Robotica (2008) volume 26, pp. 691–701. © 2008 Cambridge University Press
doi:10.1017/S0263574708004323 Printed in the United Kingdom

Decentralized controllers for shape generation with robotic swarms

M. Ani Hsieh[1,*], Vijay Kumar[1] and Luiz Chaimowicz[2]

[1]*GRASP Laboratory, University of Pennsylvania*
[2]*Computer Science Department, Universidade Federal de Minas Gerais*

(Received in Final Form: February 7, 2008. First published online: April 11, 2008)

SUMMARY
We address the synthesis of controllers for a swarm of robots to generate a desired two-dimensional geometric pattern specified by a simple closed planar curve with local interactions for avoiding collisions or maintaining specified relative distance constraints. The controllers are decentralized in the sense that the robots do not need to exchange or know each other's state information. Instead, we assume that the robots have sensors allowing them to obtain information about relative positions of neighbors within a known range. We establish stability and convergence properties of the controllers for a certain class of simple closed curves. We illustrate our approach through simulations and consider extensions to more general planar curves.

KEYWORDS: robot swarms, decentralised control; motion planning.

1. Introduction

We consider the deployment of a robotic team in an urban environment that can autonomously perform surveillance monitoring of specified areas, e.g., perimeter surviillance/boundary coverage, with additional applications in cooperative manipulation where robots can transport/capture large objects by surrounding it, and cordoning off hazardous regions after chemical spills or biological terrorist attacks. In all these problems, the robots must have the ability to organize and generate complex shapes in two dimensions, often maintaining constraints with respect to neighbors for communication. In these situations, it is often difficult, if not impossible, to efficiently manage or control the team through centralized algorithms or tele-operation. Accordingly, it makes sense to develop strategies where the robots can be programmed with simple but identical behaviors that can be realized with limited on-board computational, communication, and sensing resources.

In nature, the emergence of complex group level behaviors from simple agent level behaviors are often seen in the group dynamics of bee[1] and ant[2] colonies, bird flocks,[3] and fish schools.[4] These systems generally consist of large numbers of organisms that individually lack either the communication or computational capabilities required for centralized control. As such, when considering the deployment of large robot teams, it makes sense to consider such "swarming paradigms" where agents have the capability to operate asynchronously and can determine their trajectories based on local sensing and/or communication. In this paper, we present a scalable approach that allows a swarm of robots to synthesize shapes and patterns in two dimensions while maintaining nearest neighbor constraints. Our main contribution is the synthesis and analysis of gradient based decentralized controllers that allow a large team of robots to converge to some desired two-dimensional boundary curve while maintaining inter-agent constraints via local interactions. The stability and convergence properties of these controllers for a certain class of simple closed curves are established. Lastly, we consider the extension of our methodology to more general planar curves.

This paper is organized as follows: Section 2 presents related work in motion planning and control of large groups of robots. The problem is formulated in Section 3 and our methodology is presented in Section 4. The properties of our controller, including stability and convergence, are discussed in Section 5. A method for synthesizing general two dimensional patterns and shapes using radial basis functions is presented in Section 6. Section 7 summarizes our simulation results. Finally, we discuss directions for future work in Section 8.

2. Related Work

Our goal is to synthesize decentralized feedback control laws to enable a team of robots to align themselves along boundaries of two-dimensional shapes while satisfying inter-agent constraints. We would like these controllers to be decentralized in the sense that agents do not need to exchange or know each other's state information, rather we will assume that the agents have the necessary sensors to infer the relative positions of other agents within a given range. Additionally, we would like simple controllers that can be identically implemented at the agent level and result in provably correct group level behavior, i.e., a strategy consistent with the swarming paradigm.

One of the earliest works to take inspiration from biological swarms for motion generation was presented by Reynolds in 1987[5] where he proposed a method for generating visually satisfying computer animations of bird flocks, often referred as *boids*. Almost a decade later, Vicsek *et al.* showed through simulations that a team of autonomous

* Corresponding author. E-mail: mya@seas.upenn.edu

https://doi.org/10.1017/S0263574708004323 Published online by Cambridge University Press

agents moving in the plane with same speeds but different headings converge to the same heading using nearest neighbor update rules.[6] The theoretical explanation for this observed phenomena was provided by Jadbabaie *et al.*[7] and Tanner *et al.* extended these results to provide detailed analysis of the stability and robustness of such flocking behaviors.[8] These works show that teams of autonomous agents can stably achieve concensus through local interactions alone, i.e., without centralized coordination, and have attracted much attention in the multi-robot community.

Previous works in group coordination using decentralized controllers to synthesize geometric patterns include the works by Albayrak *et al.*[9] and Suzuki *et al.*[10] Albayrak *et al.* considered line and circle formations,[9] while Suzuki *et al.* considered more general geometric patterns.[10] However, both approaches required each robot to have full knowledge of the positions of all the other robots. The use of decentralized leader-follower controllers to synthesize robot formations was proposed by Desai *et al.*,[11] however, the approach required the assignment of different controllers and set points to different robots which makes scaling to large groups difficult. Asymptotic stability of decentralized leader-follower formation control for a group of nonholonomic robots in $SE(2)$ with fixed interconnection topologies was presented by Fierro *et al.*[12] A navigation function based approach for multi-robot navigation and coordination is presented by Tanner *et al.*[13] and Loizou *et al.*[14]

In general, leader-follower controllers require labeling of robots; Ogren *et al.* relaxed this assumption in the development of coordination strategies for a group of unidentified, holonomic robots.[15] Similar approaches for multi-robot manipulation were presented by Song and Kumar[16] and Pereira and Kumar[17] respectively. Chaimowicz *et al.* extended these approaches to arbitrary shapes and established convergence to patterns that approximate the desired shape.[18] Correll *et al.* experimentally compared three distributed algorithms for boundary coverage for a robotic swarm.[19] In addition, they modeled a robotic swarm as a collection of probabilistic finite state machines and presented a methodology for system identification of both the linear and non-linear robotic swarm systems for parts inspection applications.[20] Although experimental and simulation results were shown in these works, they did not provide theoretical results for stability and convergence.

Other recent works in formation control include the work by Sepulchre *et al.*[21] where a complete communication interconnection topology is used to stabilize a group of agents in circular orbits to isolated relative equilibria in the plane. Paley *et al.*[22] extended these results to include elliptical and superelliptical orbits and relaxed the communication interconnection topology to undirected circulant graphs. The stabilization of multiple agents to star-shaped orbits with relative arc-length constraints was presented by Zhang *et al.*[23, 24] In these works, boundary coverage is achieved by maintaining inter-agent separation distances specified in terms of the arc-length of the boundary of interest rather than inter-agent Euclidean distances. The problem of detecting and tracking a specific environmental boundary where the control laws were determined using a partial differential equation approach was discussed by Bertozzi's group.[25] Lastly, the surveillance of an environment with obstacles was achieved by Kerr *et al.*,[26] where individual robots in a swarm were modeled as gas particles. Unlike the works by Peng, Fierro, and Loizou,[12, 14, 16] these works treat individual agents as point particles.

Belta *et al.* presents a different approach to the shape generation/formation control problem.[27] Control abstractions for groups of planar robots were derived along with decentralized controllers such that motion planning for the group can be achieved in a lower dimensional space. They showed how groups of robots can be modeled as deformable ellipses, and presented decentralized controllers that allowed the control of the pose and position of the ellipses. The approach was extended to teams of heterogeneous robots by building a hierarchy of ground and air vehicles which allowed groups to split and merge,[28] and to robot teams in three dimensions.[29] Formations for small teams of robots can also be achieved by modeling the team as controlled Lagrangian systems on Jacobi shape space.[30] More recently, the problem of positioning a team of robots to generate different shapes in two and three dimensions was formulated as a second-order cone program.[31] Lastly, a coordination strategy that stabilizes a group of vehicles to an arbitrary desired group shape derived from spatial networks of interconnected struts and cables, i.e., tensegrity structures, was presented by Nabet *et al.*[32]

In this work, we build on the results of Chaimowicz[18] and Hsieh,[33] and in the spirit of the works by Belta, Chaimowicz, and Michael[27, 28, 29] mentioned previously, we address the synthesis of decentralized controllers that guarantees the convergence of the team to the boundary of some desired shape as well as the stability of the resulting formation, all the while maintaining inter-agent constraints through local interactions. While our approach is similar the works of Zhang *et al.*,[23, 24] we take a slightly different approach to the pattern generation problem and consider inter-agent constraints that are functions of Euclidean distances between agents rather than arc-lengths. Lastly, our strategy does not require inter-agent communication and can be achieved via sensing alone. This may be relevant in applications like persistent surveillance where limited bandwidth must be preserved to enable robots to communicate with each other in order to integrate and fuse the information acquired by various sensors.

