Motion Coordination with Distributed Information

Sonia Martínez Jorge Cortés Francesco Bullo

Think globally, act locally
René J. Dubos, 1972

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 common point, or jointly move in a synchronized manner. These coordination 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 emergent and self-organized behaviors in biological groups with distributed agent-to-agent interactions. Interesting dynamical systems arise in biological networks at multiple levels of resolution, all the way from interactions between molecules and cells, e.g., see [1], to the behavioral ecology of animal groups, e.g., see [2]. Flocks of birds and school of fish are able to travel in formation and act as one unit (see [3] and Figures 1 and 2); these swarming behaviors allow animals to defend themselves against predators and to protect areas that they claim as their own. Wildebeest and other animals exhibit complex collective behaviors when migrating (see [4, 5] and Figure 3). Certain foraging behaviors include individual animals partitioning their environment in non-overlapping zones (see [6] and Figure 4). Honey bees [7], gorillas [8], and whitefaced capuchins [9] exhibit synchronized group activities such as initiation of motion and change of travel direction. These remarkable dynamic capabilities are achieved apparently without following a group leader; see [2, 3, 5, 6, 7, 8, 9] for specific examples of animal species and [10, 11] for general studies. In other words, these complex coordinated behaviors emerge while each individual has no global knowledge of the network state and can only plan its motion by observing its closest neighbors.

At the same time, an important engineering reason to study motion coordination stems from the recent interest in man-made groups of embedded systems (such as multi-vehicles and sensor networks). Indeed, it is envisioned that groups of autonomous agents with computing, communication and mobility capabilities will soon become economically feasible and

Figure 1. A school of fish. Photograph taken by the authors at the 50th IEEE Conference in Decision and Control at Paradise Island, Bahamas, in December 2004.

Figure 2. Flock of snow geese flying in formation during migration near the Eastern Shore of Virginia National Wildlife Refuge. Photograph taken from U.S. Fish and Wildlife Service, `http://www.fws.gov`**.**

Figure 3. Aerial photograph of a large wildebeest herd during the migration season on the Serengeti National Park, Tanzania. Photograph taken from [4].

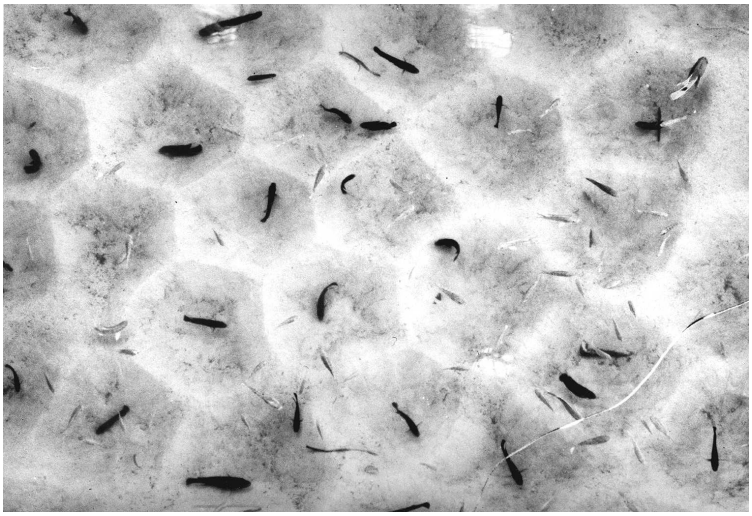

Figure 4. Territorial behavior of fish. Top-view photograph of the territories of the male Tilapia mossambica; each is a pit dug in the sand by its occupant. The boundaries of the territories, the rims of the pits, form a pattern of polygons. The breeding males are the black fish, which range in size from about 15cm to 20cm. The gray fish are the females, juveniles, and nonbreeding males. Photograph and caption taken from [6].

perform a variety of spatially-distributed sensing tasks such as search and rescue, surveillance, environmental monitoring, and exploration.

As a consequence of this growing interest, the research activity on cooperative control has increased tremendously over the last few years. A necessarily incomplete list of works on distributed, or leaderless, motion coordination includes [12, 13, 14] on pattern formation, [15, 16, 17] on flocking, [18] on self-assembly, [19] on swarm aggregation, [20] on gradient climbing, [21, 22, 23, 24] on deployment and task allocation, [25, 26, 27, 28] on rendezvous, [29, 30] on cyclic pursuit, [31] on vehicle routing and [32, 33, 34] on consensus. Heuristic approaches to the design of interaction rules and emergent behaviors have been thoroughly investigated within the literature on behavior-based robotics; see for example [35, 36, 37, 38].

The objective of this paper is to illustrate ways in which systems theory helps us analyze emergent behaviors in animal groups and design autonomous and reliable robotic networks. We present and survey some recently-developed theoretical tools for modeling, analysis and design of motion coordination. We pay special attention to the following issues:

(i) in what sense is a coordination algorithm spatially distributed? To arrive at a satisfactory notion, we will resort to the concept of proximity graph from computational geometry [39]. Proximity graphs of different type 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;

(ii) how can we mathematically express motion coordination tasks? This is an important question if we are interested in providing analytical guarantees for the performance of coordination algorithms. We will discuss various aggregate objective functions from geometric optimization for tasks such as deployment (via a class of multi-center functions that encode area-coverage, detection likelihood, and visibility coverage), rendezvous (via the diameter of convex hull function), cohesiveness, and agreement (via the so-called Laplacian potential from algebraic graph theory). We also discuss their smoothness properties and identify their extreme points via nonsmooth analysis;

(iii) what tools are available to assess the performance of coordination algorithms? We will discuss a combination of system-theoretic and linear algebraic tools that are helpful in establishing stability and convergence of motion coordination algorithms. This includes methods from circulant and Toeplitz tridiagonal matrices and a recently-developed version of the LaSalle Invariance Principle for non-deterministic discrete-time dynamical systems;

(iv) finally, how can we design distributed coordination algorithms? We will build upon the tools introduced earlier and present various approaches. A first approach is based on the design of gradient flows: here we are typically given a coordination task to be performed by the network and a proximity graph as communication constraint. A second approach is based on the analysis of emergent behaviors: in this case a notion

of neighboring agents and an interaction law between them is usually given. A third approach is based on the identification of meaningful local objective functions whose optimization helps the network achieve the desired global task. The last and fourth approach relies on the composition of basic behaviors. We apply these approaches to numerous examples of coordination algorithms proposed in the literature.

Making sense of *distributed*

Our first goal is to provide a formally accurate notion of spatially distributed coordination algorithms. Roughly speaking, one would characterize an algorithm as distributed, as opposed to centralized, if it relies on local information (instead of on global knowledge). One can find precise notions of distributed algorithms for networks with fixed topology in the literature of automata theory and parallel computing [40]. Here, however, we are interested in ad-hoc networks of mobile agents, where the topology changes dynamically, and these definitions are not completely applicable. This motivates our current effort to arrive at a satisfactory definition of spatially distributed algorithms. In doing so, we will borrow the notion of proximity graph from computational geometry. Before getting into this, let us recall some basic geometric notions.

Basic geometric notions

A *partition* of a set S is a collection of subsets of S with disjoint interiors and whose union is S. We denote by $\mathbb{F}(S)$ 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 $\{p_1, \ldots, p_n\}$ in S, the *Voronoi partition* [41, 42] of S generated by $\mathcal{P}$ with respect to the Euclidean norm $\|\cdot\|$ is the collection of sets $\{V_i(\mathcal{P})\}_{i\in\{1,\ldots,n\}}$ defined by $V_i(\mathcal{P}) = \{q \in S \mid \|q - p_i\| \leq \|q - p_j\|, \text{ for all } p_j \in \mathcal{P}\}$. The left plot in Figure 5 illustrates the notion of Voronoi partition.

For $p \in \mathbb{R}^2$ and $r \in \mathbb{R}_+ = (0, +\infty)$, we denote by $\overline{B}(p, r)$ the closed ball in $\mathbb{R}^2$ centered at p of radius r. For $\mathcal{P} \in \mathbb{F}(S)$ with n elements, the collection $\{V_{i,r}(\mathcal{P}) = V_i(\mathcal{P}) \cap \overline{B}(p_i, r)\}_{i\in\{1,\ldots,n\}}$ is a partition of $\cup_i \overline{B}(p_i, r) \cap S$. The right plot in Figure 5 shows an example of this geometric construction. We refer to $V_i(\mathcal{P})$ and $V_{i,r}(\mathcal{P})$ as the *Voronoi cell* and the *r-limited Voronoi cell* of p_i, respectively.

