Analysis and design tools for distributed motion coordination

Jorge Cortés
School of Engineering
University of California at Santa Cruz
Santa Cruz, California 95064, USA
jcortes@soe.ucsc.edu

Sonia Martínez Francesco Bullo
Mechanical and Environmental Engineering
University of California at Santa Barbara
Santa Barbara, California 93106, USA
{smartine,bullo}@engineering.ucsb.edu

***Abstract*— This paper presents recently-developed theoretical tools for the analysis and design of coordination algorithms for networks of mobile autonomous agents. First, various motion coordination tasks are encoded into aggregate cost functions from Geometric Optimization. Second, the limited communication capabilities of the mobile agents are modeled via the notions of proximity graphs from Computational Geometry and of spatially distributed maps. Third, the algorithms correctness is established via advanced versions of the LaSalle Invariance Principle for non-deterministic systems in discrete and continuous time. Finally, we illustrate how to apply these tools in a variety of motion coordination problems such as deployment, rendezvous, and flocking.**

I. INTRODUCTION

Motion coordination is a remarkable phenomenon in biological systems and an extremely useful tool in man-made groups of vehicles, mobile sensors and embedded robotic systems. Just like animals do, groups of mobile autonomous agents need the ability to deploy over a given region, assume a specified pattern, rendezvous at a given point, or jointly move in a synchronized manner. These coordinations tasks are typically to be achieved with little available communication between the agents, and therefore, with limited information about the state of the entire system.

An important scientific motivation for the study of motion coordination is the analysis of emerging and self-organized behaviors in biological groups with distributed agent-to-agent interactions. At the same time, an important engineering reason to study motion coordination stems from the recent interest in sensor networks. Indeed, it is envisioned that groups of autonomous agents with computing, communication and mobility capabilities will soon become economically feasible and perform a variety of spatially-distributed sensing tasks such as search and rescue, surveillance, environmental monitoring, and exploration.

The objective of this paper is to illustrate ways in which systems theory helps us analyze emerging behaviors in animal groups and design autonomous and reliable robotic networks. Indeed, the interest of the control community for motion coordination has increased tremendously over the last few years. A necessarily incomplete list of works on distributed, or leaderless, motion coordination includes [1], [2] on pattern formation, [3] on flocking, [4] on self-assembly, [5] on swarm aggregation, [6] on gradient climbing, [7] on deployment, [8], [9], [10], [11] on rendezvous, [12] on cyclic pursuit, and [13], [14], [15] on consensus. This paper presents and surveys some recently-developed theoretical tools for modeling, analysis and design of motion coordination. The next paragraphs summarize the various sections.

Section II reviews the computational geometric notion of *proximity graph*. Proximity graphs of various kinds model agent-to-agent interactions that depend only on the agents' location in space. This is the case for example in wireless communication or in communication based on line-of-sight. Thus, the notion of proximity graph allows us to model the information flow between mobile agents. Useful examples include the disk and the visibility graphs. A coordination algorithm is said to be *spatially distributed* over a proximity graph if the control input of each agent can be computed only with the information encoded in the given graph.

The focus of Section III is on motion coordination tasks and on how to encode them into aggregate cost functions from Geometric Optimization. We discuss various aggregate cost functions for tasks such as deployment (area-coverage deployment, maximum detection likelihood deployment, and visibility-based deployment), rendezvous (via the diameter of convex hull function), cohesiveness, and consensus (via the so-called Laplacian potential from algebraic graph theory). We also discuss some results on their smoothness properties and extreme points via nonsmooth analysis.

Section IV highlights a recently-developed version of the LaSalle Invariance Principle for non-deterministic discrete-time dynamical systems. This is an example of a system-theoretic tool that is helpful in establishing stability and convergence of motion coordination algorithms. Non-determinism may arise because of different reasons including asynchronicity (the asynchronous, deterministic evolution of a mobile network may be subsumed into a larger set of synchronous, non-deterministic evolutions, e.g. [9]), design choices when devising the motion coordination algorithm (at each time instant throughout the evolution, each agent can choose among multiple possible control actions, as opposed to a single one, e.g. [7]) and communication, control and sensor errors in the agents during the execution of the coordination algorithm (e.g. [8], [11]).

Section V builds upon the tools introduced earlier and presents various approaches to the design and analysis of scalable motion coordination algorithms. A first approach is based on the design of gradient flows: here we typically are given a sensing task to be performed by the network and a proximity graph as communication constraint. A second approach is based on the analysis of emerging behaviors: in this case a notion of neighboring agents and an interaction law between them is usually given. We apply these approaches to numerous examples of coordination algorithms proposed in the literature.

Let us finally mention that, for reasons of space, the present exposition does not include a more in-depth discussion of various techniques that have been proved useful in motion coordination problems. Among them, we highlight ergodic [3] and circulant [12] matrices from matrix analysis, graph Laplacians and algebraic connectivity [3], [13] from algebraic graph theory, graph grammars [4], symmetries of differential equations [2], and nonsmooth and stability analysis for differential inclusions [16].

II. Spatially distributed maps over proximity graphs

In this section we introduce the notion of proximity graphs and of spatially distributed maps.

A. Basic geometric notions

A *partition* of a set S is a collection of subsets of S with disjoint interiors and whose union is S. Let $\mathbb{F}(S)$ be the collection of finite subsets of S. Given $S \subset \mathbb{R}^2$ and $\mathcal{P} \in \mathbb{F}(S)$ a set of n distinct points $\{p_1, \dots, p_n\}$ in S, the *Voronoi partition of* S generated by $\mathcal{P}$ with respect to the Euclidean norm $\|\cdot\|$ is the collection of sets $\{V_i(\mathcal{P})\}_{i\in\{1,\dots,n\}}$ defined by $V_i(\mathcal{P}) = \{q \in S \mid \|q-p_i\| \leq \|q-p_j\|\,,$ for all $p_j \in \mathcal{P}\}$. We usually refer to $V_i(\mathcal{P})$ as V_i. For a detailed treatment of Voronoi partitions we refer to [17], [18]. Unless otherwise stated, we usually deal with the Voronoi partition of $S = \mathbb{R}^2$.

For $p \in \mathbb{R}^2$ and $r \in \mathbb{R}_+ = (0, +\infty)$, let $B(p,r)$ and $\overline{B}(p,r)$ denote the open and closed ball in $\mathbb{R}^2$ centered at p of radius r, respectively. For $\mathcal{P} \in \mathbb{F}(S)$ with n elements, consider the collection $\{V_i(\mathcal{P}) \cap \overline{B}(p_i,r)\}_{i\in\{1,\dots,n\}}$, which is a partition of $\cup_i \overline{B}(p_i,r) \cap S$. For $i,j \in \{1,\dots,n\}$, let

$$\Delta_{ij}(\mathcal{P},r) \triangleq \big(V_i(\mathcal{P}) \cap \overline{B}(p_i,r)\big) \cap \big(V_j(\mathcal{P}) \cap \overline{B}(p_j,r)\big).$$

Fig. 1 shows an example of these geometric constructions.