3. Problem Formulation

Assume a swarm of N planar fully actuated robots each with the following dynamics

$$\dot{q}_i = v_i \tag{1a}$$

$$\dot{v}_i = u_i \tag{1b}$$

where $q_i = (x_i, y_i)^T$, v_i, and u_i respectively denote the ith agent's position, velocity, and control input. Thus, the robot state is a 4×1 state vector $\mathbf{x}_i = [q_i^T v_i^T]^T$. We define the configuration space as $Q \subset \mathbb{R}^{2N}$, and the configuration of the swarm of robots as $\mathbf{q} = [q_1^T \ldots q_N^T]^T \in Q$. Similarly, the

https://doi.org/10.1017/S0263574708004323 Published online by Cambridge University Press

state space vector is given by $\mathbf{x} = [\mathbf{x}_1^T \dots \mathbf{x}_N^T]^T \in \mathbf{X} \subset \mathbb{R}^{4N}$. In general, we consider all systems of N agents whose individual dynamics can be transformed into (1) via some diffeomorphic state transformation.

Our goal is to design control inputs that will drive the group of N robots to the boundary (curve) of a desired smooth, compact set, i.e., *shape*, while maintaining inter-robot constraints. This is relevant for applications such as perimeter surveillance or surrounding an object for capture and/or transportation. Thus, the controller synthesis problem for *pattern generation* is to find a controller that can drive the team to the desired boundary while:

[P1] avoiding collisions with other agents, or
[P2] maintaining specified proximity constraints with other agents, such as for communication maintenance.

We outline our methodology in the following section.

4. Methodology

4.1. Assumptions and definitions

For a given desired shape, $\mathcal{S}$, whose boundary is denoted by $\partial\mathcal{S}$, assume $\partial\mathcal{S}$ is a two dimensional planar curve in an obstacle-free workspace that can be described by an implicit function, $s(x, y) = 0$. In general, we will assume that the boundary curves of interest are regular closed, simple, smooth planar curves enclosing star shaped sets. The regular and simple assumptions are necessary to ensure that the closed curves do not self intersect.[34]

In situations, where it is important that the team maintains a connected communication network or specific team members remain within a prescribed range of one another to enable grasping of objects for transportation/manipulation, we will further require the team to maintain a desired interconnection topology. We use a *proximity graph*, $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, to model the inter-robot constraints, where $\mathcal{V}$ and $\mathcal{E}$ denote the set of vertices and edges of $\mathcal{G}$. Each robot is then represented by a vertex in $\mathcal{V}$ and proximity relations between pairs of robots are represented by the edges in $\mathcal{E}$. As such, for any two robots represented by $a, b \in \mathcal{V}$, we say a and b are adjacent or neighbors, denoted by $a \sim b$, if a is in the neighborhood of b and b is in the neighborhood of a, and as such the edge $(a, b) \in \mathcal{E}$. In our analysis, robot i's neighborhood is defined as the ball $\mathcal{B}_i = \{q \,|\, \|q_i - q\| \leq d\}$, where $d > 0$ denotes the interaction range. For the constraints under consideration, we choose $d = \delta$ for collision avoidance or $d = \Delta$ for proximity maintenance. In practice, whether for collision avoidance or proximity maintenance, this prescribed range can be determined based on the communication and/or sensing hardware, performance requirements within a given environment, and/or experimental results. For any proximity graph $\mathcal{G}$, the the $N \times N$ adjacency matrix is defined as:

$$A_{ij} = \begin{cases} 1 & \text{if } j \in \Gamma_i \\ 0 & \text{otherwise} \end{cases}.$$

Given a set of inter-agent constraints, we encode the information in a *desired proximity graph*, $\mathcal{G}^d$, such that every inter-agent constraint is represented by an edge. Thus, the graph $\mathcal{G}^d$ represents the desired interconnection topology and we denote its associated adjacency matrix as A^d. We will assume that the desired proximity graph is always a subgraph of the initial proximity graph to ensure the team initializes in a feasible configuration.

Lastly, since our goal is to synthesize decentralized controllers that will allow a team of robots to converge to the boundary while satisfying inter-agent constraints, we note that given $d > 0$, the length of $\partial\mathcal{S}$ naturally imposes an upper bound on the number of robots, denoted by e.g., $N_{max} > 0$, that can fit on the boundary. Similarly, the length of the boundary will naturally impose a lower bound, N_{min}, on the number of robots that can effectively cover the desired boundary given a fixed sensing range. We refer the interested reader to the Appendix for a brief discussion on determining N_{max} and N_{min}.

In this work, we are primarily concerned with convergence to the desired boundary and as such we will make the following additional assumptions:

1. $N < N_{max}$;
2. $\rho_{min} > \delta$, where ρ_{min} denotes the smallest radius of curvature of $\partial\mathcal{S}$;
3. $\min_{s\in[\frac{\pi\rho_0}{2}, L-\frac{\pi\rho_0}{2}]} \|q_0(s) - q(s)\| > \delta$ for any $q_0(s) \in \partial\mathcal{S}$, where $s \in [0, L]$ denotes the arclength and ρ_0 denotes the radius of curvature at q_0.

Assumption 1 ensures all agents with finite interaction range will be able to converge to the desired boundary while satisfying all constraints. Assumptions 2 and 3 ensure convergence by excluding boundaries with sharp turns and star shapes with narrow passages, e.g., hourglass shapes, that may result in robots repelling each other away from the boundary when avoiding collisions.

4.2. Controller synthesis

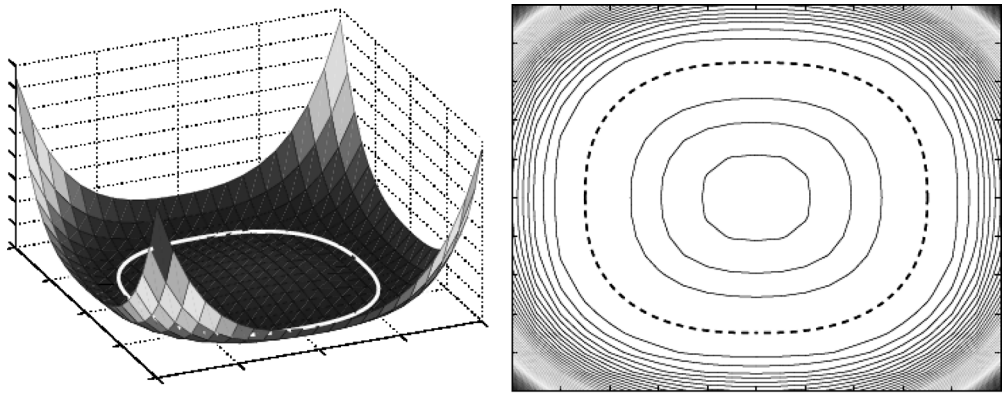
4.2.1. Shape functions. For a desired shape, $\mathcal{S}$, we define the *shape function*, $f : \mathbb{R}^2 \to \mathbb{R}$, such that f is positive semi-definite in Q and for all $(x, y) \in \partial\mathcal{S}$, i.e., points on the curve $s(x, y) = 0$, $f(x, y) = 0$. In general, for any parameterization $s(x, y) = 0$, $f = (s(x, y))^2$ is a candidate shape function. For star shapes, we choose $f = s^2$ such that

1. $s(x, y)$ is at least twice differentiable on Q; and
2. $s(x, y)$ is polar at some $\hat{q}$, where $\hat{q}$ exists in the interior of $\mathcal{S}$, i.e., s has a unique minimum at $\hat{q}$.

Shape functions of this form have level set curves that are consistent with the desired boundary curve, i.e., if the desired boundary curve is convex then so are the level sets of f. This is relevant for stability and convergence analysis. Figure 1 shows the shape function for a compact set enclosed by Piet Hein's superellipse and its corresponding level set curves.