Proximity graphs and their properties

Roughly speaking, a proximity graph is a graph whose vertex set is a set of distinct points on the Euclidean space and whose edge set is a function of the relative locations of the point set. Let us now make this notion more precise. For a set $S \subset \mathbb{R}^d$, 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 $\mathcal{E}_\mathcal{G}(\mathcal{P})$. Here $\mathcal{E}_\mathcal{G} : \mathbb{F}(\mathbb{R}^d) \to \mathbb{F}(\mathbb{R}^d \times \mathbb{R}^d)$ satisfies $\mathcal{E}_\mathcal{G}(\mathcal{P}) \subseteq \{(p, q) \in \mathcal{P} \times \mathcal{P} \mid p \neq q\}$, i.e., a point cannot be its own neighbor. From this definition, one observes that the distinguishing feature of

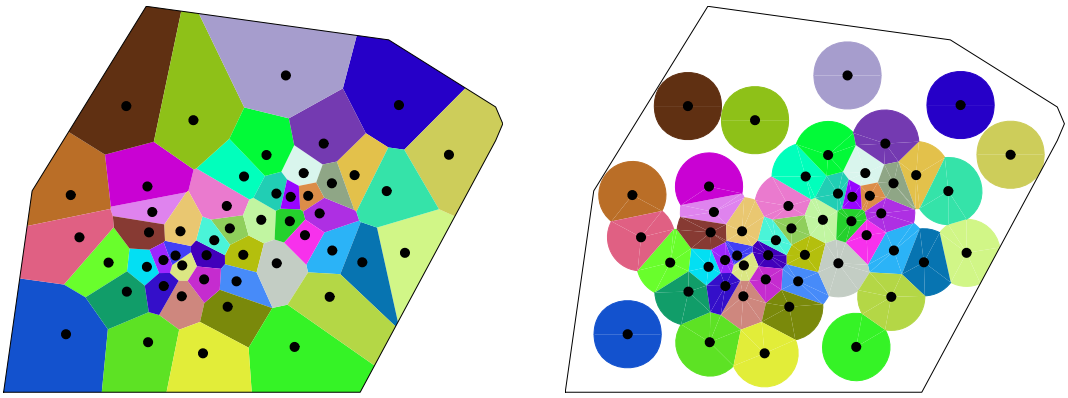

Figure 5. Left, Voronoi partition of the convex polygon Q with vertices $\{(0, 0.1), (2.125, 0.1), (2.975, 1.7), (2.9325, 1.8), (2.295, 2.2), (0.85, 2.4), (0.17, 1.3)\}$ and, right, Voronoi partition of $\cup_i \overline{B}(p_i, r) \cap Q$, with $r = .2$, generated by 50 points randomly selected. The colored regions are Voronoi cells and r-limited Voronoi cells, respectively.

proximity graphs is that their edge sets change with the location of their vertices. A related notion is that of state-dependent graphs, e.g., see [43]. Let us provide some examples of proximity graphs (see [22, 39, 41] for further reference):

(i) the *complete graph* $\mathcal{G}_{\text{complete}}$ has $\mathcal{E}_{\mathcal{G}_{\text{complete}}}(\mathcal{P}) = \{(p, q) \in \mathcal{P} \times \mathcal{P} \mid p \neq q\}$;

(ii) the *r-disk graph* $\mathcal{G}_{\text{disk}}(r)$, for $r \in \mathbb{R}_+$, has $(p_i, p_j) \in \mathcal{E}_{\mathcal{G}_{\text{disk}}(r)}(\mathcal{P})$ if $\|p_i - p_j\| \leq r$;

(iii) the *Delaunay graph* $\mathcal{G}_{\text{D}}$ has $(p_i, p_j) \in \mathcal{E}_{\mathcal{G}_{\text{D}}}(\mathcal{P})$ if $V_i(\mathcal{P}) \cap V_j(\mathcal{P}) \neq \emptyset$;

(iv) the *r-limited Delaunay graph* $\mathcal{G}_{\text{LD}}(r)$, for $r \in \mathbb{R}_+$, has $(p_i, p_j) \in \mathcal{E}_{\mathcal{G}_{\text{LD}}}(\mathcal{P})$ if $V_{i,r}(\mathcal{P}) \cap V_{j,r}(\mathcal{P}) \neq \emptyset$;

(v) for each $\mathcal{P}$, the *Euclidean Minimum Spanning Tree* $\mathcal{G}_{\text{EMST}}$ is a minimum-weight spanning tree of $\mathcal{G}_{\text{complete}}(\mathcal{P})$ whose edge (p_i, p_j) has weight $\|p_i - p_j\|$;

(vi) given a simple polytope Q in $\mathbb{R}^d$, the *visibility graph* $\mathcal{G}_{\text{vis,Q}} : \mathbb{F}(Q) \to \mathbb{G}(Q)$ has $(p_i, p_j) \in \mathcal{E}_{\mathcal{G}_{\text{vis,Q}}}(\mathcal{P})$ if the closed segment from p_i to p_j is contained in Q.

We denote by $\mathcal{G}_{\text{D}\cap\text{disk}}(r)$ the intersection of $\mathcal{G}_{\text{D}}$ with $\mathcal{G}_{\text{disk}}(r)$, $r \in \mathbb{R}_+$. Figure 6 shows some examples of these proximity graphs in $\mathbb{R}^2$. Some important relationships between these graphs are the following:

(i) $\mathcal{G}_{\text{EMST}} \subset \mathcal{G}_{\text{D}}$ and $\mathcal{G}_{\text{LD}}(\frac{r}{2}) \subset \mathcal{G}_{\text{D}\cap\text{disk}}(r)$. Since $\mathcal{G}_{\text{EMST}}$ is connected by definition, this implies that $\mathcal{G}_{\text{D}}$ is connected;

(ii) $\mathcal{G}_{\text{disk}}(r)$ is connected if and only if $\mathcal{G}_{\text{EMST}} \subset \mathcal{G}_{\text{disk}}(r)$;

(iii) $\mathcal{G}_{\text{EMST}\cap\text{disk}}(r)$ and $\mathcal{G}_{\text{LD}}(\frac{r}{2})$ have the same connected components as $\mathcal{G}_{\text{disk}}(r)$.

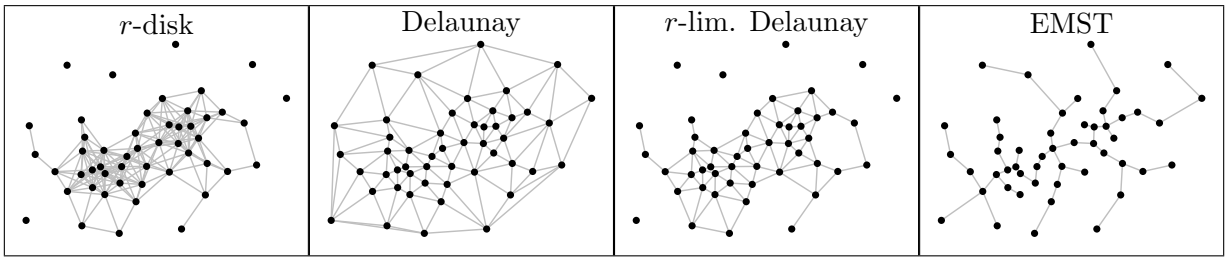

Figure 6. From left to right, $2r$-disk, Delaunay, r-limited Delaunay and Euclidean Minimum Spanning Tree graphs in $\mathbb{R}^2$ for the point set in Figure 5, with $r = .2$.

In the case of graphs with a fixed topology, it is typical to alternatively describe the edge set by means of the sets of neighbors of the individual graph vertices. Likewise, one can associate to each proximity graph $\mathcal{G}$, 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 $p \in \mathbb{R}^d$, we will use the shorthand notation $\mathcal{N}_{\mathcal{G},p}(\mathcal{P}) = \mathcal{N}_{\mathcal{G}}(p, \mathcal{P})$.

It is often the case that, if an agent has the information about the location of its neighbors defined according to certain proximity graph, then it can compute its set of neighbors with respect to a different proximity graph. For instance, if an agent knows the position of its neighbors in the complete graph (i.e., of every other agent in the network), then it is clear that it can compute its set of neighbors with respect to any proximity graph. Let us formalize this idea. Given $\mathcal{G}_1$ and $\mathcal{G}_2$, we say that $\mathcal{G}_1$ is *spatially distributed over* $\mathcal{G}_2$ if, for all $p \in \mathcal{P}$,

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

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. For instance, $\mathcal{G}_{\mathrm{LD}}(\frac{r}{2})$ is spatially distributed over $\mathcal{G}_{\mathrm{disk}}(r)$. On the other hand, $\mathcal{G}_{\mathrm{D} \cap \mathrm{disk}}$ is a subgraph of $\mathcal{G}_{\mathrm{disk}}$, but it is not spatially distributed over it (see [22] for further details). Next, we build up on this discussion to provide a notion of spatially distributed map.

Spatially distributed maps

We are now ready to provide a formally accurate notion of spatially distributed map. To simplify the exposition, we will not distinguish notationally between a tuple in $(p_1, \ldots, p_n) \in (\mathbb{R}^d)^n$ and its associated point set $\{p_1, \ldots, p_n\} \in \mathbb{F}(\mathbb{R}^d)$; we shall denote both quantities by P. The interested reader is referred to [28] for a detailed treatment in this regard.

