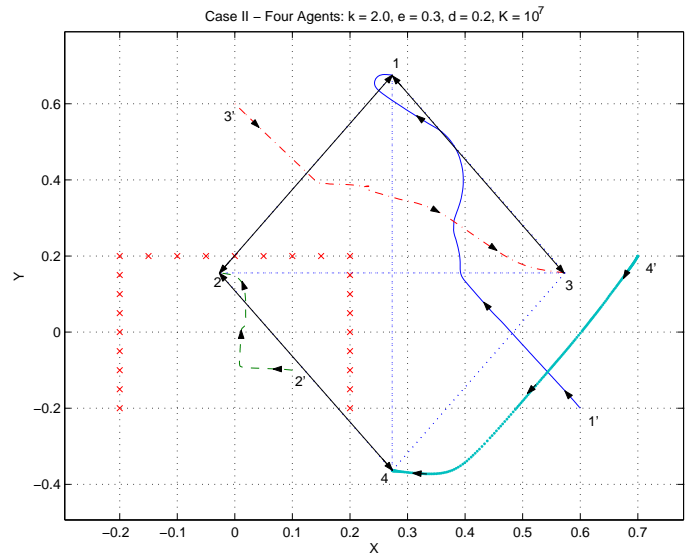


Fig. 7. Collision avoidance during a 4-agent formation stabilization maneuver in an environment with stationary point-obstacles.

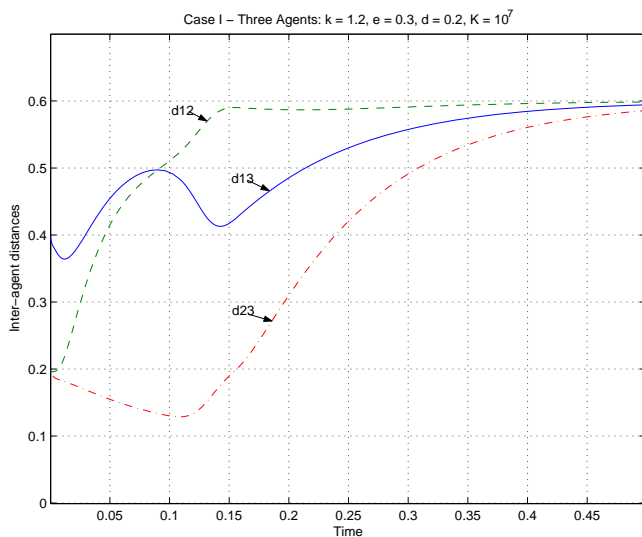


Fig. 6. Inter-agent distances with $k = 1.2$.

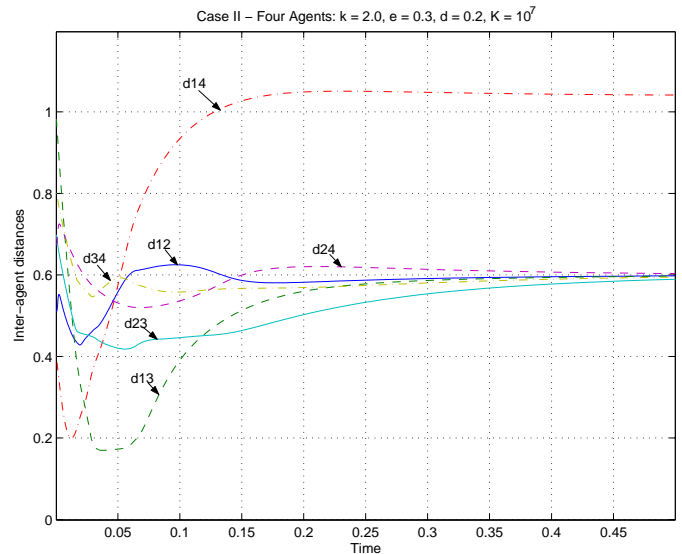


Fig. 8. Inter-agent distances during the diamond formation stabilization maneuver.

completely decentralized, global convergence is still achieved because we can theoretically adjust the local potential fields that generate the agents' control inputs to align with the gradient directions of a common Lyapunov function.

Future work includes investigation of the performance of the decentralized formation stabilization scheme in terms of network parameters, topology and information flow, and quantification of the performance deterioration compared to "ideal" centralized alternative control architectures. Another area in which we would like to focus our attention is in determining a generalized structure for the parameter c_{ij} , which is critical for determining the final shape of the formation.