4.2.2. Shape discrepancy functions. To determine the peformance of our controller, we define the *shape discrepancy function*, $\phi_S : Q \to \mathbb{R}$, such that ϕ_S is real analytic and positive semi-definite, whose zero isocontour is identically the boundary of the desired shape $\mathcal{S}$. While there

https://doi.org/10.1017/S0263574708004323 Published online by Cambridge University Press



(a) (b)

Fig. 1. (a) The shape function for $\mathcal{S}$ enclosed by Piet Hein's superellipse, $s(x, y) = |\frac{x}{a}|^r + |\frac{y}{b}|^r - 1$ with $a = b = 7$ and $r = 2.5$. The boundary is shown in white. (b) The level set curves for the corresponding shape function with the boundary shown by the dashed line.

are many choices for this measure, we use the definition

$$\phi_{\mathcal{S}}(\mathbf{q}) = \sum_i f(q_i). \tag{2}$$

Thus, the shape discrepancy function provides a measure of how close the team is to $\partial\mathcal{S}$.

4.2.3. Controller. For the system of N robots with dynamics given by Eq. (1), consider the following feedback policy for each robot

$$u_i = -\nabla_i f(q_i) - c\, v_i - \sum_{j\, s.t.\, A^d_{ij}=1} \nabla_i g_{ij}(\|q_{ij}\|) \tag{3}$$

where $c > 0$ is a constant scalar and $q_{ij} = q_i - q_j$. The first term of (3) drives the agent toward the desired curve while the second term adds damping to the system. The function $g_{ij}(\|q_{ij}\|)$ in the third term is an artificial potential function whose gradient models the interactions between each robot and its neighbors in the neighborhood given by $\mathcal{B}_i$. In the remainder of the paper, we will denote $g_{ij}(\|q_{ij}\|)$ as simply g_{ij}. Lastly, ∇_i denotes the partial derivatives with respect to the coordinates of the ith robot.

When solving problem **[P1]** (see Section 3), consider the following candidate function for g_{ij}:

$$g_{ij} = \frac{1}{(\|q_{ij}\|)^{k_1}} \tag{4}$$

where the postive even scalar k_1 is chosen such that the interaction forces are negligible when $\|q_{ij}\| > \delta$ and repulsive when $0 < \|q_{ij}\| \leq \delta$, thus ensuring collisions do not occur. Similarly, for problem **[P2]** (see Section 3), where the maintenance of a specific proximity graph is required, consider the following candidate function:

$$g_{ij} = \frac{1}{(\Delta - \|q_{ij}\|)^{k_2}} \tag{5}$$

where the positive even scalar k_2 is chosen such that the interaction forces are negligible when $\|q_{ij}\| < \Delta - \epsilon$ and attractive when $\Delta - \epsilon \leq \|q_{ij}\| \leq \Delta$, with $0 < \epsilon < \Delta$.

Figure 2 shows some candidate functions for g_{ij} with the corresponding $\nabla_i g_{ij}$. It is important to note that both (4) and (5) result in gradients of the form $\nabla_i g_{ij} = -\frac{dg_{ij}}{d\|q_{ij}\|} q_{ij}$ and thus $\nabla_i g_{ij} = -\nabla_j g_{ij}$. Finally, we note that the desired interconnection topology, $\mathcal{G}^d$, for problem **[P1]** is a position dependent graph while $\mathcal{G}^d$ for problem **[P2]** is a static graph whose edges are determined by the inter-agent constraints specified *a priori*.

5. Analysis

In this section, we study the stability and convergence properties of the controller given by Eq. (3) for the system of N robots each with dynamics given by Eq. (1). The system is in equilibrium when $\dot{\mathbf{q}} = \mathbf{0}$ and $\dot{\mathbf{v}} = \mathbf{0}$ or equivalently

$$\begin{aligned} \dot{q}_i &= v_i = 0 \\ \dot{v}_i &= -\nabla f(q_i) - c\, v_i - \sum_{j\, s.t.\, A^d_{ij}=1} \nabla_i g_{ij} = 0 \end{aligned} \tag{6}$$

where $c > 0$ for all $i = 1, \ldots, N$.

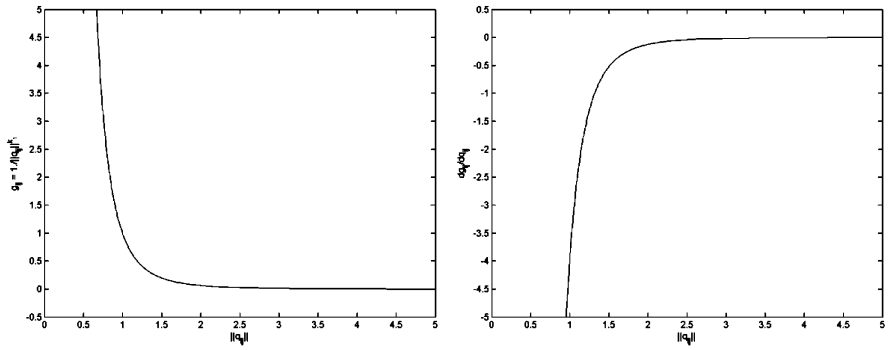
Our first lemma shows that the equilibrium points of the N robot system are extremal points of the shape discrepancy function.

Lemma 5.1. *For a system of N robots each with dynamics (1), shape function, f, and control (3), the set of equilibrium points satisfy the necessary condition for the shape discrepancy function to be at an extremum.*

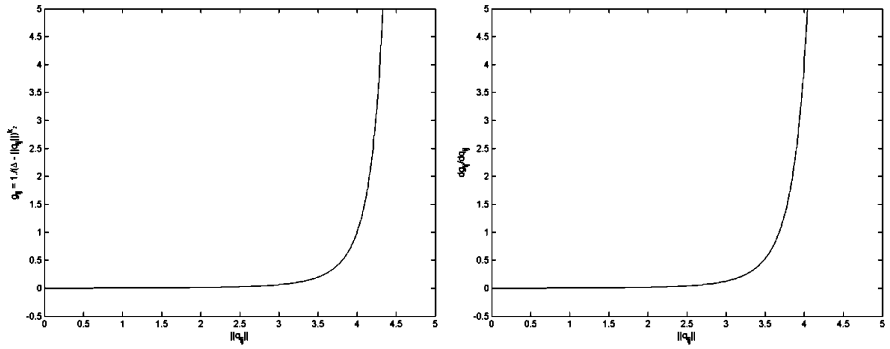
proof. When the system is in equilibrium, (6) simplifies to

$$u_i = -\nabla f(q_i) - \sum_{j\, s.t.\, A^d_{ij}=1} \nabla_i g_{ij} = 0$$

https://doi.org/10.1017/S0263574708004323 Published online by Cambridge University Press



(a) $k = 4$ (b) $k = 4$



(c) $k = 4$, $\Delta = 5$ (d) $k = 4$, $\Delta = 5$

Fig. 2. (a) A potential function of the form given by Eq. (4) with $k = 4$. (b) Gradient of the potential function shown in (a) with respect to $\|q_{ij}\|$. (c) A potential function of the form given by Eq. (5) with $k = 4$ and $\Delta = 5$. (b) Gradient with respect to $\|q_{ij}\|$ of the potential function shown in (c).

Recall $\nabla_i f = 2s\nabla_i s$ and as such, when summed over all agents, we obtain

$$\sum_i u_i = \sum_i \left(2s(q_i)\nabla s(q_i) + \sum_{j\, s.t.\, A^d_{ij}=1} \nabla_i g_{ij} \right) = 0.$$

Since $\nabla_i g_{ij} = -\nabla_j g_{ij}$, the second term sums to zero, resulting in

$$\sum_i u_i = \sum_i s(q_i)\nabla s(q_i) = 0.$$

This is identically the necessary condition for the shape discrepancy function to be at an extremum, i.e.,

$$\nabla\phi_{\mathcal{S}}(\mathbf{q}) = \nabla \sum_i f(q_i) = \sum_i s(q_i)\nabla s(q_i) = 0. \qquad (7)$$

□

The next proposition concerns the stability of the system. To show that our proposed controller is stable, consider the following positive semi-definite function:

$$E(\mathbf{q}, \mathbf{v}) = \phi_{\mathcal{S}}(\mathbf{q}) + \sum_i \sum_{j\, s.t.\, A^d_{ij}=1} g_{ij} + \frac{1}{2}\mathbf{v}^T\mathbf{v}. \qquad (8)$$

One can interpret E as an artificial energy function for the system. In this next proposition, we will use this function to show that the proposed controller drives the system toward a stable equilibrium configuration.