Consider a map $T : (\mathbb{R}^d)^n \to Y^n$, with Y a given set. Let us denote by T_j the jth component of T. We will say that T is spatially distributed over a proximity graph $\mathcal{G}$ if the jth component T_j evaluated at $(p_1, \ldots, p_n)$ can be computed with only the knowledge of the vertex p_j and the neighboring vertices in $\mathcal{G}(\{p_1, \ldots, p_n\})$. In mathematical terms, this is expressed as follows:

The 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 $(p_1, \ldots, p_n) \in (\mathbb{R}^d)^n$ and all $j \in \{1, \ldots, n\}$, one has $T_j(p_1, \ldots, p_n) = \widetilde{T}(p_j, \mathcal{N}_{\mathcal{G},p_j}(\{p_1, \ldots, p_n\}))$.

According to this definition, a proximity graph $\mathcal{G}_1$ is spatially distributed over a proximity graph $\mathcal{G}_2$ if and only if the set of neighbors map $\mathcal{N}_{\mathcal{G}_1}$ is spatially distributed over $\mathcal{G}_2$. Later, when discussing various coordination algorithms, we will characterize them as being spatially distributed with regards to appropriate proximity graphs.

Encoding coordination tasks

Our second goal is to develop mathematically sound methods to express motion coordination tasks. In the following, we argue that aggregate behaviors of the entire mobile network can be typically quantified by means of appropriately defined objective functions. Using tools from geometric optimization, we will show how to encode various network objectives into locational optimization functions. We will also pay special attention to the smoothness properties of these functions and the spatially distributed character of their gradients.

Aggregate objective functions for deployment

Loosely speaking, the deployment problem consists of placing a network of mobile agents inside an environment of interest in order to achieve maximum coverage of it. Of course, "coverage" can be defined in many possible ways, as we illustrate in the following discussion.

Let $Q \subset \mathbb{R}^d$ be a simple convex polytope. A *density function* $\phi : Q \to \overline{\mathbb{R}}_+$ is a bounded measurable function. One can regard ϕ as a function measuring the probability that some event takes place over the environment. A *performance function* $f : \overline{\mathbb{R}}_+ \to \mathbb{R}$ is a non-increasing and piecewise differentiable function with finite jump discontinuities. This function describes the utility of placing an agent at a certain distance from a location in the environment. To illustrate this notion, consider a sensing scenario in which the agents are equipped with acoustic sensors that take measurements of sounds originating in the environment. Because of noise and loss of resolution, the ability to detect a sound originating at a point q from the ith sensor at the position p_i degrades with the distance $\|q - p_i\|$. This ability is measured by the performance function f.

Given a density function ϕ and a performance function f, we are interested in maximizing the *expected value of the coverage performance* provided by the group of agents over any point in Q. To this end, let us define the function $\mathcal{H} : Q^n \to \mathbb{R}$ by

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

Note that $\mathcal{H}$ is an aggregate objective function since it depends on all the locations $p_1, \ldots, p_n$. It will be of interest to find local maxima for $\mathcal{H}$.

Different choices of performance function give rise to different aggregate objective functions with particular features. We now examine some important cases (let us remind the reader that $\{V_i(P)\}_{i\in\{1,\dots,n\}}$ and $\{V_{i,R}(P)\}_{i\in\{1,\dots,n\}}$ denote the Voronoi partition and the limited-range Voronoi partition of Q generated by $P \in (\mathbb{R}^d)^n$, respectively):

Distortion problem: If $f(x) = -x^2$, then $\mathcal{H}$ takes the form

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

where $J(W, p)$ denotes the polar moment of inertia of the set $W \subset Q$ about the point p. In signal compression $-\mathcal{H}_{\mathrm{C}}$ is referred to as the distortion function and is relevant in many disciplines including facility location, numerical integration, and clustering analysis, see [44].

Area problem: For a set S, let 1_S denote the indicator function, $1_S(q) = 1$, if $q \in S$, and $1_S(q) = 0$, if $q \notin S$. If $f = 1_{[0,R]}$, then $\mathcal{H}$ corresponds to the area, measured according to ϕ, covered by the union of the n balls $\overline{B}(p_1, R), \dots, \overline{B}(p_n, R)$; that is,

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

where $\mathrm{area}_\phi(S) = \int_S \phi(q) dq$.

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

$$\mathcal{H}_R(P) = -\sum_{i=1}^{n} J(V_{i,R}(P), p_i) + b\ \mathrm{area}_\phi(Q \setminus \cup_{i=1}^{n} \overline{B}(p_i, R)).$$

Aggregate objective functions for visibility-based deployment

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

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

Roughly speaking, the function $\mathcal{H}_{\mathrm{vis}}$ 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 is of interest to find local maxima of $\mathcal{H}_{\mathrm{vis}}$. Note that one can also extend the definition of $\mathcal{H}_{\mathrm{vis}}$ using a density function $\phi : Q \to \overline{\mathbb{R}}_+$, so that more importance is given to some regions of the environment being visible to the network (for instance, doors) than others.

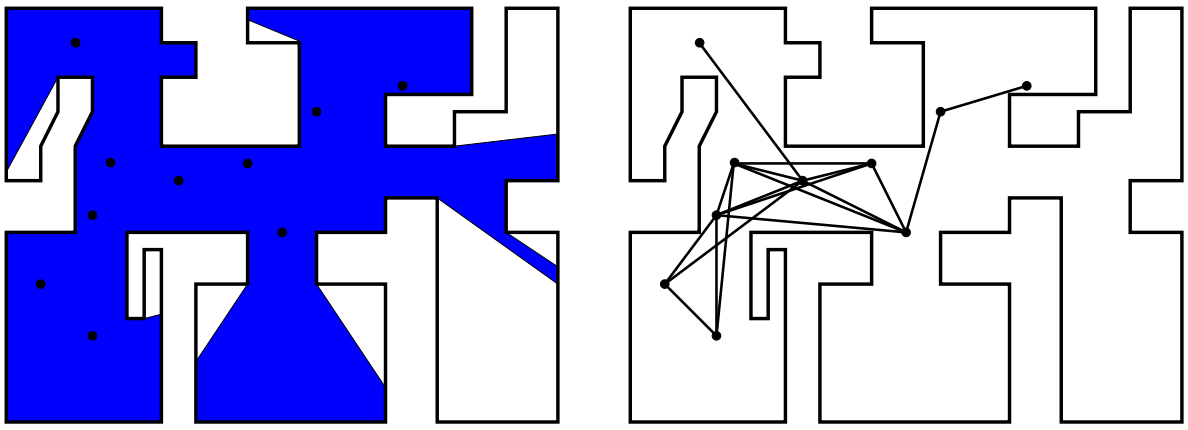

Figure 7. Visible area to the network of a nonconvex environment, or equivalently, value of the function $\mathcal{H}_{\text{vis}}$, at a given configuration (left, blue-colored region) and the underlying visibility graph (right).

Aggregate objective functions for consensus

Let us consider a setup based on a fixed graph instead of a proximity graph. Let $G = (\{1, \ldots, n\}, E)$ be an undirected graph with n vertices. The graph Laplacian matrix L associated with G (see, for instance, [45]) is defined by $L = \Delta - A$, where Δ is the degree matrix and A is the adjacency matrix of the graph. The Laplacian matrix has some useful properties: it is symmetric, positive semi-definite and has $\lambda = 0$ as an eigenvalue with eigenvector $(1, \ldots, 1)^T$. More importantly, the graph G is connected if and only if $\text{rank}(L) = n - 1$, i.e., the eigenvalue 0 has algebraic multiplicity one. Following [32], we define the *disagreement function* or *Laplacian potential* $\Phi_G : \mathbb{R}^n \to \overline{\mathbb{R}}_+$ associated with G by

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

For $i \in \{1, \ldots, n\}$, the variable x_i is associated with the agent p_i. The variable x_i might represent physical quantities including heading, 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.

Aggregate objective function for rendezvous

Roughly speaking, rendezvous means agreement over the location of the agents in a network. Let us define this notion more precisely. Recall that the convex hull $\text{co}(S)$ of a set $S \subset \mathbb{R}^d$

is the smallest convex set containing S. Additionally, the diameter of a set S is defined by

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

With these two ingredients, we can now define the function $V_{\text{diam}} = \operatorname{diam} \circ \operatorname{co} : (\mathbb{R}^d)^n \to \overline{\mathbb{R}}_+$ by

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

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

Aggregate objective functions for cohesiveness

Let us consider one final example of aggregate objective function that encodes a motion coordination task. Let $h : \mathbb{R}_+ \to \mathbb{R}$ be a continuously differentiable function satisfying the following conditions: (i) $\lim_{R \to 0^+} h(R) = +\infty$, (ii) there exists $R_0 \in \mathbb{R}_+$ such that h is convex on $(0, R_0)$ and concave on $(R_0, +\infty)$, (iii) h achieves its minimum at all the points in the interval $[R_*, R'_*] \subset (0, R_0)$, and (iv) there exists $R_1 \geq R_0$ such that $h(R) = c$ for all $R \geq R_1$. Figure 8 gives a particular example of a function h with these features.