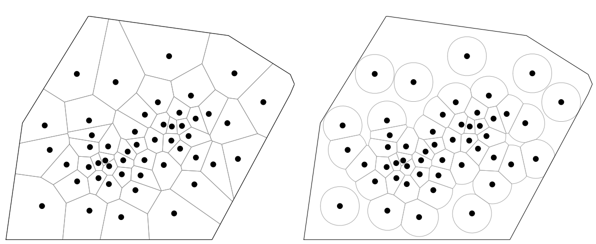

Fig. 1. Voronoi partition of a convex polygon Q generated by 50 points selected randomly (left) and Voronoi partition of Q generated by the same configuration restricted to $\cup_i \overline{B}(p_i,r) \cap Q$, with $r = .2$ (right).

B. Proximity graphs and their properties

For standard notions in graph theory we refer to [19, Chapter 1]. Here, we start by briefly reviewing the notion of Laplacian matrix. Let $G = (V,E)$ be an undirected graph with n vertices. The graph Laplacian matrix associated with G is defined as $L = \Delta - A$, where Δ is the degree matrix and A is the adjacency matrix. The graph Laplacian is symmetric, positive semi-definite and has an eigenvector at $\lambda = 0$ with eigenvector $(1,\dots,1)^T$. Furthermore, the graph G is connected if and only if $\operatorname{rank}(L) = n-1$.

Let us introduce some concepts about proximity graphs for point sets in $\mathbb{R}^d$. For a set S, let $\mathbb{G}(S)$ be the set of undirected graphs whose vertex set is an element of $\mathbb{F}(S)$. A *proximity graph* $\mathcal{G} : \mathbb{F}(\mathbb{R}^d) \to \mathbb{G}(\mathbb{R}^d)$ associates to $\mathcal{P} \in \mathbb{F}(\mathbb{R}^d)$, an undirected graph with vertex set $\mathcal{P}$ and edge set $\mathcal{E}_{\mathcal{G}}(\mathcal{P})$, where $\mathcal{E}_{\mathcal{G}} : \mathbb{F}(\mathbb{R}^d) \to \mathbb{F}(\mathbb{R}^d \times \mathbb{R}^d)$ satisfies $\mathcal{E}_{\mathcal{G}}(\mathcal{P}) \subseteq \{(p,q) \in \mathcal{P} \times \mathcal{P} \mid p \neq q\}$. In other words, the edge set of a proximity graph depends on the location of its vertices. The following examples are defined in [17], [20], [21], [7]:

(i) the *complete* graph $\mathcal{G}_{\text{complete}}$, with $\mathcal{E}_{\mathcal{G}_{\text{complete}}}(\mathcal{P}) = \{(p,q) \in \mathcal{P} \times \mathcal{P} \mid p \neq q\}$;
(ii) the *r-disk* graph $\mathcal{G}_{\text{disk}}(r)$, for $r \in \mathbb{R}_+$, with $(p_i,p_j) \in \mathcal{E}_{\mathcal{G}_{\text{disk}}(r)}(\mathcal{P})$ if $\|p_i - p_j\| \leq r$;
(iii) the *Delaunay* graph $\mathcal{G}_{\text{D}}$, with $(p_i,p_j) \in \mathcal{E}_{\mathcal{G}_{\text{D}}}(\mathcal{P})$ if $V_i(\mathcal{P}) \cap V_j(\mathcal{P}) \neq \emptyset$;
(iv) the *r-limited Delaunay* graph $\mathcal{G}_{\text{LD}}(r)$ with $(p_i,p_j) \in \mathcal{E}_{\mathcal{G}_{\text{LD}}}(\mathcal{P})$ if $\Delta_{ij}(\mathcal{P}, \frac{r}{2}) \neq \emptyset$;
(v) the *Relative Neighborhood* graph $\mathcal{G}_{\text{RN}}$, with $(p_i,p_j) \in \mathcal{E}_{\mathcal{G}_{\text{RN}}}(\mathcal{P})$ if, for all $p_k \in \mathcal{P} \setminus \{p_i,p_j\}$, $p_k \notin B(p_i, \|p_i - p_j\|) \cap B(p_j, \|p_i - p_j\|)$;
(vi) the *Gabriel* graph $\mathcal{G}_{\text{G}}$, with $(p_i,p_j) \in \mathcal{E}_{\mathcal{G}_{\text{G}}}(\mathcal{P})$ if, for all $p_k \in \mathcal{P} \setminus \{p_i,p_j\}$,

$$p_k \notin B\Big(\frac{p_i+p_j}{2}, \frac{\|p_i-p_j\|}{2}\Big);$$

(vii) the *Euclidean Minimum Spanning Tree* $\mathcal{G}_{\text{EMST}}$, which for each $\mathcal{P}$, is a minimum-weight spanning tree of $\mathcal{G}_{\text{complete}}$ whose edge (p_i,p_j) has weight $\|p_i - p_j\|$;
(viii) given a simple polytope in $\mathbb{R}^d$, the *visibility* graph $\mathcal{G}_{\text{vis,Q}} : \mathbb{F}(Q) \to \mathbb{G}(Q)$ is defined by $(p_i,p_j) \in \mathcal{E}_{\mathcal{G}_{\text{vis,Q}}}(\mathcal{P})$ if the closed segment from p_i to p_j, denoted $[p_i,p_j]$, is contained in Q.

If needed, we will write $\mathcal{G}_{\text{disk}}(\mathcal{P},r)$ to denote $\mathcal{G}_{\text{disk}}(r)$ at $\mathcal{P}$. We will also work with the proximity graphs $\mathcal{G}_{\text{RN}\,\cap\text{disk}}(r)$, $\mathcal{G}_{\text{G}\,\cap\text{disk}}(r)$ and $\mathcal{G}_{\text{D}\,\cap\text{disk}}(r)$ defined by the intersection of $\mathcal{G}_{\text{RN}}$, $\mathcal{G}_{\text{G}}$ and $\mathcal{G}_{\text{D}}$ with $\mathcal{G}_{\text{disk}}(r)$, $r \in \mathbb{R}_+$, respectively.

To each proximity graph $\mathcal{G}$, we associate the *set of neighbors map* $\mathcal{N}_{\mathcal{G}} : \mathbb{R}^d \times \mathbb{F}(\mathbb{R}^d) \to \mathbb{F}(\mathbb{R}^d)$ defined by

$$\mathcal{N}_{\mathcal{G}}(p,\mathcal{P}) = \{q \in \mathcal{P} \mid (p,q) \in \mathcal{E}_{\mathcal{G}}(\mathcal{P} \cup \{p\})\}.$$

Given $p \in \mathbb{R}^d$, define $\mathcal{N}_{\mathcal{G},p} : \mathbb{F}(\mathbb{R}^d) \to \mathbb{F}(\mathbb{R}^d)$ by $\mathcal{N}_{\mathcal{G},p}(\mathcal{P}) = \mathcal{N}_{\mathcal{G}}(p,\mathcal{P})$. Given $\mathcal{G}_1$ and $\mathcal{G}_2$, we say that $\mathcal{G}_1$ is *spatially distributed over* $\mathcal{G}_2$ if, for all $p \in \mathcal{P}$,

$$\mathcal{N}_{\mathcal{G}_1,p}(\mathcal{P}) = \mathcal{N}_{\mathcal{G}_1,p}\big(\mathcal{N}_{\mathcal{G}_2,p}(\mathcal{P})\big).$$

It is clear that if $\mathcal{G}_1$ is spatially distributed over $\mathcal{G}_2$, then $\mathcal{G}_1(\mathcal{P}) \subset \mathcal{G}_2(\mathcal{P})$ for all $\mathcal{P} \in \mathbb{F}(\mathbb{R}^d)$. The converse is in general not true (e.g., $\mathcal{G}_{\text{D}\,\cap\text{disk}}$ is a subgraph of $\mathcal{G}_{\text{disk}}$, but it is not spatially distributed over it, see [7]).