Proposition 5.2. *Given the set $\mathcal{S}$ whose boundary is given by the closed, smooth curve $s(x, y) = 0$, consider the system of N robots with dynamics (1), each with feedback control law (3). For any initial condition given by $(\mathbf{q}_0, \mathbf{v}_0) \in \Omega_0$, where $\Omega_0 = \{(\mathbf{q}, \mathbf{v}) \in \mathbf{X} | E(\mathbf{q}, \mathbf{v}) \leq e_0\}$ with $e_0 > 0$, the system converges to some invariant set, $\Omega_I \subset \Omega_0$, such that the points in Ω_I minimize the shape discrepancy function.*

Proof. We begin by showing the set Ω_0 is compact. Given e_0, the set Ω_0 is closed by continuity of E. To show boundedness, given $E \leq e_0$, we can conclude that both $(\phi_{\mathcal{S}} + \sum_i \sum_j g_{ij}) \leq e_0$ and $\mathbf{v}^T\mathbf{v} \leq e_0$. Moreover, $\phi_{\mathcal{S}} \leq e_0$, which implies $f(q_i) \leq e_0$ for all $i = 1, \ldots, N$. Since the shape function f is a radially unbounded function, $f(q_i) \leq e_0$ implies bounded $\|q_i\|$ and consequently bounded $\|q_{ij}\|$ when $\sum_i \sum_j g_{ij} \leq e_0$. We note this is not always true in the general case where bounded g_{ij} only implies bounded $\|q_{ij}\|$. Lastly, given $c\mathbf{v}^T\mathbf{v} \leq e_0$, then $\|\mathbf{v}\|$ is bounded by $\sqrt{e_0}/c$. Thus, Ω_0 is compact.

The time derivative of E is given by

$$\dot{E} = \sum_i \left(\nabla f(q_i)^T \dot{q}_i + v_i^T \dot{v}_i \right) + \sum_i \sum_{j\, s.t.\, A^d_{ij}=1} \nabla_i g_{ij}^T \dot{q}_i$$

https://doi.org/10.1017/S0263574708004323 Published online by Cambridge University Press

Recall $\dot{q}_i = v_i$ and $\dot{v}_i = u_i$ which is given by (3). Substituting these into the above equation results in

$$\begin{aligned}\dot{E} &= \sum_i \nabla f(q_i)^T v_i + \sum_i v_i^T \left(-\nabla f(q_i) - c\, v_i - \sum_{j\ s.t.\ A_{ij}^d=1} \nabla_i g_{ij} \right) \\ &\quad + \sum_i \sum_{j\ s.t.\ A_{ij}^d=1} \nabla_i g_{ij}^T v_i \\ &= \sum_i \nabla f(q_i)^T v_i - \sum_i v_i^T \nabla f(q_i) - c \sum_i v_i^T v_i \\ &\quad - \sum_i \sum_{j\ s.t.\ A_{ij}^d=1} v_i^T \nabla_i g_{ij} + \sum_i \sum_{j\ s.t.\ A_{ij}^d=1} \nabla_i g_{ij}^T v_i \\ &= -c\, \mathbf{v}^T \mathbf{v}.\end{aligned}$$

Recall from Section 4.2, c is a positive scalar constant used to add damping to the system. Then, $\dot{E} = 0$ if and only if $\mathbf{v} = \mathbf{0}$. By LaSalle's Invariance Principle, for any initial condition in Ω_0, the system of N agents with dynamics (1), converges asymptotically to the largest invariant set Ω_I, where $\Omega_I = \{(\mathbf{q}, \mathbf{v}) \in \mathbf{X} | \dot{E}(\mathbf{q}, \mathbf{v}) = 0\}$ and $\Omega_I \subset \Omega_0$.

Furthermore, Ω_I contains all equilibrium points in Ω_0 and as such, by Lemma 5.1, satisfy the necessary condition for $\phi_{\mathcal{S}}$ to be at an extremum. □

The above proposition states the convergence of the system of N agents to equilibrium points that satisfy the necessary condition for $\phi_{\mathcal{S}}$ to be at an extremum, however, it does not guarantee the final positions of the robots to be on $\partial\mathcal{S}$. To show this, we begin with the following proposition which shows the set $\Omega_{\mathcal{S}}$, defined as

$$\Omega_{\mathcal{S}} = \{(\mathbf{q}, \mathbf{v}) | s(q_i) = 0, \quad v_i = 0, \quad \nabla_i g_{ij} = 0 \quad \forall i\},$$

is a stable subset of Ω_I.

Proposition 5.3. *Consider the system of N robots each with dynamics (1) and feedback control (3). For convex g_{ij}, the set $\Omega_{\mathcal{S}}$ is a stable submanifold and $\Omega_{\mathcal{S}} \subset \Omega_I$.*

Proof. From (8) the system's artificial potential energy, $\mathcal{P}$, is given by

$$\mathcal{P} = \phi_{\mathcal{S}} + \sum_i \sum_{j\ s.t.\ A_{ij}^d=1} g_{ij}$$

To show that $\Omega_{\mathcal{S}}$ is a stable submanifold, it suffices to show that the Hessian of $\mathcal{P}$, $\mathcal{H}_{\mathcal{P}}$, is positive semi-definite on ∂S. $\mathcal{H}_{\mathcal{P}}$ is given by

$$\mathcal{H}_{\mathcal{P}} = \left(\mathbf{H}_\phi^I + \mathbf{H}_\phi^{II}\right) + \sum_i \sum_{j\ s.t.\ A_{ij}^d=1} \mathbf{H}_{g_{ij}} \tag{9}$$

$\mathbf{H}_\phi^I$ and $\mathbf{H}_\phi^{II}$ are $2N \times 2N$ block diagonal matrices with $\nabla_i s(q_i) \nabla_i s(q_i)^T$ and $s(q_i) H(q_i)$ respectively on the ith diagonal block with $H(q_i)$ defined as the 2×2 matrix of partial derivatives of $s(q)$ evaluated at q_i. $\mathbf{H}_{g_{ij}}$ is a $2N \times 2N$ matrix with $\frac{\partial^2 g_{ij}}{\partial q_i^2}$ and $\frac{\partial^2 g_{ij}}{\partial q_j^2}$ in the i, i and j, j entries, $\frac{\partial^2 g_{ij}}{\partial q_i \partial q_j}$ in the i, j and j, i entries, and zero everywhere else. We note that for convex g_{ij}, all $\mathbf{H}_{g_{ij}}$ are symmetric positive semi-definite matrices. Since $s(q_i) = 0$ for all $q_i \in \partial\mathcal{S}$, $\mathcal{H}_{\mathcal{P}}$ is the sum of positive semi-definite matrices and therefore positive semi-definite. Thus, the set $\partial\mathcal{S}$ is a stable submanifold. Furthermore, when $\mathbf{q} \in \partial\mathcal{S}$ and $\dot{q}_i = \dot{v}_i = 0$ with $\nabla_i g_{ij} = 0$ for all i, then $(\mathbf{q}, \mathbf{v}) \in \Omega_{\mathcal{S}} \subset \Omega_I$ by Proposition 5.2. □

We note that Lemma 5.1 and Propositions 5.2 and 5.3 can be extended to include boundaries defined by a collection of disjoint convex regions.

The following propositions concern the convergence of the team to the desired boundary for different desired interconnection topologies. We begin with the case when no interconnection topology is imposed, i.e., no inter-agent constraints.

Proposition 5.4. *For any smooth star shape, $\mathcal{S}$, the system of N robots each with dynamics (1), control law (3) with $g_{ij} = 0$ for all i, j, the system converges to $\Omega_{\mathcal{S}} \equiv \partial\mathcal{S}$ for any initial condition in Ω_0.*

Proof. For any desired circular boundary centered around $\hat{q}$, the boundary can be parameterized by the following implicit function

$$s(q) = \|q - \hat{q}\| - R = 0$$

with $f(q) = s^2(q)$ as the corresponding shape function. The system equilibrium condition is given by (6) which simplifies to $\nabla_i f(q_i) = 0\ \forall i$ when all $g_{ij} = 0$. Furthermore, for any initial condition in Ω_0, the system converges to the invariant set Ω_I. For the circular boundary, $\Omega_I \equiv \Omega_{\mathcal{S}} \equiv \partial\mathcal{S}$. By Proposition 5.3, $\partial\mathcal{S}$ is stable and by Proposition 5.2, the system converges asymptotically toward the circular boundary.