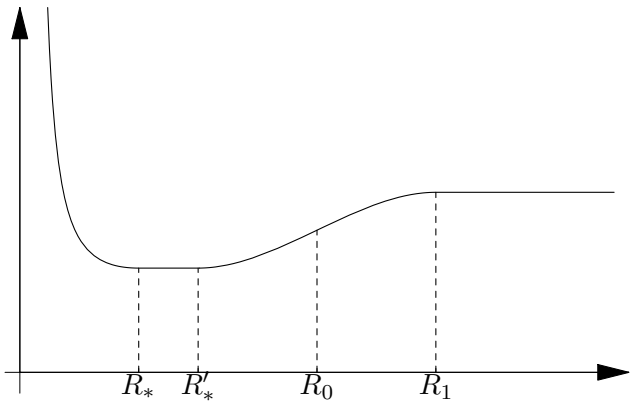

Figure 8. Sample function $h : \mathbb{R}_+ \to \mathbb{R}$ in the definition of the aggregate objective functions for cohesiveness.

Let $\mathcal{G}$ be a proximity graph. Define the aggregate objective 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 that achieve the minimum value of $\mathcal{H}_{\text{cohe},\mathcal{G}}$ have all neighboring

agents' locations within a distance contained in the interval $[R_*, R'_*]$. This objective function and its variations have been employed over different proximity graphs in a number of works in the literature ([19] and [20] over the complete graph, [16] over the r-disk graph) to guarantee collision avoidance and cohesiveness of the mobile network.

Correctness and performance analysis of coordination algorithms

In this section we discuss various techniques that have recently proved useful to analyze cooperative control problems. Let us first describe informally the notion of coordination algorithm. Roughly speaking, such an algorithm consists of a control law for each agent of the network. Mathematically, a coordination algorithm is either a vector field or a map depending on whether the dynamical model is defined on continuous or discrete time.

Given a coordination algorithm, a first scientific concern is the investigation of its correctness. Correctness should be loosely understood as the property that certain sets (encoding the desired behaviors) are invariant and attractive for the evolution of the closed-loop network. A second relevant concern regards the properties of the algorithm execution. It is of interest to have estimates on how quickly a coordination algorithm completes the required task, as well as on how costly the algorithm is in terms of computations, exchanged messages and energy consumption. Further ahead lie other important issues including the analysis of asynchronous executions, the investigation of the algorithm's robustness to communication delays and sensor errors, and the implementation of coordination algorithms under quantization constraints and bandwidth limitations.

Among the proposed analysis methods, we roughly distinguish between linear techniques (ergodic, stochastic [15] and circulant matrices [30] from matrix analysis, graph Laplacians and algebraic connectivity [15, 32] from algebraic graph theory) and nonlinear techniques (symmetries of differential equations [14], LaSalle Invariance Principles for differential inclusions and for non-deterministic dynamical systems [23], graph grammars [18] from automata theory). For reasons of brevity, it is not possible to include here a comprehensive account of all these methods. Instead, the remainder of this section contains a snapshot of two insightful techniques: a class of Toeplitz matrices and the LaSalle Invariance Principle for non-deterministic dynamical systems. The interested reader is invited to explore the references in the bibliography for more in-depth discussions of these and other methods.

Tridiagonal Toeplitz and circulant matrices. Toeplitz and circulant matrices are classic research subjects and we refer to [46, 47] for extensive treatments. It turns out that they can model certain basic coordination algorithms. To prepare this discussion, let us introduce some useful notation. For $n \geq 2$ and $a, b, c \in \mathbb{R}$, we define the $n \times n$ matrices

$\mathrm{Trid}_n(a,b,c)$ and $\mathrm{Circ}_n(a,b,c)$ by

$$\mathrm{Trid}_n(a,b,c) = \begin{bmatrix} b & c & 0 & \dots & 0 \\ a & b & c & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & a & b & c \\ 0 & \dots & 0 & a & b \end{bmatrix}, \quad \mathrm{Circ}_n(a,b,c) = \mathrm{Trid}_n(a,b,c) + \begin{bmatrix} 0 & \dots & \dots & 0 & a \\ 0 & \dots & \dots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & 0 \\ c & 0 & \dots & 0 & 0 \end{bmatrix}.$$

Loosely speaking, we refer to Trid_n and Circ_n as tridiagonal Toeplitz and circulant, respectively. These matrices appear in coordination problems where the communication network has the chain or the ring topology. For instance, the chain and the ring topology play an important role in rendezvous problems [48] and in cyclic pursuit [29, 30] problems, respectively. In Figure 9, we illustrate two algorithms in which the control action of each agent depends on the location of its clockwise and counterclockwise neighbors. For both algorithms, the

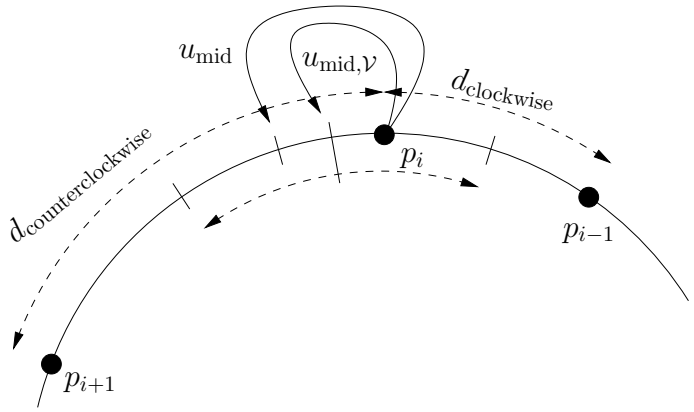

Figure 9. Clockwise and counterclockwise neighbors of an agent in a network evolving in $\mathbb{S}^1$. Control laws such as "go towards the midpoint of the locations of the clockwise and counterclockwise neighbors" (u_{mid}), or "go towards the midpoint of the Voronoi segment of the agent" ($u_{\mathsf{mid},\mathcal{V}}$) give rise to linear dynamical systems described by circulant matrices. In the closed-loop system determined by $u_{\mathsf{mid},\mathcal{V}}$, the agents achieve a uniform distribution along $\mathbb{S}^1$. Oscillations instead persist when the law u_{mid} is adopted.

resulting linear dynamical system is determined by a circulant matrix.

An important feature of tridiagonal Toeplitz and circulant matrices is that their eigenvalues and their dependency on n can be explicitly computed. The importance of this point is illustrated by the following two example results from [48]. First, in certain rendezvous problems, the closed-loop discrete-time trajectory $x : \mathbb{N}_0 = \mathbb{N} \cup \{0\} \to \mathbb{R}^n$ satisfies

$$x(\ell+1) = \mathrm{Trid}_n(a,b,c)\, x(\ell), \qquad x(0) = x_0.$$

For the relevant case where $a = c \neq 0$ and $|b| + 2|a| = 1$, one can show not only that $\lim_{\ell \to +\infty} x(\ell) = 0$, but more importantly, that the maximum time required for $\|x(\ell)\|_2 \leq$

$\varepsilon\|x_0\|_2$ is of order $n^2 \log \varepsilon^{-1}$, for $\varepsilon \in (0,1)$. Second, in deployment problems of the type depicted in Figure 9, the closed-loop discrete-time trajectory $y : \mathbb{N}_0 \to \mathbb{R}^n$ satisfies

$$y(\ell+1) = \mathrm{Circ}_n(a,b,c)\, y(\ell), \qquad y(0) = y_0.$$

For the relevant case where $a \geq 0$, $c \geq 0$, $b > 0$, and $a + b + c = 1$, one can show that $\lim_{\ell \to +\infty} y(\ell) = y_{\mathrm{ave}}\mathbf{1}$, where $y_{\mathrm{ave}} = \frac{1}{n}\mathbf{1}^T y_0$, and that the maximum time required for $\|y(\ell) - y_{\mathrm{ave}}\mathbf{1}\|_2 \leq \varepsilon \|y_0 - y_{\mathrm{ave}}\mathbf{1}\|_2$ is again of order $n^2 \log \varepsilon^{-1}$. (Here $\mathbf{1} = (1, \ldots, 1)^T$.) The key point in these two examples is that stability and time complexity can be both established whenever the tridiagonal Toeplitz and circulant structure is present.

LaSalle Invariance Principle for non-deterministic dynamical systems. From a mathematical viewpoint, once a coordination algorithm for a networked control system is designed, or a particular interaction law modeling a biological behavior is identified, what we have is a set of coupled dynamical systems to analyze. Of course, the couplings between the various dynamical systems change as the topology of the mobile network changes, and this makes things intriguing and complicated at the same time. For instance, one often faces issues such as discontinuity and non-smoothness of the vector fields modeling the evolution of the network. Other times, non-determinism arises because of asynchronism (for example, to simplify the analysis of an algorithm, the asynchronous, deterministic evolution of a mobile network may be subsumed into a larger set of synchronous, non-deterministic evolutions, see e.g., [26]), design choices when devising the coordination algorithm (at each time instant throughout the evolution, each agent may choose among multiple possible control actions, as opposed to a single one, e.g., [22]) or communication, control and sensor errors during the execution of the coordination algorithm (e.g., [25, 28]). Here we briefly survey a recently-developed LaSalle Invariance Principle for non-deterministic discrete-time dynamical systems.