Theorem 2.1: For $r \in \mathbb{R}_+$, we have

(i) $\mathcal{G}_{\text{EMST}} \subset \mathcal{G}_{\text{RN}} \subset \mathcal{G}_{\text{G}} \subset \mathcal{G}_{\text{D}}$ and $\mathcal{G}_{\text{G}\cap\text{disk}}(r) \subset \mathcal{G}_{\text{LD}}(r) \subset \mathcal{G}_{\text{D}\cap\text{disk}}(r)$;
(ii) $\mathcal{G}_{\text{disk}}(r)$ is connected iff $\mathcal{G}_{\text{EMST}} \subset \mathcal{G}_{\text{disk}}(r)$;
(iii) $\mathcal{G}_{\text{RN}\,\cap\text{disk}}(r)$, $\mathcal{G}_{\text{G}\,\cap\text{disk}}(r)$, and $\mathcal{G}_{\text{LD}}(r)$ are spatially distributed over $\mathcal{G}_{\text{disk}}(r)$;

Fig. 2 shows some examples of proximity graphs in $\mathbb{R}^2$.

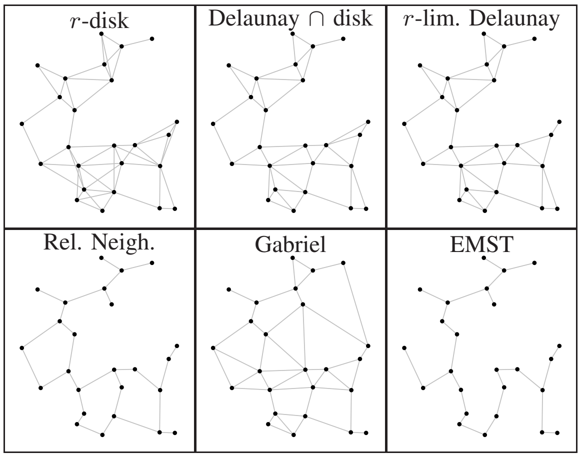

Fig. 2. From top left, in counterclockwise order, r-disk, intersection of Delaunay and r-disk, r-limited Delaunay, Euclidean Minimum Spanning Tree, Gabriel and Relative Neighborhood graphs in $\mathbb{R}^2$ for 25 agents with coordinates uniformly randomly generated within $[-7, 7] \times [-7, 7]$ and $r = 4$.

C. Spatially distributed maps

Here we provide a formally accurate notion of spatially distributed maps. Let $i_{\mathbb{F}} : (\mathbb{R}^d)^n \to \mathbb{F}(\mathbb{R}^d)$ be the natural immersion, i.e., $i_{\mathbb{F}}(P)$ is the point set that contains only the distinct points in $P \in (\mathbb{R}^d)^n$. Note that $i_{\mathbb{F}}$ is invariant under permutations of its arguments.

Given a set Y and a proximity graph $\mathcal{G}$, a map $T : (\mathbb{R}^d)^n \to Y^n$ is *spatially distributed over* $\mathcal{G}$ if there exist a map $\tilde{T} : \mathbb{R}^d \times \mathbb{F}(\mathbb{R}^d) \to Y$, with the property that, for all $(p_1, \dots, p_n) \in (\mathbb{R}^d)^n$ and for all $j \in \{1, \dots, n\}$,

$$T_j(p_1, \dots, p_n) = \tilde{T}(p_j, \mathcal{N}_{\mathcal{G},p_j}(i_{\mathbb{F}}(p_1, \dots, p_n)))\,,$$

where T_j denotes the jth component of T. In other words, the jth component of a spatially distributed map at $(p_1, \dots, p_n)$ can be computed with only the knowledge of the vertex p_j and the neighboring vertices in $\mathcal{G}(i_{\mathbb{F}}(p_1, \dots, p_n))$.

III. ENCODING SENSING TASKS INTO AGGREGATE COST FUNCTIONS

Here we define various locational optimization functions that encode network sensing objectives and characterize their smoothness properties. Cost functions of interest typically quantify an aggregate behavior of the entire network.

A. Aggregate cost functions for deployment

Loosely speaking, deployment consists of the following problem: a network of mobile agents must deploy within an environment of interest in order to achieve maximum coverage of it. Let us formalize this idea. We start by introducing some necessary notation. For $\varepsilon \in \mathbb{R}_+$, let $n_{B(p,\varepsilon)}(q)$ denote the unit outward normal to $B(p, \varepsilon)$ at $q \in \partial B(p, \varepsilon)$. Let $Q \subset \mathbb{R}^d$ be a simple convex polytope. Given $S \subset Q$, let 1_S denote the indicator function, $1_S(q) = 1$ if $q \in S$, and $1_S(q) = 0$ if $q \notin S$. In the remainder of the paper, $\{V_i(\mathcal{P})\}_{i\in\{1,\dots,n\}}$ refers to the Voronoi partition of the polytope Q generated by $\mathcal{P} \in \mathbb{F}(\mathbb{R}^d)$.

A *density function* $\phi : Q \to \overline{\mathbb{R}}_+$ is a bounded function. Given $S \subset Q$, let $\text{area}_\phi(S) = \int_S \phi(q)dq$. A *performance function* $f : \overline{\mathbb{R}}_+ \to \mathbb{R}$ is a non-increasing and piecewise differentiable function with finite jump discontinuities. Given a density function ϕ and a performance function f, consider the function $\mathcal{H} : Q^n \to \mathbb{R}$ defined by

$$\mathcal{H}(P) = \int_Q \max_{i\in\{1,\dots,n\}} f(\|q - p_i\|)\phi(q)dq. \tag{1}$$

Note that $\mathcal{H}$ is an aggregate cost function since it depends on all the locations $p_1, \dots, p_n$. Roughly speaking, $\mathcal{H}$ provides the *expected value of the sensing performance* provided by the group of agents over any point in Q, where ϕ represents a probability that some event take place over Q, and f describes the sensing performance of the sensors. Because of noise and loss of resolution, the sensing performance at point q taken from the ith sensor at the position p_i degrades with the distance $\|q - p_i\|$. Therefore, it will be of interest to find local maxima for $\mathcal{H}$.

Different choices of performance function give rise to different aggregate cost functions with particular features.

Distortion problem: If $f(x) = -x^2$ (differentiable with no jump discontinuities), $\mathcal{H}$ takes the form

$$\mathcal{H}_{\text{C}}(P) = -\sum_{i=1}^n \int_{V_i(P)} \|q - p_i\|^2 \phi(q)dq \triangleq -\sum_{i=1}^n J_{V_i,p_i},$$

where $J_{W,p}$ denotes the polar moment of inertia of the set $W \subset Q$ about the point p. In signal compression, see [22], $-\mathcal{H}_{\text{C}}$ is referred to as the distortion function, and is relevant in many disciplines including facility location, numerical integration, and clustering analysis.