For any smooth star shape, $\bar{\mathcal{S}}$, there exists a diffeomorphic transformation that maps the boundary of the star shape onto the boundary of the circle given by $s(q)$ and the interior and exterior points of the star boundary to interior and exterior points of the circular boundary.[35] Since such a diffeomorphic map exists, stationary points are diffeomorphically mapped between the circular and the star boundary. Thus, from Lemma 5.1, the system is in equilibrium when $q_i \in \partial\bar{\mathcal{S}}$ for all i. And from Proposition 5.2, the system converges asymptotically toward $\partial\bar{\mathcal{S}}$. □

In the remainder of the section, we prove the convergence of N robots to the desired boundary curve for two desired interconnection topologies. These desired interconnection topologies can be either static proximity graphs for maintaing team cohesion or dynamic position dependent proximity graphs for collision avoidance. In both scenarios, the edges of the proximity graphs will represent the constraints that need to be maintained. We begin with the convergence of the team to $\partial\mathcal{S}$ with g_{ij} given by Eq. (4) and remind the reader of the key assumptions outlined in Section 4.1.

1. $N < N_{max}$;
2. $\rho_{min} > \delta$;
3. $\min_{s \in [\frac{\pi\rho_0}{2}, L - \frac{\pi\rho_0}{2}]} \|q_0(s) - q(s)\| > \delta$ for any $q_0(s) \in \partial\mathcal{S}$, where $s \in [0, L]$ denotes the arclength and ρ_0 denotes the radius of curvature at q_0.

https://doi.org/10.1017/S0263574708004323 Published online by Cambridge University Press

Proposition 5.5. *Given a smooth star-shaped set, $\mathcal{S}$, and the system of N robots, each with dynamics (1) and control law (3), such that N and $\partial\mathcal{S}$ satisfy the above assumptions, then the system with only repulsive interaction forces under arbitrary interconnection topologies and initial conditions in Ω_0, can only be in stable equilibrium if $q_i \in \partial\mathcal{S}$ for all i.*

Proof. Recall, that the system equilibrium condition is given by

$$\nabla_i f = -\sum_{j\,s.t.\,A^d_{ij}=1} \nabla_i g_{ij} \quad \forall i = 1, \ldots, N.$$

Assume N and $\partial\mathcal{S}$ satisfy the above assumptions and the system of N robots is in stable equilibrium such that not all $q_i \in \partial\mathcal{S}$, i.e., $s(q_i) \neq 0$ for some i's. Without loss of generality, assume $\mathcal{S}$ is centered about $\hat{q}$ and let $\theta_i \in [0, 2\pi)$ denote the angle between the vector $(q_i - \hat{q})$ and the horizontal axis. Let q_N denote the agent with the maximum value of θ_i in the team. Then $\theta_j < \theta_N$ for all $q_j \in \mathcal{B}_N$.

For every q_i, we choose a body-fixed coordinate frame such that the basis is given by unit vectors in the directions of $(q_i - \hat{q})$ and $(q_i - \hat{q})^\perp$. Then for every $\nabla_N g_{Nj}$ for q_N, we denote the component of $\nabla_N g_{Nj}$ in the direction of $(q_N - \hat{q})$ as $(\nabla_N g_{Nj})_\parallel$ and the component of $\nabla_N g_{Nj}$ in the direction of $(q_N - \hat{q})$ as $(\nabla_N g_{Nj})_\perp$. Since θ_N is the maximum θ for all N agents, $\sum_{j\,s.t.\,A^d_{Nj}=1} \nabla_N g_{Nj}$ would result in a net force on q_N that would push q_N away from its neighbors in $\mathcal{B}_N$. In other words, the net force from the neighbors of q_N would result in pushing q_N in the direction given by $(q_N - \hat{q})^\perp$ such that θ_N would increase. Thus, for q_N to be in stable equilibrium, $\nabla_N f$ must have a component that is equal and opposite in the $(q_N - \hat{q})^\perp$ direction. For this to happen, either $\rho_{min} < \delta$ or $\min_{s\in[\frac{\pi\rho_0}{2}, L-\frac{\pi\rho_0}{2}]} \|q_0(s) - q(s)\| < \delta$ which violates our assumption on $\partial\mathcal{S}$ since the radius of curvature increases monotonically as one moves away from the boundary. Thus, the system can only be in stable equilibrium when $q_i \in \partial\mathcal{S}$ for all i. □

We note that while it may be possible that the net repulsive forces exerted on a particular agent by its neighbors sum to zero, this does not imply the system as a whole is in equilibrium. As shown in the previous proposition, such stable configurations would result in violating our assumptions on $\partial\mathcal{S}$. Our final proposition proves the convergence of the team to convex boundaries while maintaining a desired proximity graph where the edges of $\mathcal{G}^d$ are specified *a priori* and g_{ij} are of the form given by Eq. (5).

Proposition 5.6. *For any smooth convex shape, $\mathcal{S}$, the system of N robots with dynamics (1), control law (3), a tree $\mathcal{G}^d$ where the edges represent attractive forces, and initial conditions in Ω_0, can only be in stable equilibrium if $q_i \in \partial\mathcal{S}$ for all i.*

Proof. Once again, the equilibrium condition is given by

$$\nabla_i f = -\sum_{j\in N_i} \nabla_i g_{ij} \qquad \forall i = 1, \ldots, N.$$

Assume the system of N robots is in stable equilibrium such that not all $q_i \in \partial\mathcal{S}$, i.e., $s(q_i) \neq 0$ for some i's. Denote the level set of f evaluated at q_i as $\hat{s}_{q_i}$. Since $\partial\mathcal{S}$ is convex, the level sets of f are also convex. Furthermore, for any $q_i \in \mathbb{R}^2$, $\hat{s}_{q_i}$ lies entirely to one side of the tangent line defined by $\nabla_i f^\perp$. Additionally, since the level sets do not intersect, given q_i and q_j such that $s(q_i) > 0$ and $s(q_j) > 0$, $s(q_j) > s(q_i)$ implies q_j lies outside the level set $\hat{s}_{q_i}$. Similarly, given q_i and q_j such that $s(q_j) > s(q_i)$ implies q_i lies outside the level set $\hat{s}_{q_j}$.

Since the system is in stable equilibrium, for every q_i there exists a q_j such that $A_{ij} = 1$, $f(q_j) > f(q_i)$, and q_j lies in the halfplane, defined by $\nabla_i f^\perp$, that does not contain s_{q_i}. Define $q_N = \arg\max_i f(q_i)$. If $s(q_N) > 0$, then for q_N to be in stable equilibrium, there must exist a q_k such that $A_{Nk} = 1$, $f(q_k) > f(q_N)$, and q_k lies in the halfplane that does not contain s_{q_N}. This contradicts the definition of q_N and thus the system cannot be in equilibrium.

If $s(q_N) < 0$, we must show that the equilibrium configuration cannot be a stable one if the desired interconnection topology is a tree. To achieve this, consider any leaf node k, the equilibrium condition for q_k is given by $\nabla_k f = -\nabla_k g_{kl}$ with $A_{kl} = 1$. This implies $\nabla_k f = -\nabla_l f$ for every leaf node k. This resulting configuration is unstable since any slight perturbation about the point $(q_k - q_l)/2$ results in non-zero velocities for agents k and l.

If $s(q_N) = 0$, this implies $s(q_i) = 0$ for all $i \in \{1, \ldots, N-1\}$. Thus, equilibrium can only be reached when $q_i \in \partial\mathcal{S}$ for all i.

To show that $\nabla_i g_{ij} = 0$ for all i, j pairs, again consider the equilibrium condition of any leaf vertex q_u in $\mathcal{G}^d$ given by

$$\nabla_u f = -\nabla_u g_{uv} \tag{10}$$

where v denotes a neighbor of q_u such that $A_{uv} = 1$. Since the only equilibrium is when all q_i are on $\partial\mathcal{S}$, $\nabla_u f = 0$ and thus $\nabla_u g_{uv} = 0$ for all leaf vertices. Further, since $\nabla_i g_{ij} = -\nabla_j g_{ij}$, then $\nabla_v g_{uv} = 0$ for every neighbor q_v of each leaf vertex q_u.