REFERENCES

- [1] E. Rimon and D. Koditschek, "Exact robot navigation using artificial potential functions," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 5, pp. 501–518, October 1992.
- [2] J. Barraquand, B. Langlois, and J. C. Latombe, "Numerical potential field techniques for robot path planning," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 22, no. 2, pp. 224–241, 1992.
- [3] R. W. Beard, J. Lawton, and F. Y. Hadaegh, "A coordination architecture for spacecraft formation control," *IEEE Transactions on Control Systems Technology*, vol. 9, no. 6, pp. 777–790, November 2001.
- [4] H. G. Tanner, S. G. Loizou, and K. J. Kyriakopoulos, "Nonholonomic navigation and control of multiple mobile manipulators," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 1, pp. 53–64, February 2003.

- [5] P. Ögren, M. Egerstedt, and X. Hu, "A control Lyapunov function approach to multi-agent coordination," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, October 2002.
- [6] T. Balch and R. C. Arkin, "Behavior-based formation control for multirobot teams," *IEEE Transactions on Robotics and Automation*, vol. 14, no. 6, pp. 926–939, December 1998.
- [7] J. Lawton, B. Young, and R. Beard, "A decentralized approach to elementary formation maneuvers," in *Proceedings of the IEEE International Conference on Robotics and Automation*, San Francisco, CA, April 2000, pp. 2728–2733.
- [8] J. Fredslund and M. J. Matarić, "A general algorithm for robot formations using local sensing and minimal communications," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 837–846, October 2002.
- [9] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1520–1533, September 2004.
- [10] H. G. Tanner, A. Jadbabaie, and G. J. Pappas, "Stable flocking of mobile agents, Part I: Fixed topology," in *Proceedings of the IEEE Conference on Decision and Control*, Maui, Hawaii, December 9–12 2003, pp. 2010–2015.
- [11] V. Gazi and K. M. Passino, "Stability analysis of swarms," in *American Control Conference*, Anchorage, Alaska, May 2002, pp. 1813 – 1818.
- [12] Z. Lin, M. Broucke, and B. Francis, "Local control strategies for groups of mobile autonomous agents," *IEEE Transactions on Automatic Control*, vol. 49, no. 4, pp. 622–629, April 2004.
- [13] E. Klavins, "Toward the control of self-assembling systems," in *Control Problems in Robotics*, ser. Springer Transactions in Advanced Robotics, A. Bicchi, H. Christensen, and D. Prattichizzo, Eds. Springer, 2003, vol. 4, pp. 153–168.
- [14] J. Lin, A. Morse, and B. Anderson, "The multi-agent rendezvous problem," in *Proceedings of the 42th IEEE Conference on Decision and Control*, Maui HI, December 2003, pp. 1508–1513.
- [15] R. Sandoval-Rodriguez and C. T. Abdallah, "Internet-like protocols for the control and coordination of multiple agents with time delay," in *Proceedings of the IEEE International Symposium on Intelligent Control*, Houston, TX, October 2003.
- [16] J. Cortés and F. Bullo, "Coordination and geometric optimization via distributed dynamical systems," *SIAM Journal of Control and Optimization*, May 2004, (to appear).
- [17] D. V. Dimarogonas, M. M. Zavlanos, S. G. Loizou, and K. Kyriakopoulos, "Decentralized motion control of multiple holonomic agents under input constraints," in *Proceedings of 42nd IEEE Conf. on Decision and Control*, Hawaii, December 2003, pp. 3390–3395.
- [18] W. Ren and R. W. Beard, "A decentralized scheme for spacecraft formation flying via the virtual structure approach," *AIAA Journal of Guidance Control and Dynamics*, vol. 27, no. 1, pp. 73–82, January 2004.
- [19] M. Erdmann, "Understanding action and sensing by designing action-based sensors," *International Journal of Robotics Research*, vol. 14, no. 5, pp. 483–509, 1995.
- [20] J. M. Esposito and V. Kumar, "A method for modifying closed-loop motion plans to satisfy unpredictable dynamic constraints at runtime," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Washington, CA, May 2002, pp. 1691–1696.
- [21] H. G. Tanner and A. Kumar, "Towards decentralization of multi-robot navigation functions," in *Proceedings of the 2005 IEEE Conference on Robotics and Automation*, April 2005.
- [22] E. Rimon and D. E. Koditschek, "Exact robot navigation using artificial potential functions," *IEEE Trans. on Robotics and Automation*, vol. 8, no. 5, pp. 501–518, 1992.