Area problem: If f is the indicator function of the set $[0, R]$, then $\mathcal{H}$ corresponds to the area, measured according to ϕ, covered by the union of the n balls $B(p_1, R), \dots, B(p_n, R)$; that is,

$$\mathcal{H}_{\text{area}}(P) = \text{area}_\phi(\cup_{i=1}^n B(p_i, R))\,.$$

Mixed distortion-area problem: If f is given by $x \mapsto -x^2\, 1_{[0,R)}(x) + b \cdot 1_{[R,+\infty)}(x)$, for $b \leq -R^2$, then $\mathcal{H}$ takes the form

$$\begin{aligned}\mathcal{H}_R(P) = &-\sum_{i=1}^n J_{V_i\cap B(p_i,R),p_i}\\ &+ b\ \text{area}_\phi(Q \setminus \cup_{i=1}^n B(p_i, R))\,.\end{aligned}$$

B. Aggregate cost function for visibility-based deployment

Let Q be a simple non-convex polytope in $\mathbb{R}^d$. Given $p \in Q$, let $S(p) = \{q \in Q \mid [q, p] \subset Q\}$ denote the *visible* *region in* Q from the location p (recall that $[q,p]$ is the closed segment from q to p). Define

$$\mathcal{H}_{\text{vis}}(P) = \int_Q \max_{i\in\{1,\dots,n\}} 1_{S(p_i)}(q)dq.$$

Roughly speaking, the function $\mathcal{H}$ measures the amount of area of the non-convex polygon Q which is visible from any of the agents located at $p_1,\dots,p_n$. Therefore, it will be of interest to find local maxima of $\mathcal{H}_{\text{vis}}$.

C. Aggregate cost functions for consensus

Let us here briefly consider a setup based on a fixed graph instead of a proximity graph. Let $G = (\{1,\dots,n\}, E)$ be an undirected graph with n vertices. Following [13], define the *disagreement function* or *Laplacian potential* $\Phi_G : \mathbb{R}^n \to \overline{\mathbb{R}}_+$ associated with G as

$$\Phi_G(x) = x^T L x = \frac{1}{2}\sum_{(i,j)\in E}(x_j - x_i)^2 .$$

For $i \in \{1,\dots,n\}$, the variable x_i is associated with the agent p_i. The variable x_i might represent physical quantities including attitude, position, temperature, or voltage. Two agents p_i and p_j are said to *agree* if and only if $x_i = x_j$. It is clear that $\Phi_G(x) = 0$ if and only if all neighboring nodes in the graph G agree. If, in addition, the graph G is connected, then all nodes in the graph agree and a consensus is reached. Therefore, $\Phi_G(x)$ is a meaningful function that quantifies the group disagreement in a network.

Note that achieving consensus is a network coordination problem that does not necessarily refer to physical variables such as spatial coordinates or velocities. In what follows we consider two "spatial versions" of consensus, that we refer to as rendezvous and cohesiveness.

D. Aggregate cost function for rendezvous

Roughly speaking, rendezvous means agreement over the location of the agents in a network. With a slight abuse of notation, we introduce the convex hull function $\text{co} : (\mathbb{R}^d)^n \to 2^{(\mathbb{R}^d)}$ as $\text{co}(P) = \text{co}(i_{\mathbb{F}}(P))$, where we represent a polytope in $\mathbb{R}^d$ by its vertex set. The *diameter* function $\text{diam} : 2^{(\mathbb{R}^d)} \to \overline{\mathbb{R}}_+ \cup \{+\infty\}$ is defined by

$$\text{diam}(S) = \sup\{\|p-q\| \mid p,q \in S\}.$$

Consider now the function $V_{\text{diam}} = \text{diam} \circ \text{co} : (\mathbb{R}^d)^n \to \overline{\mathbb{R}}_+$, defined by

$$\begin{aligned} V_{\text{diam}}(P) &= \text{diam}(\text{co}(P)) \\ &= \max\{\|p_i - p_j\| \mid i,j \in \{1,\dots,n\}\}. \end{aligned}$$

Let $\text{diag}((\mathbb{R}^d)^n) = \{(p,\dots,p) \in (\mathbb{R}^d)^n \mid p \in \mathbb{R}^d\}$. One can show that $V_{\text{diam}} = \text{diam}\circ\text{co} : (\mathbb{R}^d)^n \to \overline{\mathbb{R}}_+$ is locally Lipschitz and invariant under permutations of its arguments, and, moreover, $V_{\text{diam}}(P) = 0$ if and only if $P \in \text{diag}((\mathbb{R}^d)^n)$. Therefore, the set of global minima of V_{diam} corresponds to the network configurations where the agents rendezvous.

E. Aggregate cost functions for cohesiveness

Let $h : \mathbb{R}_+ \to \mathbb{R}$ be a continuously differentiable function satisfying the following conditions: (i) $\lim_{R\to 0} h(x) = +\infty$, (ii) there exists $R_0 \in \mathbb{R}_+$ such that h is convex on $(0, R_0)$ achieving its minimum at all the points in the interval $[R_*, R'_*] \subset (0, R_0)$ and h is concave on $(R_0, +\infty)$, and (iii) there exists $R_1 \in \mathbb{R}_+$, $R_1 \geq R_0$ such that $h(R) = c$ for all $R \geq R_1$. Let $\mathcal{G}$ be a some proximity graph. Define now the aggregate cost function

$$\mathcal{H}_{\text{cohe},\mathcal{G}}(P) = \sum_{(p_i,p_j)\in\mathcal{E}_{\mathcal{G}}(P)} h(\|p_i - p_j\|) .$$

The minima of $\mathcal{H}_{\text{cohe},\mathcal{G}}$ correspond to "cohesive" network configurations. Specifically, for $n \in \{2,3\}$, configurations of minimum for $\mathcal{H}_{\text{cohe},\mathcal{G}}$ have all neighboring agents' locations within a distance contained in the interval $[R_*, R'_*]$. This objective function, or variations of it, has been employed over different proximity graphs in a number of works in the literature ([5] and [6] over the complete graph, [23] over the r-disk graph) to guarantee collision avoidance and cohesiveness of the network.

IV. LaSalle Invariance Principle for non-deterministic discrete-time systems

In recent years, various techniques have been proved useful in the analysis of coordination problems and in establishing correctness guarantees of motion coordination algorithms. Correctness should be loosely understood as the property that consists of making certain sets that encode the desired behaviors of the network invariant and attractive for the evolution of the group of agents under a given algorithm. Among the proposed methods, we distinguish between linear techniques (ergodic and circulant matrices from matrix analysis, graph Laplacians from algebraic graph theory) and nonlinear techniques (symmetries of differential equations, LaSalle Invariance Principles for differential inclusions and for discrete-time non-deterministic dynamical systems).