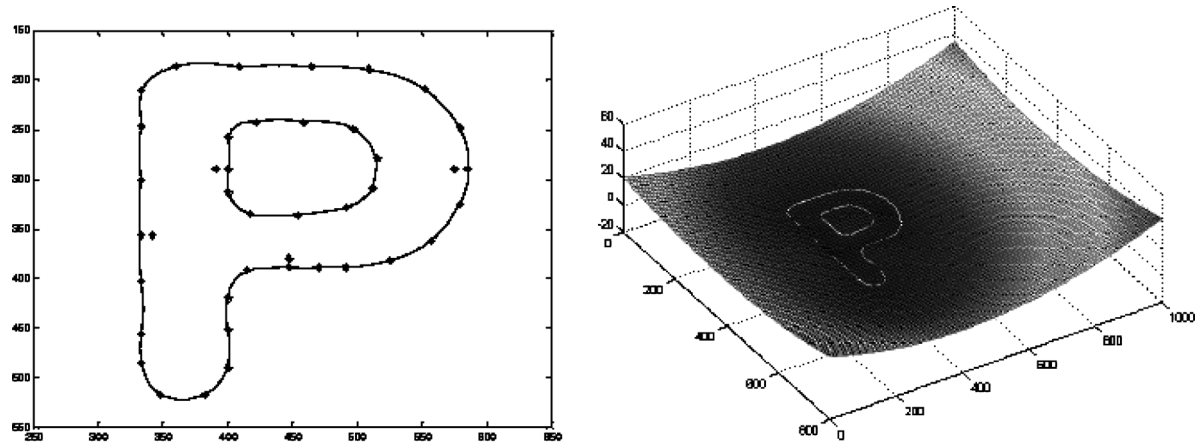
Denote $\mathcal{V}_1$ as the set of leaf vertices of $\mathcal{G}^d$ and consider the subgraph $\mathcal{G}^d_1 = \mathcal{G}^d \setminus \mathcal{V}_1$. Since $\mathcal{G}^d$ is a tree, $\mathcal{G}^d_1$ is also a tree. Then $\nabla_v g_{uv} = 0$ for each neighbor q_v of every $q_u \in \mathcal{V}_1$ which implies that every leaf vertex q_m of $\mathcal{G}^d_1$, must also have equilibrium conditions of the form given by Eq. (10). Thus, on the boundary, for each neighbor q_n of q_m, $\nabla_m g_{mn} = -\nabla_n g_{mn} = 0$. By induction, we can conclude that $\nabla_i g_{ij} = 0$ for all i, j pairs. □

The above proof can be extended to show convergence to star shaped $\partial\mathcal{S}$ if we require the largest radius of curvature of $\partial\mathcal{S}$, denoted by ρ_{max}, satisfies the condition $\rho_{max} < \Delta$. Furthermore, the above proof can be extended for arbitrary $\mathcal{G}^d$, however the inter-agent constraints are not guaranteed to be satisfied, i.e. $\nabla_i g_{ij} \neq 0$.

6. Toward General 2D Shapes

Often times there are applications which may require the team of robots to converge to boundaries of shapes where closed form parameterizations are not always available.

https://doi.org/10.1017/S0263574708004323 Published online by Cambridge University Press



(a) (b)

Fig. 3. Implicit function for a letter "P". (a) Constraint points used to generate the function. (b) Shape function with the zero isocontour depicted in white.

Under these circumstances, one approach is to synthesize complex boundaries through the interpolation of a set of points located along the desired boundary curve. We refer to these points as *constraint points*. Toward this end, we propose the synthesis of *general shape functions* using radial basis functions (RBFs).

A radial basis function can be described in terms of a center point q_c and a function $h(r)$, where r defines the Euclidean distance between any point q and the center q_c. The term radial comes from the fact that h evaluates exactly the same for all points at a fixed radius from q_c. Then given a set of m constraint points, q_{c_k}, the general shape function, f, is given by

$$f(q) = \sum_k w_k h(\|q - q_{c_k}\|). \tag{11}$$

where w_k denotes scalar weights.

In this work, we choose $h(r) = r^2 log(r)$ to be our candidate RBF. To generate a specific shape function, we first specify a set of constraint points q_{c_k}, for $k = 1, \ldots, m$, along the desired boundary curve. Additionally, we will specify at least one constraint point inside and/or outside the desired boundary which will allow us to avoid degenerate solutions when obtaining the weights, w_k. Then, the weights for each RBF, is determined by solving a system of linear equations of the form $\mathbf{Mw} = \mathbf{b}$ where $\mathbf{M}$ is an $m \times m$ matrix of coefficients whose ijth entry is given by $h(\|q_{c_i} - q_{c_k}\|)$, $\mathbf{w}$ denotes the vector of weights, and $\mathbf{b}$ is a vector of scalars whose ith entry is chosen to assume the desired value of $f(q_{c_i})$. This is a common approach used in computer graphics for generating three dimensional models and computer animations.[36,37]

It is important to note that unlike the shape function defined in Section 4.2, the general shape function obtained via this construction is only guaranteed to be zero at the chosen constraint points along the boundary. Thus, this approach, by construction, may result in trapping robots in regions other than the desired boundary, i.e., regions with local minima. However, this can often be resolved by adding new constraint points that do not alter the zero isocontour but effectively change the function's overall gradient. Nevertheless, the main advantage of this approach is the flexibility it affords when synthesizing general shape functions since complex two dimensional boundaries can be generated by interpolating an adequate number of constraint points. Lastly, the resulting surfaces generated by the radial basis functions are smooth by construction, thus allowing the use of the gradient controllers given by Eq. (3).

The boundary of a 'P' shape synthesized using this approach is shown in Fig. 3. The function was generated using the constraint points shown in Fig. 3(a).

7. Simulations

We illustrate the algorithm presented in the previous sections with some simulation results. We begin with the case when $\mathcal{S}$ is star and consider teams consisting of approximately 40 robots to demonstrate the scalability of the algorithm. Figure 4 shows the initial and final positions and the trajectories of the team for: (i) no interactions, $g_{ij} = 0$; (ii) collision avoidance only and g_{ij} given by Eq. (4); (iii) proximity maintenance only with a path graph as the desired proximity graph, i.e., robot i maintains constraints with robots $i - 1$ and $i + 1$ and no constraints between robots 1 and N, and g_{ij} given by Eq. (5); and (iv) both collision avoidance and proximity maintenance with a path graph as the the desired proximity graph and g_{ij} given by the sum of Eq. (4) and (5). In these results, we chose $\delta = 2$, $\Delta = 10$, and $k_1 = k_2 = 4$. Note the difference in the final positions resulting from the different g_{ij} choices.

For boundaries where close form solutions are not available, we provide simulation results using the approach discussed in Section 6. The function depicted in Fig. 5 was created using 95 constraint points and the zero isocontour is given by the disjoint boundaries of the letters L, U, I, and Z. Figure 6 shows a simulation snapshot of 90 robots spreading out along this isocontour. For the simulation depicted in Fig. 7, we first generated implicit functions forming the letters G, R, A, S, P and achieved dynamic shape change by switching to the desired shape function once the initial

https://doi.org/10.1017/S0263574708004323 Published online by Cambridge University Press