Let $\mathfrak{P}(\mathbb{R}^d)$ denote the collection of subsets of $\mathbb{R}^d$. For $d \in \mathbb{N}$, let $T : \mathbb{R}^d \to \mathfrak{P}(\mathbb{R}^d) \setminus \emptyset$ be a set-valued map (note that a map from $\mathbb{R}^d$ to $\mathbb{R}^d$ can be interpreted as a singleton-valued map). A *trajectory* of T is a sequence $\{p_m\}_{m \in \mathbb{N}_0} \subset \mathbb{R}^d$ with the property that

$$p_{m+1} \in T(p_m)\,, \quad m \in \mathbb{N}_0\,.$$

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 $T(p_m)$. Therefore, T induces a non-deterministic discrete-time dynamical system [22, 49].

In order to study the stability of this type of discrete-time dynamical systems, we need to introduce a couple of new notions. According to [49], 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 $p' \in T(p)$. In particular, every map $T : \mathbb{R}^d \to \mathbb{R}^d$ continuous at $p \in \mathbb{R}^d$ is closed at p. A set C is *weakly positively invariant with respect to* T if, for any initial condition $p_0 \in C$, there exists at least a trajectory of T starting at p_0 that remains in C, or equivalently, if there exists $p \in T(p_0)$ such that $p \in C$. Finally, a 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 from [22], see also [50].

Theorem 1 *(LaSalle Invariance Principle for closed set-valued maps) Let T be a set-valued map closed at p, for all $p \in W \subset \mathbb{R}^d$, and let $V : \mathbb{R}^d \to \mathbb{R}$ be a continuous function non-increasing along T on W. Assume the trajectory $\{p_m\}_{m\in\mathbb{N}_0}$ 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 in $\{p \in \overline{W} \mid \exists p' \in T(p)$ with $V(p') = V(p)\}$.

Designing emergent behaviors

In this section, we elaborate on the role played by the tools introduced in the previous sections for the design and analysis of motion coordination algorithms. We do not enter into technical details throughout the discussion, 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 motion coordination algorithms for a variety of spatially-distributed tasks.

Given a network of identical agents equipped with motion control and communication capabilities, the following subsections contain various approaches to the study of distributed and coordinated motions. Loosely speaking, a first approach is based on the *design* of *gradient flows*: here a coordination task and a proximity graph are typically specified together with a proximity graph imposing a communication constraint. A second approach is based on the *analysis* of *emergent behaviors*: in this case a notion of neighboring agents and an interaction law between them is usually given. A third approach is based on the identification of meaningful local objective functions whose optimization helps the network achieve the desired global task. Finally, the last and fourth approach relies on the composition of basic behaviors.

Designing the coordination algorithm from the aggregate objective function

The first step of this approach consists of identifying a global and aggregate objective function which is relevant to the desired coordination 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 that allow the computation of the gradient of the objective function in a spatially distributed manner. If any of these proximity graphs can be determined with the capabilities of the mobile network, then a control law for each agent simply consists of following the gradient of the aggregate objective function. By the LaSalle Invariance Principle, such a coordination algorithm automatically guarantees convergence of the closed-loop network trajectories to the set of critical points.

Example 1 (Distortion and area problems): The coordination algorithms for the distortion problem and for the area problem proposed in [22] are examples of this approach. Given Q a simple convex polygon in $\mathbb{R}^2$ and $R > 0$, one can prove that the functions $\mathcal{H}_{\mathrm{C}}$ and $\mathcal{H}_{\mathrm{area},R}$ are locally Lipschitz on Q^n and differentiable on $Q^n \setminus \{(p_1, \dots, p_n) \in (\mathbb{R}^2)^n \mid p_i = p_j \text{ for some } i, j \in \{1, \dots, n\}, i \neq j\}$, with

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

$$\frac{\partial \mathcal{H}_{\mathrm{area},R}}{\partial p_i}(P) = \int_{\mathrm{arc}(V_{i,R}(P))} n_{\overline{B}(p_i,R)}\,\phi\,, \tag{2b}$$

where $n_{\overline{B}(p,R)}(q)$ denotes the unit outward normal to $\overline{B}(p,R)$ at $q \in \partial\overline{B}(p,R)$ and, for each $i \in \{1, \dots, n\}$, $\mathrm{arc}(\partial V_{i,R}(P))$ denotes the union of the arcs in $\partial V_{i,R}(P)$. The symbols $\mathrm{M}(W)$ and $\mathrm{CM}(W)$ 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 $\mathcal{H}_{\mathrm{C}}$ satisfy $p_i = \mathrm{CM}(V_i(P))$ for all $i \in \{1, \dots, n\}$. Such configurations are usually referred to as *centroidal Voronoi configurations*, see [44]. The critical points $P \in Q^n$ of $\mathcal{H}_{\mathrm{area},R}$ have the property that each p_i is a local optimum for the area covered by $V_{i,R} = V_i \cap \overline{B}(p_i, R)$ at fixed V_i. We will refer to such configurations as *area-centered Voronoi configurations*.

From equation (2a) it is clear that the gradient of $\mathcal{H}_{\mathrm{C}}$ is spatially distributed over $\mathcal{G}_{\mathrm{D}}$, whereas from equation (2b) one deduces that the gradient of $\mathcal{H}_{\mathrm{area},R}$ is spatially distributed over $\mathcal{G}_{\mathrm{LD}}(R)$. The gradient flows of $\mathcal{H}_{\mathrm{C}}$ and of $\mathcal{H}_{\mathrm{area},R}$ 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. Figures 10 and 12 show an example of the execution of these algorithms. Figures 11 and 13 illustrate the adaptive properties of these algorithms with respect to agent arrivals and departures. □

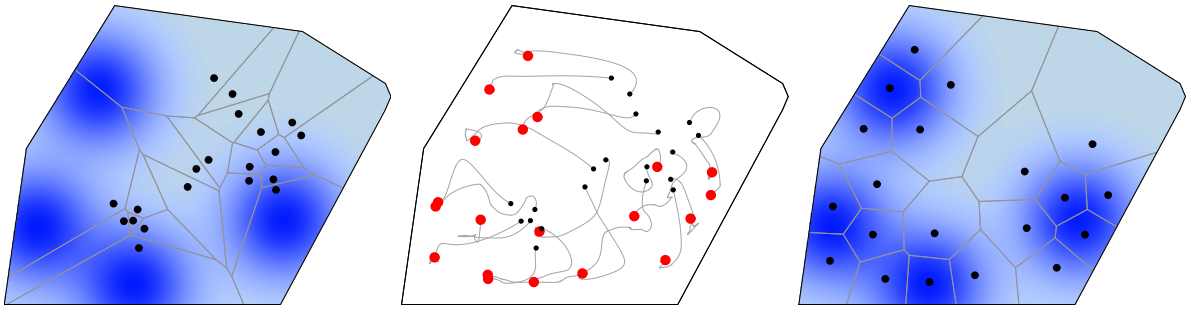

Figure 10. Distortion problem: 20 mobile agents in a convex polygon follow the gradient of $\mathcal{H}_{\mathrm{C}}$ (cf. equation (2a)). The density function ϕ (represented by means of its contour plot) is the sum of four Gaussian functions. The left (respectively, right) figure illustrates the initial (respectively, the final) locations and Voronoi partition. The central figure illustrates the gradient descent flow.

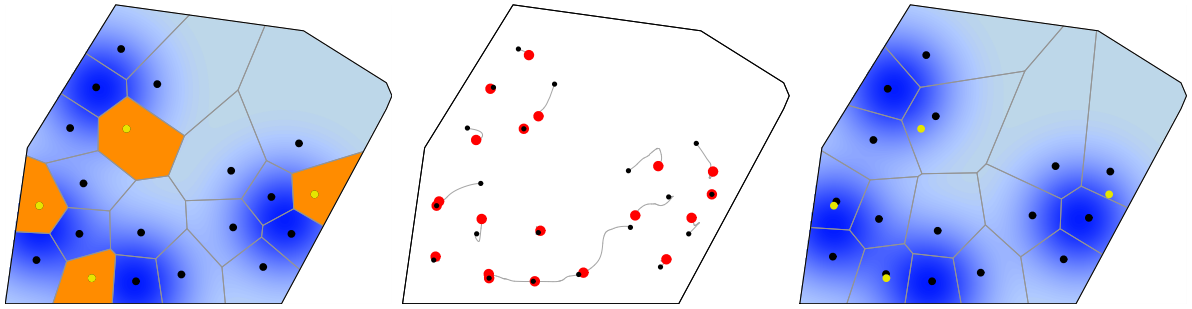

Figure 11. Distortion problem: adaptive behavior of the coordination algorithm. After the final configuration in Figure 10 is reached, four network agents (colored in yellow) fail and cease to provide coverage in their respective Voronoi cells (colored in orange). The rest of the network adapts to the new situation satisfactorily. The left figure illustrates the location of the agents when the failures occur, and the right figure shows the final location of the remaining agents. The central figure illustrates the gradient descent flow since the failure occurred.