In the interest of brevity, here we just present one of these tools, a recently-developed version of the LaSalle Invariance Principle. We first review some concepts on the stability of discrete-time dynamical systems and set-valued maps following [24], [7]. For $d \in \mathbb{N}$, an *algorithm on* $\mathbb{R}^d$ is a set-valued map $T : \mathbb{R}^d \to 2^{(\mathbb{R}^d)}$ such that $T(p) \neq \emptyset$ for all $p \in \mathbb{R}^d$. Note that a map from $\mathbb{R}^d$ to $\mathbb{R}^d$ can be interpreted as a singleton-valued map. A *trajectory* of an algorithm T is a sequence $\{p_m\}_{m\in\mathbb{N}\cup\{0\}} \subset \mathbb{R}^d$ with the property that

$$p_{m+1} \in T(p_m), \quad m \in \mathbb{N}\cup\{0\} .$$

In other words, given any initial $p_0 \in \mathbb{R}^d$, a trajectory of T is computed by recursively setting p_{m+1} equal to an arbitrary element in $T(p_m)$. An algorithm is therefore a non-deterministic discrete-time dynamical system.

An algorithm T is *closed at* $p \in \mathbb{R}^d$ if for all pairs of convergent sequences $p_k \to p$ and $p'_k \to p'$ such that $p'_k \in T(p_k)$, one has $p' \in T(p)$. An algorithm is *closed on* $W \subset \mathbb{R}^d$ if it is closed at p, for all $p \in W$. In particular, every continuous map $T : \mathbb{R}^d \to \mathbb{R}^d$ is closed on $\mathbb{R}^d$. A set

C is *weakly positively invariant with respect to* T if, for any $p_0 \in C$, there exists $p \in T(p_0)$ such that $p \in C$. The function $V : \mathbb{R}^d \to \mathbb{R}$ is *non-increasing along* T *on* $W \subset \mathbb{R}^d$ if $V(p') \leq V(p)$ for all $p \in W$ and $p' \in T(p)$. We are ready to state the following result.

Theorem 4.1: (LaSalle Invariance Principle for closed algorithms [7]) Let T be a closed algorithm on $W \subset \mathbb{R}^d$ and let $V : \mathbb{R}^d \to \mathbb{R}$ be a continuous function non-increasing along T on W. Assume the trajectory $\{p_m\}_{m \in \mathbb{N} \cup \{0\}}$ of T takes values in W and is bounded. Then there exists $c \in \mathbb{R}$ such that

$$p_m \longrightarrow M \cap V^{-1}(c)\,,$$

where M is the largest weakly positively invariant set contained in

$$\{p \in \overline{W} \mid \exists p' \in T(p) \text{ such that } V(p') = V(p)\}.$$

V. Toward a systematic methodology for the design of motion coordination algorithms

In this section, we elaborate on the role played by the tools introduced in the previous sections. Throughout the discussion we do not enter into technical details, but rather refer to various works for further reference. Our intention is to provide a first step toward the establishment of a rigorous systems theoretic approach to the design and analysis of coordination algorithms for a variety of sensing tasks.

We start by informally describing the notion of coordination algorithm. Roughly speaking, a coordination algorithm consists of a control law for each agent of the network. In particular, we mainly focus on coordination algorithms which specify the same control law for all agents. Mathematically, a coordination algorithm will be described in different forms, depending on whether it is implemented in continuous time (a vector field, or more generally, a differential inclusion over the configuration space of the network) or in discrete time (a map, or more generally, a set-valued map over the configuration space of the network).

Given a network of identical agents equipped with motion control, communication and sensing capabilities, the following subsections contain various approaches to the study of distributed motion coordination. Loosely speaking, a first approach is based on the *design* of *gradient flows*: here we typically are given a sensing task to be performed by the network and a proximity graph as communication constraint. A second approach is based on the *analysis* of *emerging behaviors*: in this case a notion of neighboring agents and an interaction law between them is usually given. The remaining two approaches build on these two.

A. Designing the coordination algorithm from the aggregate cost function

The first step of this approach is to identify the aggregate cost function which is relevant for the desired sensing task. Once this objective function is determined, one analyzes its differentiable properties and computes its (generalized) gradient. With this information, it is possible to characterize its critical points, i.e., the desired network configurations. The next step is to identify those proximity graphs with respect to which the gradient of the objective function is spatially distributed (cf. Section II-C). If any of these proximity graphs is computable with the capabilities of the mobile network, then a control law for each agent simply consists of following the gradient of the aggregate cost function. By LaSalle Invariance Principle, such a coordination algorithm is automatically guaranteed to ensure convergence of the closed-loop network trajectories to the set of critical points.

Example 5.1: (Distortion and area problems): The coordination algorithms proposed in [7] for the distortion problem and in [7] for the area problem are examples of this approach. For Q a simple convex polygon in $\mathbb{R}^2$, one can prove that the functions $\mathcal{H}_C$ and $\mathcal{H}_{\text{area}}$ are locally Lipschitz on Q^n and differentiable on $Q^n \setminus \{(p_1, \dots, p_n) \in (\mathbb{R}^2)^n \mid p_i = p_j$ for some $i, j \in \{1, \dots, n\}$, $i \neq j\}$, with

$$\frac{\partial \mathcal{H}_{\text{C}}}{\partial p_i}(P) = 2\mathcal{M}_{V_i(P)}(\text{CM}_{V_i(P)} - p_i)\,, \tag{2a}$$

$$\frac{\partial \mathcal{H}_{\text{area}}}{\partial p_i}(P) = \sum_{k=1}^{M_i(R)} \int_{\text{arc}_{i,k}(R)} n_{B(p_i,R)}\, \phi\,, \tag{2b}$$

where $\text{arc}_{i,1}(R), \dots, \text{arc}_{i,M_i(R)}(R)$ correspond to the arcs in $\partial(V_i(i_{\mathbb{F}}(P)) \cap \overline{B}(p_i, R))$, for each $i \in \{1, \dots, n\}$. Here $\mathcal{M}_W$ and CM_W denote, respectively, the mass and the center of mass with respect to ϕ of $W \subset Q$. The critical points $P \in Q^n$ of $\mathcal{H}_{\text{C}}$ satisfy $p_i = \text{CM}_{V_i(P)}$ for all $i \in \{1, \dots, n\}$. Such configurations are called *centroidal Voronoi configurations*, see [22].

From equation (2a) it is clear that the gradient of $\mathcal{H}_{\text{C}}$ is spatially distributed over $\mathcal{G}_{\text{D}}$, whereas from equation (2b) one deduces that the gradient of $\mathcal{H}_{\text{area}}$ is spatially distributed over $\mathcal{G}_{\text{LD}}(2R)$. The gradient flows of $\mathcal{H}_{\text{C}}$ and of $\mathcal{H}_{\text{area}}$ correspond to the coordination algorithms "move-toward-the-centroid of own Voronoi cell" and "move in the direction of the (weighted) normal to the boundary of own cell," respectively. Figs. 3 and 4 show an example of the execution of these algorithms. □

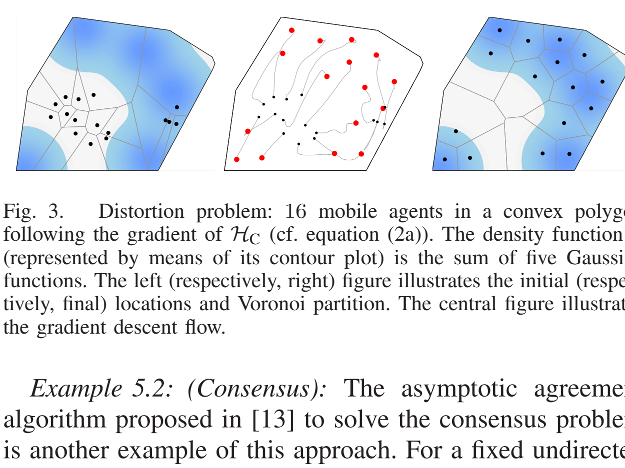

Fig. 3. Distortion problem: 16 mobile agents in a convex polygon following the gradient of $\mathcal{H}_{\text{C}}$ (cf. equation (2a)). The density function ϕ (represented by means of its contour plot) is the sum of five Gaussian functions. The left (respectively, right) figure illustrates the initial (respectively, final) locations and Voronoi partition. The central figure illustrates the gradient descent flow.