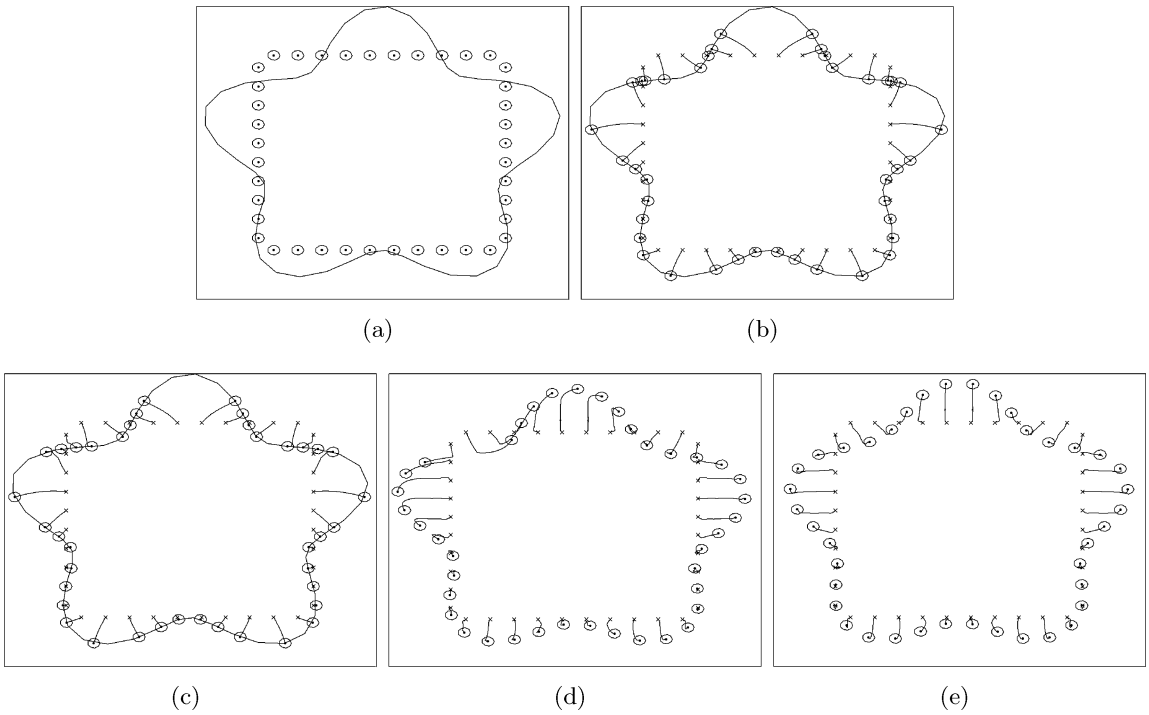


Fig. 4. A 40-robot team converging to the star shaped boundary, denoted by the black solid line in (a), using the control law given (3) with $\delta = 2$, $\Delta = 10$, $k_1 = k_4 = 4$. The solid circles represent the robots and the empty circles denote the circular region around the robot defined with radius δ. Robot trajectories are the solid lines connecting the solid circles and the $\times$'s used to denote the initial positions. (a) Initial position of the team with respect to the desired boundary. (b) Trajectories of the robots when $g_{ij} = 0$ for all i and j. (c) Trajectories of the robots when g_{ij} is given by (4), i.e., collision avoidance only. (d) Trajectories of the robots when g_{ij} is given by (5), i.e., proximity maintenance. The desired proximity graph is a path graph. (e) Trajectories of the robots when g_{ij} is the sum of Eqs. (4) and (5), i.e., collision avoidance and proximity maintenance.

pattern was achieved. Figure 7 shows the shape function for each letter and a snapshot of the simulation for each letter, in which the robots are represented by the small circles. Each of these functions was generated using an average of 40 constraint points. As expected the robots are attracted to and spread along the desired boundaries.

We note that when synthesizing patterns composed of disjoint unions of complex boundaries, the ability of the team to align themselves along the desired boundaries often depends on the initial positions of the individual robots since the convergence results obtained in Section 5 do not extend to these patterns. Consider the letter "P" shown in Fig. 3. If all robots start from outside the shape, they will have to overcome the minima imposed by the outer curve to reach the inner boundary of the shape. Here the minima is not an undesired local minima but rather a boundary that must also be reached. In the simulations presented here, the initial distribution of the robots is roughly uniform which allowed the team to converge to both boundaries. In general, local minima situations may potentially be resolved using approaches such as random exploration.

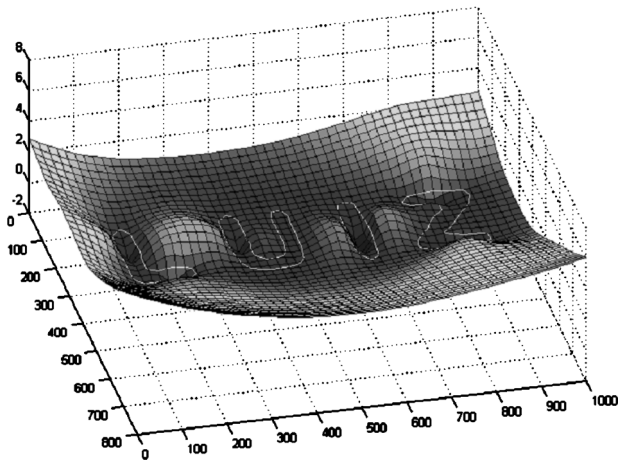


Fig. 5. Function composed of 95 radial basis functions that has the string "LUIZ" as its zero isocontour.

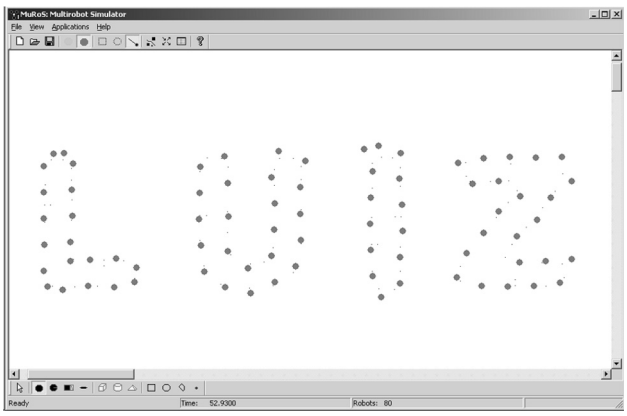


Fig. 6. Snapshot of a simulation where 90 robots converge and spread along the isocontour depicted in the previous figure.

https://doi.org/10.1017/S0263574708004323 Published online by Cambridge University Press

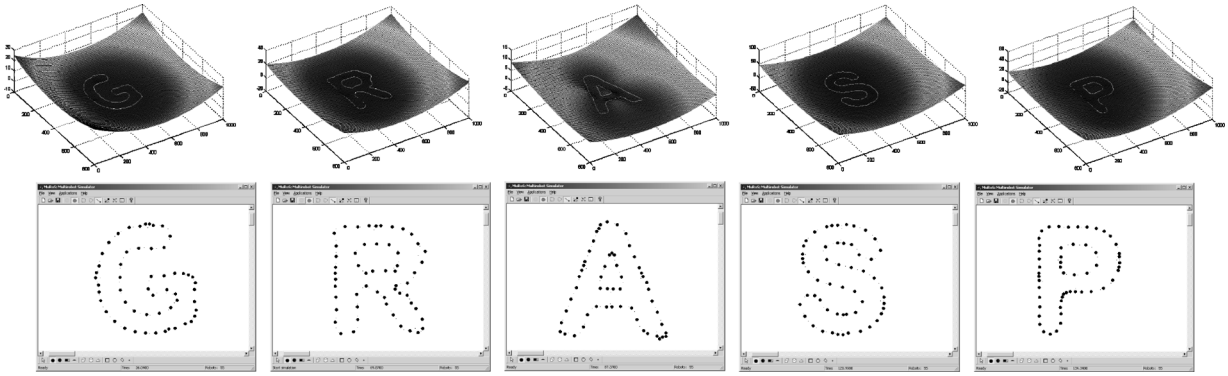


Fig. 7. Simulations of 55 robots tracking different functions to form the letters G, R, A, S, P. The figures on the top row show the functions generated for each letter, while the figures on the bottom row show snapshots of the simulation (robots are the small circles).

8. Conclusions

We have presented decentralized controllers for generating formations that conform to specified two dimensional patterns with constraints on proximity. These controllers can be used to deploy multiple robots to surround buildings or fenced off areas, or to self-assemble robots to build a two dimensional structure. The algorithm was shown to be stable and convergence to the boundary was established for star shapes in both the absence and the presence of inter-agent interactions. Convergence to the boundary of more general shapes with different interactions was shown in simulation.

One direction for future work includes the extension of our convergence results to the boundaries given by the union of disjoint convex sets and time-varying boundaries. We also acknowledge the need to implement sensing on individual robots to obtain local state information. It may not be reasonable to expect small, resource-constrained robots to be able to sense their individual states. However, it is difficult to get robots to perform tasks like pattern generation in a fixed coordinate frame without a hardware (or software) solution to the localization problem. The important point is that the robots need only local state information and the algorithm is completely decentralized.

Acknowledgements

The authors would like to thank Drs. Adam Halasz and Savvas Loizou from the University of Pennsylvania for discussions on stability and convergence. We would also like to thank our reviewers for their valuable comments. We gratefully acknowledge the support of ARO Grants W911NF-05-1-0219, W911NF-04-1-0148 and CNPq grants 305127/2005-5 and 490743/2006-4.