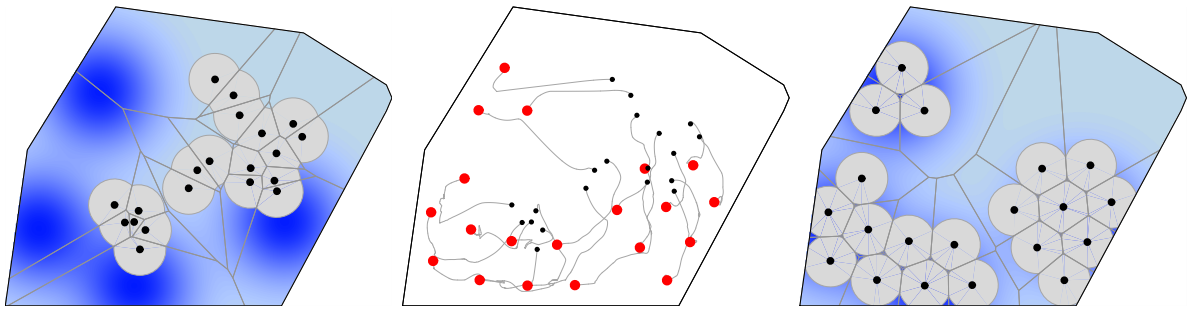

Figure 12. Area problem: 20 mobile agents in a convex polygon follow the gradient of $\mathcal{H}_{\mathsf{area},\frac{r}{2}}$ (cf. equation $(2\mathrm{b})$). The density function ϕ and the environment are the same as in Figure 10. Each agent operates with a finite communication radius equal to $r = .4$. For each agent i, the $\frac{r}{2}$-limited Voronoi cell $V_{i,\frac{r}{2}}(P)$ is plotted in light gray.

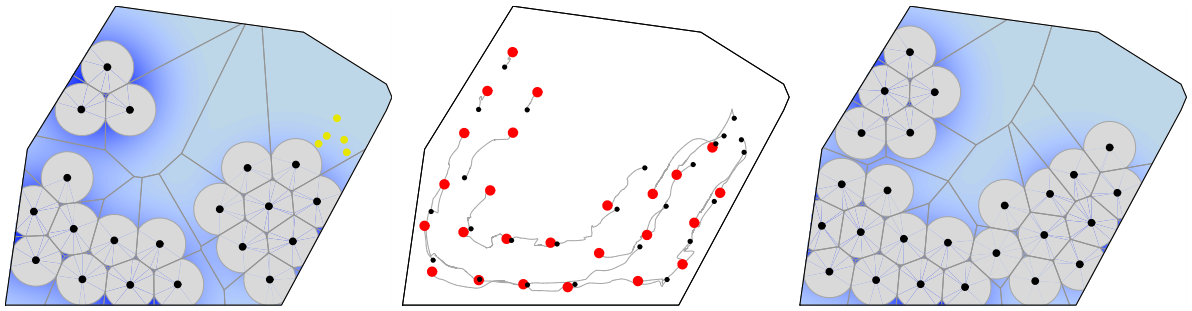

Figure 13. Area problem: adaptive behavior of the coordination algorithm. After the final configuration in Figure 12 is reached, five new agents (colored in yellow) enter the environment. The rest of the network adapts to the new situation satisfactorily. The left figure illustrates the location of the agents when the arrival of the new agents occurs, and the right figure show the final location of the network. The central figure illustrates the gradient descent flow from this event on.

Example 2 (Consensus): The asymptotic agreement algorithm proposed in [32] to solve the consensus problem is another example of this approach. For a fixed undirected graph $G = (\{1, \ldots, n\}, E)$, the function Φ_G is smooth, and its partial derivative takes the form

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

Clearly, this gradient is distributed with respect to the graph G itself. The implementation of the gradient control law leads to the algorithm $\dot{x}_i = \sum_{(i,j)\in E}(x_j - x_i)$, for $i \in \{1, \ldots, n\}$, which asymptotically achieves average-consensus, i.e., the final value upon which all agents agree can be proved to be equal to $\frac{1}{n}\sum_{i=1}^{n} x_i(0)$. □

Example 3 (Cohesiveness): Another example of this approach are the various coordination algorithms proposed in the literature to achieve cohesiveness [16, 19, 20]. For the complete graph $\mathcal{G}_{\text{complete}}$, the function $\mathcal{H}_{\text{cohe},\mathcal{G}_{\text{complete}}}$ is smooth on $Q^n \setminus \{(p_1, \ldots, p_n) \in (\mathbb{R}^2)^n \mid p_i = p_j \text{ for some } i, j \in \{1, \ldots, n\},\ i \neq j\}$, with

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

where we used the fact that dh/dR vanishes for $R \geq R_1$. According to the notions we introduced earlier, this gradient is spatially distributed over $\mathcal{G}_{\text{disk}}(R_1)$. The gradient descent algorithm guarantees that the network of agents will asymptotically approach the set of critical points of $\mathcal{H}_{\text{cohe},\mathcal{G}_{\text{complete}}}$. □

Not always does the aggregate objective 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, the corresponding gradient algorithm can not always be computed. If this is the case, a possible approach is the following: (i) consider constant-factor approximations of the objective function, (ii) identify those approximations whose gradient is spatially distributed with respect to an appropriate proximity graph, and (iii) implement as coordination algorithm the one that makes each agent follow the gradient of the approximation.

Example 4 (Mixed distortion-area problem): The coordination algorithm proposed in [22] 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 $\beta = r^2/(2 \operatorname{diam} Q)^2$,

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

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

$$\frac{\partial \mathcal{H}_{\frac{r}{2}}}{\partial p_i}(P) = 2\,\mathrm{M}\left(V_{i,\frac{r}{2}}(P)\right) \cdot \left(\mathrm{CM}\left(V_{i,\frac{r}{2}}(P)\right) - p_i\right) - \left(\frac{r^2}{4} + b\right) \int_{\operatorname{arc}(\partial V_{i,\frac{r}{2}}(P))} n_{\overline{B}(p_i,\frac{r}{2})}\, \phi\,,$$

where we recall that $\operatorname{arc}(\partial V_{i,\frac{r}{2}}(P))$ denotes the union of the arcs in $\partial V_{i,\frac{r}{2}}(P)$. Clearly, the gradient of $\mathcal{H}_{\frac{r}{2}}$ is spatially distributed over $\mathcal{G}_{\mathrm{LD}}(\frac{r}{2})$, and therefore, the coordination algorithm based on the corresponding gradient control law is implementable over a network with r-limited interactions. Figure 14 illustrates the execution of this algorithm. □

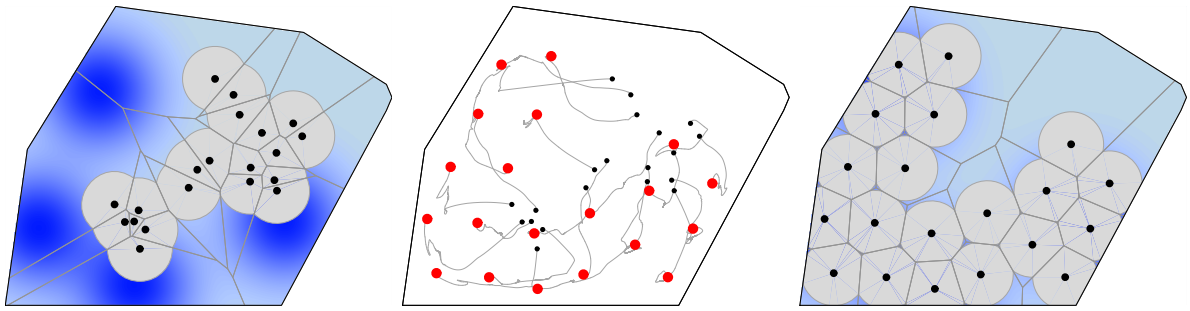

Figure 14. Mixed distortion-area problem: 20 **mobile agents in a convex polygon follow the gradient of** $\mathcal{H}_{\frac{r}{2}}$**. The density function** ϕ **and the environment are the same as in Figure 10. Each agent operates with a finite radius** $r = .5$**. From the constant-factor approximation** (4)**, the absolute error is less than or equal to** $(\beta - 1)\mathcal{H}_{\frac{r}{2}}(P_{\mathsf{final}}) \approx 2.89$**, 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 execution in Figure 10 is approximately equal to** 4.24%**.**

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, that implemented over each agent of the network would reasonably perform the desired 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 objective function that encodes the desired coordination task and by showing that this function is optimized along the execution of the coordination algorithm.