Example 5.2: (Consensus): The asymptotic agreement algorithm proposed in [13] to solve the consensus problem is another example of this approach. For a fixed undirected graph $G = (\{1, \dots, n\}, E)$, the function Φ_G is smooth, and its partial derivative takes the form

$$\frac{\partial \Phi_G}{\partial x} = Lx\,. \tag{3}$$

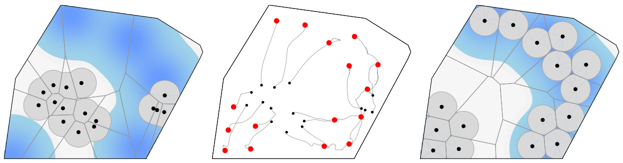

Fig. 4. Area problem: 16 mobile agents in a convex polygon following the gradient of $\mathcal{H}_{\mathrm{area}}$ (cf. equation (2b)). The density function ϕ and the environment are the same as in Fig. 3. Each agent operates with a finite sensing/communication radius equal to $r = .45$. For each agent i, the intersection $V_i \cap B(p_i, \frac{r}{2})$ is plotted in light gray.

Clearly, this gradient is spatially distributed with respect to the graph G itself. The implementation of the gradient control law leads to the algorithm $\dot{x}_i = \sum_{(i,j)\in E}(x_j - x_i)$, $i \in \{1,\dots,n\}$ which asymptotically achieves average-consensus, i.e., the final value upon which all agents agree can be proved to be equal to $\frac{1}{n}\sum_{i=1}^{n} x_i(0)$. It is also possible to consider algorithms that take into account different weights at each of the edges of the graph G. □

Example 5.3: (Cohesiveness): Another example of this approach are the various coordination algorithms proposed in the literature to achieve cohesiveness [5], [6], [23]. For the complete graph $\mathcal{G}_{\mathrm{complete}}$, the function $\mathcal{H}_{\mathrm{cohe},\mathcal{G}_{\mathrm{complete}}}$ is smooth on $Q^n \setminus \{(p_1,\dots,p_n) \in (\mathbb{R}^2)^n \mid p_i = p_j$ for some $i,j \in \{1,\dots,n\}$, $i \neq j\}$, with

$$\frac{\partial \mathcal{H}_{\mathrm{cohe},\mathcal{G}_{\mathrm{complete}}}}{\partial p_i}(P) = \sum_{j\neq i}^{n} \frac{\partial}{\partial p_i}\big(h(\|p_i - p_j\|)\big) = \sum_{p_j \in \mathcal{N}_{\mathcal{G}_{\mathrm{disk}}(R_1),p_i}} \frac{\partial}{\partial p_i}\big(h(\|p_i - p_j\|)\big),$$

where we used the fact that $0 = \partial h/\partial R$ for $R \geq R_1$. According to the notions introduced in Section II, this gradient is spatially distributed over $\mathcal{G}_{\mathrm{disk}}(R_1)$. The gradient descent control law for each agent guarantees that the network agents will asymptotically approach the set of critical points of $\mathcal{H}_{\mathrm{cohe},\mathcal{G}_{\mathrm{complete}}}$. □

Not always does the aggregate cost function enjoy the desirable property that its gradient is spatially distributed with respect to the required proximity graph. In other words, given an available information flow, not always the appropriate gradient flow can be computed. If this is the case, then one possible approach is the following: (i) consider constant-factor approximations of the cost function, (ii) identify those approximations whose gradient is spatially distributed with respect to an appropriate proximity graph, and (iii) implement as coordination algorithm that each agent follows the gradient of the approximation.

Example 5.4: (Mixed distortion-area problem): The coordination algorithm proposed in [7] for the distortion problem falls into the situation described in the previous paragraph. Since the gradient of $\mathcal{H}_{\mathrm{C}}$ is spatially distributed over $\mathcal{G}_{\mathrm{D}}$ (cf. (2a)), and this graph is not spatially distributed over the r-disk graph, the coordination algorithm "move-toward-the-centroid of own Voronoi cell" is not implementable over a network with limited-range interactions. Instead, one can try to compute constant-factor approximations of $\mathcal{H}_{\mathrm{C}}$. Indeed, for $r \in \mathbb{R}_+$, one has that (i) for $\beta = r^2/(2\operatorname{diam} Q)^2$,

$$\mathcal{H}_{\frac{r}{2}}(P) \leq \mathcal{H}_{\mathrm{C}}(P) \leq \beta\,\mathcal{H}_{\frac{r}{2}}(P) < 0\,, \qquad (4)$$

and (ii) the partial derivative of $\mathcal{H}_{\frac{r}{2}}$ with respect to the position of the ith agent is

$$\frac{\partial \mathcal{H}_{\frac{r}{2}}}{\partial p_i}(P) = 2\mathcal{M}_{V_i(P)\cap B(p_i,\frac{r}{2})}(\mathrm{CM}_{V_i(P)\cap B(p_i,\frac{r}{2})} - p_i) - \Big(\big(\frac{r}{2}\big)^2 + b\Big) \sum_{k=1}^{M_i(\frac{r}{2})} \int_{\mathrm{arc}_{i,k}(\frac{r}{2})} n_{B(p_i,\frac{r}{2})}\,\phi\,,$$

where $\mathrm{arc}_{i,1}(\frac{r}{2}),\dots,\mathrm{arc}_{i,M_i(\frac{r}{2})}(\frac{r}{2})$ correspond to the arcs in $\partial(V_i(i_{\mathbb{F}}(P)) \cap \overline{B}(p_i,\frac{r}{2}))$. Clearly, the gradient of $\mathcal{H}_{\frac{r}{2}}$ is spatially distributed over $\mathcal{G}_{\mathrm{LD}}(r)$, and therefore, the coordination algorithm based on the corresponding gradient control law is implementable over a network with limited range interactions. Fig. 5 illustrates the execution of this algorithm. □

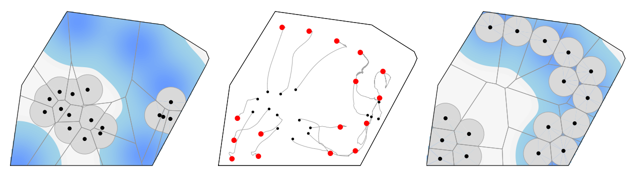

Fig. 5. Mixed distortion-area problem: 16 mobile agents in a convex polygon following the gradient of $\mathcal{H}_{\frac{r}{2}}$. The density function ϕ and the environment are the same as in Fig. 3. Each agent operates with a finite radius $r = .45$. From the constant-factor approximation (4), the absolute error is less than or equal to $(\beta - 1)\mathcal{H}_{\frac{r}{2}}(P_{\mathrm{final}}) \approx 6.77$, where P_{final} denotes the final configuration of this execution. The percentage error in the value of the $\mathcal{H}_{\mathrm{C}}$ at P_{final} with respect to the value at the final configuration of the execution in Fig. 3 is approximately equal to 30%.