References

1. N. F. Britton, N. R. Franks, S. C. Pratt and T. D. Seeley, "Deciding on a New Home: How Do Honeybees Agree?," *Proceeding of Royal Society of London B*, vol. 269, 2002, pp. 1383–1388.
2. S. C. Pratt, "Quorum sensing by encounter rates in the ant temnothorax albipennis," *Behav. Ecolo.* **16**(2), (2005) pp. 448–496.
3. I. D. Couzin, J. Krause, R. James, G. D. Ruxton and N. R. Franks, "Collective memory and spatial sorting in animal groups," *J. Theo. Bio.* **218**, (2002) pp. 1–11.
4. J. K. Parrish, S. V. Viscido and D. Grunbaum, "Self-organized fish schools: An examination of emergent properties," *Biol. Bull.* **202**, (2002) pp. 296–305.
5. C. W. Reynolds, "Flocks, Herds and Schools: A Distributed Behavioral Model," *Proceedings of the 14th annual conference on Computer Graphics (SIGGRAPH'87)*, (ACM Press, 1987), Baltimore, Maryland, USA pp. 25–34.
6. T. Vicsek, A. Czirok, E. Ben-Jacob, I. Cohen and O. Shochet, "Novel type of phase transition in a system of self-driven particles," *Phys. Rev. Lett.* **75**(6), 1226–1229 (1995).
7. A. Jadbabaie, J. Lin and A. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Trans. Autom. Control*, July 2003 **48**(6), pp. 988–1001.
8. H. G. Tanner, A. Jadbabaie and G. J. Pappas, "Flocking in fixed and switching networks," *Trans. Autom. Control* **52**(5), 863–868 (2007).
9. O. Albayrak, *Line and Circle Formation of Distributed Autonomous Mobile Robots with Limited Sensor Range* (PhD thesis, Naval Postgraduate School, Monterey, CA, 1996).
10. I. Suzuki and M. Yamashita, "Distributed anonymous mobile robots: Formation of geometric patterns," *SIAM J. Comput.*, **28**(4), 1347–1363 (1999).
11. J. P. Desai, J. P. Ostrowski and V. Kumar, "Modeling and control of formations of nonholonomic mobile robots," *IEEE Trans. Robot. Autom.* **17**(6), 905–908 (2001).
12. R. Fierro, P. Song, A. Das and V. Kumar, "Cooperative control of robot formations," *Cooperative Control and Optimization: Series on Applied Optimization* (R. Murphey and P. Paradalos, eds.) (Kluwer Academic Press, 2002) pp. 79–93.
13. H. G. Tanner and A. Kumar, *Formation Stabilization of Multiple Agents Using Decentralized Navigation Functions*. (MIT Press, 2005).
14. S. G. Loizou and K. J. Kyriakopoulos, "A feedback based multiagent navigation framework," *Int. J. Syst. Sci.* **37**(6), 377–384 (2006).
15. P. Ogren, E. Fiorelli and N. E. Leonard, "Formations with a Mission: Stable Coordination of Vehicle Group Maneuvers," *Proceedings of 15th International Symposium on Mathematical Theory of Networks and Systems*, Notre Dame, Illinois, USA (Aug. 2002) pp. 267–278.
16. P. Song and V. Kumar, "A Potential Field Based Approach to Multi-Robot Manipulation," *Proceedings of IEEE International Conference on Robotics and Automation*, Washington, DC (May 2002) pp. 1217–1222.
17. G. A. S. Pereira, V. Kumar, and M. F. M. Campos, "Decentralized algorithms for multi-robot manipulation via caging," *Int. J. Robot. Res. (IJRR)*, July-August 2004 **23**, pp. 783–795.
18. L. Chaimowicz, N. Michael and V. Kumar, "Controlling Swarms of Robots Using Interpolated Implicit Functions," *Proceedings of the 2005 International Conference on Robotics*

https://doi.org/10.1017/S0263574708004323 Published online by Cambridge University Press

and Automation (ICRA05), Barcelona, Spain, (2005) pp. 2487–2492.
19. N. Correll, S. Rutishauser and A. Martinoli, "Comparing Coordination Schemes for Miniature Robotic Swarms: A Case Study in Boundary Coverage of Regular Structures," *Proceedings of 10th International Symposium on Experimental Robotics (ISER) 2006*, Rio de Janeiro, Brazil, (2006) pp. 471–480.
20. N. Correll and A. Martinoli, "System Identification of Self-Organizing Robotic Swarms," *Proceedings of 8th Int. Symposium on Distributed Autonomous Robotic Systems (DARS) 2006*, Rio de Janeiro, Brazil (2006) pp. 31–40.
21. R. Sepulchre, D. Paley, and N. E. Leonard, "Stabilization of planar collective motion, part 1. All-to-all communication," *IEEE Trans. Autom. Control* **52**(5) 811–824 (2005).
22. D. Paley, N. E. Leonard and R. J. Sepulchre, "Collective Motion of Self-Propelled Particles: Stabilizing Symmetric Formations on Closed Curves," *Proceedings of the 45th IEEE Conference on Decision and Control* (Submitted), San Diego, CA, (2006) pp. 5067–5072.
23. F. Zhang and N. E. Leonard, "Coordinated patterns of unit speed particles on a closed curve," *Syst. Control Lett.* **56**(6), 397–407 (2007).
24. F. Zhang, D. M. Fratantoni, D. Paley, J. Lund and N. E. Leonard, "Control of coordinated patterns for ocean sampling," *Int. J. Control* **80**(7), 1186–1199 (2007).
25. A. Bertozzi, M. Kemp and D. Marthaler, "Determining environmental boundaries: Asynchronous communication and physical scales," *Coop. Control*, (2004) vol. 309, pp. 25–42.
26. W. Kerr and D. Spears, "Robotic Simulation of Gases for a Surveillance Task," *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) 2005*, Edmonton, Alberta, Canada, (August 2005) pp. 2905–2910.
27. C. Belta, G. A. S. Pereira and V. Kumar, "Abstraction and Control of Swarms of Robots," *International Symposium of Robotics Research'03*, Sienna, Italy, (2003) pp. 224–233.
28. L. Chaimowicz and V. Kumar, "Aerial sheperds: Coordination Among Uavs and Swarms of Robots," *Proceedings of the 7th International Symposium on Distributed Autonomous Robotic Systems (DARS2004)*, Toulouse, France (2004) pp. 231–240.
29. N. Michael, C. Belta, and V. Kumar, "Controlling Three Dimensional Swarms of Robots," *Proceedings of IEEE International Conference on Robotics and Automation (ICRA) 2006*, Orlando, FL (April 2006) pp. 964–969.
30. F. Zhang, M. Goldgeier, and P. S. Krishnaprasad, "Control of Small Formations Using Shape Coordinates," *Proceedings of of 2003 International Conference of Robotics and Automation*, Taipei, Taiwan (2003) pp. 2510–2515.
31. J. Spletzer and R. Fierro, "Optimal Positioning Strategies for Shape Changes in Robot Teams," *Proceedings of the IEEE International Conference on Robotics and Automation*, Barcelona, Spain (Apr. 2005) pp. 754–759.
32. B. Nabet and N. E. Leonard, "Shape Control of a Multi-agent System Using Tensegrity Structures," *IFAC Workshop on Lagrangian and Hamiltonian Methods for Nonlinear Control*, Nagoya, Japan (2006) pp. 329–339.
33. M. A. Hsieh and V. Kumar, "Pattern Generation with Multiple Robots," *Proceedings of International Conference on Robotics and Automation (ICRA) 2006*, Orlando, FL (Apr. 2006) pp. 2442–2447.
34. M. P. do Carmo, *Differential Geometry of Curves and Surfaces*, Upper Saddle River, New Jersey (Prentic-Hall Inc., 1976).
35. E. Rimon and D. E. Koditschek, "Exact robot navigation using artificial potential functions," *IEEE Trans. Robot. Autom.* **8**, 501–518 (1992).
36. G. Turk and J. F. O'Brien, "Shape Transformation Using Variational Implicit Functions," *Proceedings of the 26th Annual Conference on Computer Graphics (SIGGRAPH 99)*, ACM Press/Addison-Wesley Publishing Co., Los Angeles, California, USA (1999) pp. 335–342.
37. G. Turk, H. Q. Dinh, J. F. O'Brien and G. Yngve, "Implicit Surfaces That Interpolate," *Proceedings of the International Conference on Shape Modeling & Applications*, IEEE Computer Society, Geneva, Italy (2001) pp. 62–73.

Appendix: The Determination of N_{max} and N_{min}

Given an interaction range or a fixed sensing radius $d > 0$, N_{min} and N_{max} can be determined purely geometrically. To accomplish this, begin by selecting an initial point on $\partial\mathcal{S}$ and placing a circle of radius d centered at this point. Next, select another point on $\partial\mathcal{S}$ by going along the boundary in a counter-clockwise direction. Select this next point such that a circle of radius d centered at this second point is tangent to the first circle. By repeating this process, one will be able to generate a sequence of circles centered at points on the boundary that are tangent to one another. Since the boundary and the d are both finite, one can continue this process until one can no longer fit a circle centered at a point on $\partial\mathcal{S}$ without intersecting any of the previous circles. Thus, the number of tangent circles drawn at the end of the process gives N_{max}. A similar procedure can be used to determine N_{min}.

https://doi.org/10.1017/S0263574708004323 Published online by Cambridge University Press