Example 5 (Move-away-from-closest-neighbor): Consider the coordination algorithm studied in [23] where each agent moves away from its closest neighbor (see Figure 15). 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

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

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 communication 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 configurations where each agent is at the incenter of its own Voronoi region (recall that the incenter set of a polygon is the set of centers of the maximum-radius spheres contained in the polygon). □

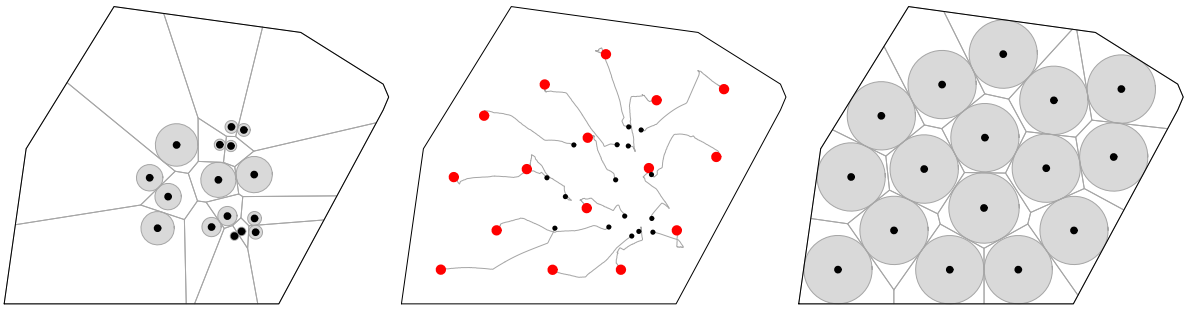

Figure 15. Non-interference problem: "move-away-from-closest-neighbor" algorithm for 16 mobile agents in a convex polygon. The left (respectively, right) figure illustrates the initial (respectively, final) locations and Voronoi partition. The central figure illustrates the network evolution. For each agent i, the ball of maximum radius contained in $V_i(P)$ and centered at p_i is plotted in light gray.

Example 6 (Flocking): Consider the coordination algorithm analyzed in [15] 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. □

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 objective function encoding the desired coordination 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 set up, it should be checked that it is spatially distributed and its asymptotic convergence properties should be characterized.

Example 7 (Non-interference problem): Consider the aggregate objective function $\mathcal{H}_{\text{SP}}$ defined in equation (5). Consider the alternative expression,

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

where $\text{sm}_W(p)$ is the distance from p to the boundary of the convex polygon W, i.e., $\text{sm}_W(p) = \text{dist}(p, \partial W)$. Now, for $i \in \{1,\ldots,n\}$, consider $\text{sm}_{V_i(P)}$ as a local objective function. Assuming that the Voronoi cell $V_i(P)$ remains fixed, then one can implement the (generalized) gradient ascent of $\text{sm}_{V_i(P)}$ as the control law for the agent p_i. One can show [23] that this interaction law precisely corresponds to the strategy "move-away-from-closest-neighbor" discussed in Example 5. A related strategy consists of each agent moving toward the incenter of its own Voronoi cell. The latter strategy can also be shown to make $\mathcal{H}_{\text{SP}}$ monotonically non-decreasing and to enjoy analogous asymptotic convergence properties. □

Example 8 (Worst-case problem): Consider the aggregate objective function

$$\mathcal{H}_{\text{DC}}(P) = \max_{q\in Q}\Big\{\min_{i\in\{1,\ldots,n\}} \|q - p_i\|\Big\} = \max_{i\in\{1,\ldots,n\}} \text{lg}_{V_i(P)}(p_i)\,,$$

where $\text{lg}_W(p)$ is the maximum distance from p to the boundary of the convex polygon W, i.e., $\text{lg}_W(p) = \max_{q\in W}\|q - p_i\|$. Now, for $i \in \{1,\ldots,n\}$, consider $\text{lg}_{V_i(P)}$ as a local objective function. Assuming that the Voronoi cell $V_i(P)$ remains fixed, then one can implement the (generalized) gradient descent of $\text{lg}_{V_i(P)}$ as the control law for the agent p_i. One can show [23] that this interaction law precisely corresponds to the strategy "move-toward-the-furthest-away-vertex-in-own-cell." A related strategy consists of each agent moving toward

the circumcenter of its own Voronoi cell (recall that the circumcenter of a polygon is the center of the minimum-radius sphere that contains it). Both strategies can be shown to make $\mathcal{H}_{\text{DC}}$ monotonically non-increasing and enjoy similar asymptotic convergence properties. These ideas can be combined in other settings with different capabilities of the mobile agents, e.g., in higher dimensional spaces (see Figure 16). □

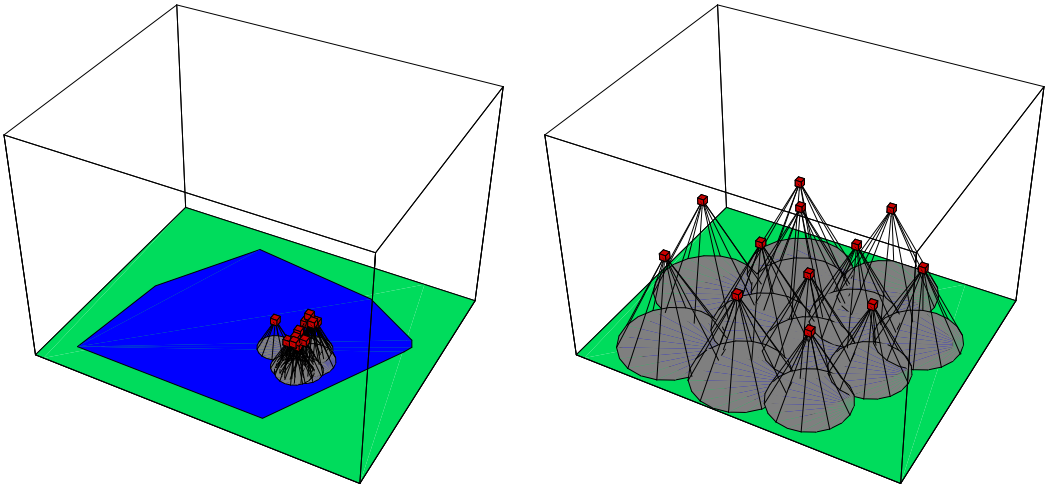

Figure 16. Worst-case scenario: "move-toward-the-circumcenter" algorithm for 12 **mobile agents. Each agent illuminates a vertical cone with a fixed and common aspect ratio. Each agent determines its Voronoi region within the planar polygon (the same as in Figure 15) and then moves its horizontal position toward the circumcenter of its Voronoi cell and its vertical position to the minimal height spanning its own Voronoi cell. The left (respectively, right) figure illustrates the initial (respectively, final) locations.**

Example 9 (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 [25, 26, 28] 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 objective function V_{diam} is monotonically non-increasing along the execution of the Circumcenter Algorithm. Using the LaSalle Invariance Principle for closed algorithms, one can indeed characterize the asymptotic correctness properties of the Circumcenter Algorithm over $\mathcal{G}$. See Figure 17 for an illustration of its execution. □

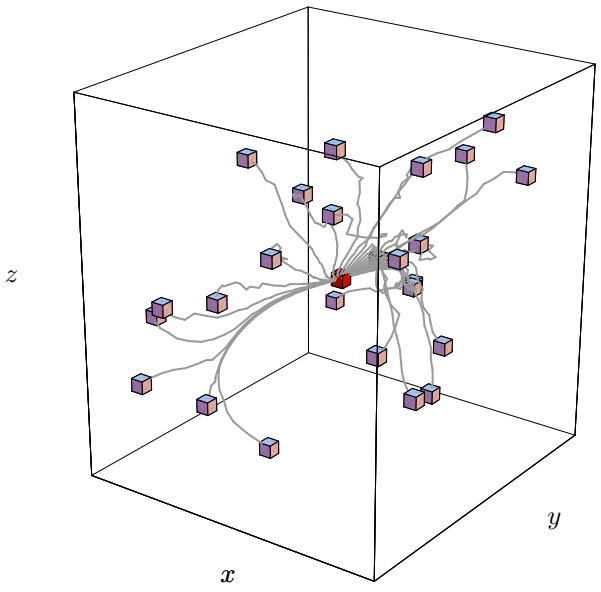

Figure 17. Evolution of the Circumcenter Algorithm in $\mathbb{R}^3$. The LaSalle Invariance Principle allows us to establish the algorithm's correctness under fairly general conditions. For instance, in this figure at each time step each agent selects $\mathcal{G}_{\mathsf{disk}}(r)$ or $\mathcal{G}_{\mathsf{LD}}(\frac{r}{2})$ to compute its set of neighbors.