B. Analyzing the coordinated behavior emerging from basic interaction laws

This approach consists of devising a simple control law, typically inspired by some sort of heuristic or behavior, that implemented over each agent of the network would reasonably perform the desired sensing task. Once this is done, one should (i) check that the resulting coordination algorithm is spatially distributed with regards to some appropriate proximity graph and (ii) characterize its asymptotic convergence properties. One way of doing the latter is by finding an aggregate cost function that encodes the desired sensing task and by showing that it is optimized along the execution of the coordination algorithm.

Example 5.5: (Move-away-from-closest-neighbor): Consider the coordination algorithm studied in [16] where each network agent moves away from its closest neighbor (see Fig. 6). This simple interaction law is spatially distributed over $\mathcal{G}_{\mathrm{D}}$. One can prove that along the evolution of the network, the aggregate cost function

$$\mathcal{H}_{\mathrm{SP}}(P) = \min_{i\neq j\in\{1,\dots,n\}} \big\{\tfrac{1}{2}\|p_i - p_j\|, \operatorname{dist}(p_i,\partial Q)\big\}\,, \qquad (5)$$

is monotonically non-decreasing. This function corresponds to the *non-interference problem*, where the network tries to maximize the coverage of the domain in such a way that the various sensing radius of the agents do not overlap or leave the environment (because of interference). Under appropriate technical conditions, one can show that the critical points of $\mathcal{H}_{\mathrm{SP}}$ are incenter Voronoi configurations. □

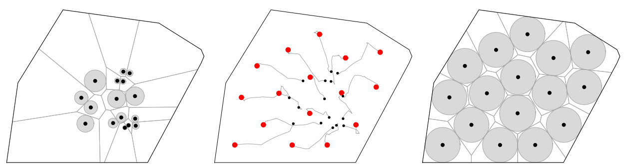

Fig. 6. Non-interference problem: "move-away-from-closest-neighbor" algorithm for 16 mobile agents in a convex polygon. The left (respectively, right) figure illustrates the initial (respectively, final) locations and Voronoi partition. The central figure illustrates the network evolution. For each agent i, the ball of maximum radius contained in V_i and centered at p_i is plotted in light gray.

Example 5.6: (Flocking): Consider the coordination algorithm analyzed in [3] for the flocking problem. Roughly speaking, flocking consists of agreeing over the direction of motion by the agents in the network. Let $\mathcal{G}$ be a proximity graph. Now, consider the coordination algorithm where each agent performs the following steps: (i) detects its neighbors' (according to $\mathcal{G}$) heading; (ii) computes the average of its neighbors' heading and its own heading, and (iii) updates its heading to the computed average. Clearly, this algorithm is spatially distributed over $\mathcal{G}$. Moreover, assuming that $\mathcal{G}$ remains connected throughout the evolution, one can show that the agents asymptotically acquire the same heading. The method of proof for this result builds on the properties of ergodic and non-negative matrices from linear algebra and the properties of graph Laplacians from algebraic graph theory. It is also worth mentioning that, for the r-disk graph $\mathcal{G}_{\text{disk}}$, one can establish [3] that there does not exist in general a quadratic Lyapunov function that helps characterize the asymptotic stability properties of the algorithm. □

C. Designing the coordination algorithm from local objective functions

This approach has common elements with the two approaches presented previously. Now, in order to derive a control law for each specific agent, one assumes that the neighboring agents of that agent, or some spatial structure attributed to it, remain fixed. One then defines a local objective function, which is somehow related with the global aggregate cost function encoding the desired sensing task, and devises a control law to optimize it. The specific control strategy might be heuristically derived or arise naturally from the gradient information of the local objective function. Once the coordination algorithm is setup, one should check that it is spatially distributed, and characterize its asymptotic convergence properties.

Example 5.7: (Non-interference problem): Consider the aggregate cost function $\mathcal{H}_{\mathrm{SP}}$ defined in equation (5). Consider the alternative expression,

$$\mathcal{H}_{\mathrm{SP}}(P) = \min_{i \in \{1,\dots,n\}} \mathrm{sm}_{V_i(P)}(p_i)\,,$$

where $\mathrm{sm}_W(p)$ is the distance from p to the boundary of the convex polygon W, i.e., $\mathrm{sm}_W(p) \triangleq \mathrm{dist}(p, \partial W)$. Now, for $i \in \{1,\dots,n\}$, consider $\mathrm{sm}_{V_i(P)}$ as a local objective function. Assuming that the Voronoi cell $V_i(P)$ remains fixed, then one can implement the (generalized) gradient ascent of $\mathrm{sm}_{V_i(P)}$ as the control law for the agent p_i. One can show [16] that this interaction law precisely corresponds to the strategy "move-away-from-closest-neighbor" discussed in Example 5.5. A related strategy consists of each agent moving toward the incenter of its own Voronoi cell, which can be shown to also make $\mathcal{H}_{\mathrm{SP}}$ monotonically non-decreasing and enjoy analogous asymptotic convergence properties (see Fig. 7). □

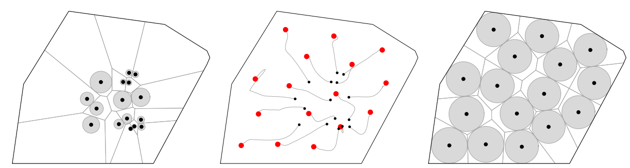

Fig. 7. Non-interference scenario: "move-toward-the-incenter" algorithm for 16 mobile agents in the same convex polygon as in Fig. 6.

Example 5.8: (Rendezvous): Let $\mathcal{G}$ be a proximity graph. Consider the Circumcenter Algorithm over $\mathcal{G}$, where each agent performs the following steps: (i) detects its neighbors according to $\mathcal{G}$; (ii) computes the circumcenter of the point set comprised of its neighbors and of itself, and (iii) moves toward this circumcenter while maintaining connectivity with its neighbors. In order to maintain connectivity, the allowable motion of each agent is conveniently restricted (see [8], [9], [11] for further details).

Note that with step (ii), assuming that all other agents remain fixed, each agent minimizes the local objective function given by the maximum distance from the agent to all its neighbors in the proximity graph $\mathcal{G}$. By construction, this coordination algorithm is spatially distributed over the proximity graph $\mathcal{G}$. Moreover, one can prove that the evolution of the aggregate cost function V_{diam} is monotonically non-increasing along the execution of the Circumcenter Algorithm. Using the LaSalle Invariance Principle for closed algorithms (cf. Theorem 4.1) presented in Section IV, one can indeed characterize the asymptotic correctness properties of the Circumcenter Algorithm over $\mathcal{G}$. See Fig. 8 for an illustration of its execution. A similar algorithm, where the agents of the network, instead of rendezvousing at a common position, rendezvous at the direction of their velocity vectors would lead to a solution of the flocking problem. □

D. Designing the coordination algorithm by composing different behaviors

This approach builds on the other approaches presented above. One approach to the composition of behaviors is to implement one coordination algorithm on most of the

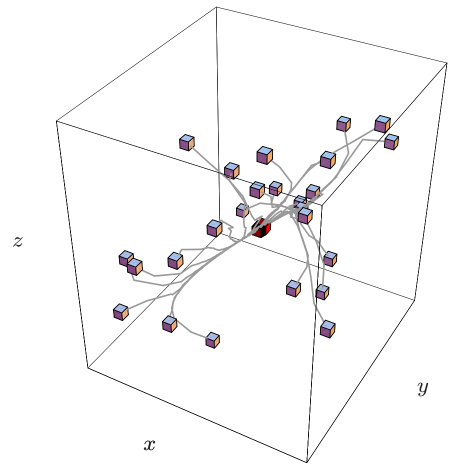

Fig. 8. Evolution of the Circumcenter Algorithm over $\mathcal{G}_{\mathrm{G}}(r) \cap \mathcal{G}_{\mathrm{disk}}(r)$ in $\mathbb{R}^3$.

network agents and a second coordination algorithm on the small subset of other agents. Coupling two algorithms in this parallel fashion might result in interesting overall network behaviors. For example, one may prescribe the open-loop motion of some of the network agents (e.g., specifying that some particular agents must stay fixed, or follow a desired path, no matter what the rest of agents will do) and implement a feedback law for the others. Examples of this approach include (1) the formation control strategy in [9] to make the network form a straight line, and (2) the leader-following algorithm proposed in [3] to make the network flock in a pre-specified direction. It is also possible to explore more general parallel, serial and hierarchical approaches to the composition of behaviors.

VI. Conclusions

We have surveyed a set of recent tools (proximity graphs, spatially distributed maps, aggregate cost functions, LaSalle Invariance Principle for non-deterministic discrete-time systems) that we believe are important in the design and analysis of motion coordination algorithms. We have also identified various approaches to the design of coordination algorithms and shown the wide applicability of the proposed tools in these approaches. We hope that in the coming years the set of control tools for motion coordination will continue to expand and will lead to the design of other spatially distributed primitives and the analysis of the algorithms' performance and complexity.

Acknowledgments

This material is based upon work supported in part by ONR YIP Award N00014-03-1-0512 and NSF SENSORS Award IIS-0330008. Sonia Martínez's work was supported in part by a Fulbright Postdoctoral Fellowship from the Spanish Ministry of Education and Science.

References

[1] I. Suzuki and M. Yamashita, "Distributed anonymous mobile robots: Formation of geometric patterns," *SIAM Journal on Computing*, vol. 28, no. 4, pp. 1347–1363, 1999.

[2] E. W. Justh and P. S. Krishnaprasad, "Equilibria and steering laws for planar formations," *Systems & Control Letters*, vol. 52, no. 1, pp. 25–38, 2004.

[3] A. Jadbabaie, J. Lin, and A. S. Morse, "Coordination of groups of mobile autonomous agents using nearest neighbor rules," *IEEE Transactions on Automatic Control*, vol. 48, no. 6, pp. 988–1001, 2003.

[4] E. Klavins, R. Ghrist, and D. Lipsky, "A grammatical approach to self-organizing robotic systems," *International Journal of Robotics Research*, 2004. Submitted.

[5] V. Gazi and K. M. Passino, "Stability analysis of swarms," *IEEE Transactions on Automatic Control*, vol. 48, no. 4, pp. 692–697, 2003.

[6] P. Ögren, E. Fiorelli, and N. E. Leonard, "Cooperative control of mobile sensor networks: adaptive gradient climbing in a distributed environment," *IEEE Transactions on Automatic Control*, vol. 49, pp. 1292–1302, Aug. 2004.

[7] J. Cortés, S. Martínez, and F. Bullo, "Spatially-distributed coverage optimization and control with limited-range interactions," *ESAIM. Control, Optimisation & Calculus of Variations*, Jan. 2004. Submitted. Electronic version available at `http://www.me.ucsb.edu/bullo`.

[8] H. Ando, Y. Oasa, I. Suzuki, and M. Yamashita, "Distributed memoryless point convergence algorithm for mobile robots with limited visibility," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 5, pp. 818–828, 1999.

[9] J. Lin, A. S. Morse, and B. D. O. Anderson, "The multi-agent rendezvous problem: an extended summary," in *Proceedings of the 2003 Block Island Workshop on Cooperative Control* (N. E. Leonard, S. Morse, and V. Kumar, eds.), Lecture Notes in Control and Information Sciences, New York: Springer Verlag, 2004. To appear.

[10] 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, 2004.

[11] J. Cortés, S. Martínez, and F. Bullo, "Robust rendezvous for mobile autonomous agents via proximity graphs in d dimensions," *IEEE Transactions on Automatic Control*, June 2004. Submitted. Electronic version available at `http://www.me.ucsb.edu/bullo`.

[12] J. Marshall, M. Broucke, and B. Francis, "Formations of vehicles in cyclic pursuit," *IEEE Transactions on Automatic Control*, 2004. To appear.

[13] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Transactions on Automatic Control*, Sept. 2004. To appear.

[14] L. Moreau, "Time-dependent unidirectional communication in multi-agent systems," 2003. Preprint. Electronically available at `http://xxx.arxiv.org/math.OC/0306426`.

[15] W. Ren and R. W. Beard, "Consensus seeking in multi-agent systems using dynamically changing interaction topologies," *IEEE Transactions on Automatic Control*, Dec. 2003. Submitted.

[16] J. Cortés and F. Bullo, "Coordination and geometric optimization via distributed dynamical systems," *SIAM Journal on Control and Optimization*, June 2004. To appear.

[17] M. de Berg, M. van Kreveld, and M. Overmars, *Computational Geometry: Algorithms and Applications*. New York: Springer Verlag, 1997.

[18] A. Okabe, B. Boots, K. Sugihara, and S. N. Chiu, *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*. Wiley Series in Probability and Statistics, New York, NY: John Wiley & Sons, second ed., 2000.

[19] R. Diestel, *Graph Theory*, vol. 173 of *Graduate Texts in Mathematics*. New York: Springer Verlag, second ed., 2000.

[20] J. W. Jaromczyk and G. T. Toussaint, "Relative neighborhood graphs and their relatives," *Proceedings of the IEEE*, vol. 80, no. 9, pp. 1502–1517, 1992.

[21] H. Edelsbrunner and N. R. Shah, "Triangulating topological spaces," *International Journal of Computational Geometry and Applications*, vol. 7, no. 4, pp. 365–378, 1997.

[22] Q. Du, V. Faber, and M. Gunzburger, "Centroidal Voronoi tessellations: applications and algorithms," *SIAM Review*, vol. 41, no. 4, pp. 637–676, 1999.

[23] H. Tanner, A. Jadbabaie, and G. J. Pappas, "Flocking in fixed and switching networks," *Automatica*, July 2003. Submitted.

[24] D. G. Luenberger, *Linear and Nonlinear Programming*. Reading, MA: Addison-Wesley, second ed., 1984.