Designing the coordination algorithm by composing different behaviors

This final approach builds on the methods presented above. An idea for the composition of behaviors is to implement one coordination algorithm on most of the network agents and a second coordination algorithm on the other agents. Coupling two algorithms in this parallel fashion results in interesting overall network behaviors. For example, one may prescribe an open-loop motion on some of the network agents (e.g., specifying that some particular agents must stay fixed or follow a desired path) and implement a feedback law for the others. Examples of this approach include (1) the formation control strategy in [26] to make the network form a straight line, and (2) the leader-following algorithm proposed in [15] to make the network flock in a pre-specified direction. Along these lines, it is interesting to explore more general parallel, serial and hierarchical approaches to the composition of behaviors.

Conclusions

We have surveyed a set of recent tools (proximity graphs, spatially distributed maps, aggregate objective functions, circulant matrices, and LaSalle invariance principles) that we believe are important in distributed motion coordination. We have also identified various

approaches to the design of coordination algorithms and shown the wide applicability of the proposed tools in these approaches, see Table 1. We believe that the coming years will witness an intense development of the field of distributed coordination and of its practical use in applications for multiple vehicles and sensor networks.

Agent motion direction	**Formal description**	**Distributed info**	**Lyapunov function**	**Asymptotic convergence**	**Ref.**
centroid of Voronoi cell	$\dot{p}_i = \mathrm{CM}(V_i(P)) - p_i$	Voronoi neighbors	$\mathcal{H}_{\mathrm{C}}$	centroidal Voronoi configurations	[21]
weighted average normal of $\frac{r}{2}$-limited Voronoi cell	$\dot{p}_i = \int_{\mathrm{arc}(\partial V_{i,\frac{r}{2}}(P))} n_{\overline{B}(p_i,\frac{r}{2})}\, \phi$	r-disk neighbors	$\mathcal{H}_{\mathrm{area},\frac{r}{2}}$	area-centered Voronoi configurations	[22]
average of neighbors	$\dot{p}_i = \sum_{j\in\mathcal{N}_G(i)}(p_j - p_i)$	neighbors in fixed G	Φ_G	Consensus	[32]
away from closest neighbor	$\dot{p}_i = \mathrm{Ln}(\partial\, \mathrm{sm}_{V_i(P)})(P)$	Voronoi neighbors	$\mathcal{H}_{\mathrm{SP}}$	Incenter Voronoi configurations	[23]
furthest-away vertex in Voronoi cell	$\dot{p}_i = -\mathrm{Ln}(\partial\, \mathrm{lg}_{V_i(P)})(P)$	Voronoi neighbors	$\mathcal{H}_{\mathrm{DC}}$	Circumcenter Voronoi configurations	[23]
circumcenter of neighbors' and own position	$p_i(t+1) = p_i(t) + \lambda_i^* \cdot (\mathrm{CC}(\mathrm{M}_i) - p_i)$	r-disk neighbors	V_{diam}	rendezvous	[25]

Table 1. Summary of example algorithms. In the interest of brevity, we refer to the corresponding references for the notation employed.

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] M. B. Miller and B. L. Bassler, "Quorum sensing in bacteria," *Annual Review of Microbiology*, vol. 55, pp. 165–199, 2001.

[2] A. Okubo, "Dynamical aspects of animal grouping: swarms, schools, flocks and herds," *Advances in Biophysics*, vol. 22, pp. 1–94, 1986.

[3] J. K. Parrish, S. V. Viscido, and D. Grunbaum, "Self-organized fish schools: an examination of emergent properties," *Biological Bulletin*, vol. 202, pp. 296–305, 2002.

[4] A. R. Sinclair, *The African Buffalo, A Study of Resource Limitation of Population*. Chicago, IL: The University of Chicago Press, 1977.

[5] S. Gueron and S. A. Levin, "Self-organization of front patterns in large wildebeest herds," *Journal of Theoretical Biology*, vol. 165, pp. 541–552, 1993.

[6] G. W. Barlow, "Hexagonal territories," *Animal Behavior*, vol. 22, pp. 876–878, 1974.

[7] T. D. Seeley and S. C. Buhrman, "Group decision-making in swarms of honey bees," *Behavioral Ecology and Sociobiology*, vol. 45, pp. 19–31, 1999.

[8] K. J. Stewart and A. H. Harcourt, "Gorillas vocalizations during rest periods - signals of impending departure," *Behaviour*, vol. 130, pp. 29–40, 1994.

[9] S. Boinski and A. F. Campbell, "Use of trill vocalizations to coordinate troop movement among whitefaced capuchins - a 2nd field-test," *Behaviour*, vol. 132, pp. 875–901, 1995.

[10] I. D. Couzin, J. Krause, N. R. Franks, and S. A. Levin, "Effective leadership and decision-making in animal groups on the move," *Nature*, vol. 433, pp. 513–516, 2005.

[11] L. Conradt and T. J. Roper, "Group decision-making in animals," *Nature*, vol. 421, no. 6919, pp. 155–158, 2003.

[12] 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.

[13] C. Belta and V. Kumar, "Abstraction and control for groups of robots," *IEEE Transactions on Robotics*, vol. 20, no. 5, pp. 865–875, 2004.

[14] 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.

[15] 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.

[16] H. Tanner, A. Jadbabaie, and G. J. Pappas, "Stability of flocking motion," tech. rep., Department of Computer and Information Science, University of Pennsylvania, Jan. 2003.

[17] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: Algorithms and theory," *IEEE Transactions on Automatic Control*, 2005. To appear.

[18] E. Klavins, R. Ghrist, and D. Lipsky, "A grammatical approach to self-organizing robotic systems," *IEEE Transactions on Automatic Control*, 2005. To appear.

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

[20] 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, no. 8, pp. 1292–1302, 2004.

[21] J. Cortés, S. Martínez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 2, pp. 243–255, 2004.

[22] J. Cortés, S. Martínez, and F. Bullo, "Spatially-distributed coverage optimization and control with limited-range interactions," *ESAIM. Control, Optimisation & Calculus of Variations*, vol. 11, pp. 691–719, 2005.

[23] J. Cortés and F. Bullo, "Coordination and geometric optimization via distributed dynamical systems," *SIAM Journal on Control and Optimization*, June 2005. To appear.

[24] E. Frazzoli and F. Bullo, "Decentralized algorithms for vehicle routing in a stochastic time-varying environment," in *IEEE Conf. on Decision and Control*, (Paradise Island, Bahamas), pp. 3357–3363, Dec. 2004.

[25] 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.

[26] 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* (V. Kumar, N. E. Leonard, and A. S. Morse, eds.), vol. 309 of *Lecture Notes in Control and Information Sciences*, pp. 257–282, New York: Springer Verlag, 2004.

[27] 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.

[28] J. Cortés, S. Martínez, and F. Bullo, "Robust rendezvous for mobile autonomous agents via proximity graphs in arbitrary dimensions," *IEEE Transactions on Automatic Control*, 2005. To appear.

[29] A. M. Bruckstein, N. Cohen, and A. Efrat, "Ants, crickets, and frogs in cyclic pursuit," Tech. Rep. 9105, Technion – Israel Institute of Technology, Haifa, Israel, July 1991. Center for Intelligent Systems.

[30] J. A. Marshall, M. E. Broucke, and B. A. Francis, "Formations of vehicles in cyclic pursuit," *IEEE Transactions on Automatic Control*, vol. 49, no. 11, pp. 1963– 1974, 2004.

[31] V. Sharma, M. Savchenko, E. Frazzoli, and P. Voulgaris, "Time complexity of sensor-based vehicle routing," in *Robotics: Science and Systems*, Cambridge, MA: MIT Press, 2005. To appear.

[32] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1520–1533, 2004.

[33] L. Moreau, "Stability of multiagent systems with time-dependent communication links," *IEEE Transactions on Automatic Control*, vol. 50, no. 2, pp. 169–182, 2005.

[34] W. Ren and R. W. Beard, "Consensus seeking in multi-agent systems using dynamically changing interaction topologies," *IEEE Transactions on Automatic Control*, vol. 50, no. 5, pp. 655–661, 2005.

[35] V. J. Lumelsky and K. R. Harinarayan, "Decentralized motion planning for multiple mobile robots: the cocktail party model," *Autonomous Robots*, vol. 4, no. 1, pp. 121–135, 1997.

[36] R. C. Arkin, *Behavior-Based Robotics*. Cambridge, MA: MIT Press, 1998.

[37] T. Balch and L. E. Parker, eds., *Robot Teams: From Diversity to Polymorphism*. Natick, MA: A K Peters Ltd., 2002.

[38] A. Howard, M. J. Matarić, and G. S. Sukhatme, "Mobile sensor network deployment using potential fields: A distributed scalable solution to the area coverage problem," in *International Conference on Distributed Autonomous Robotic Systems (DARS02)*, (Fukuoka, Japan), pp. 299–308, June 2002.

[39] 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.

[40] N. A. Lynch, *Distributed Algorithms*. San Mateo, CA: Morgan Kaufmann Publishers, 1997.

[41] M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf, *Computational Geometry: Algorithms and Applications*. New York: Springer Verlag, 2 ed., 2000.

[42] 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: John Wiley, 2 ed., 2000.

[43] M. Mesbahi, "On state-dependent dynamic graphs and their controllability properties," *IEEE Transactions on Automatic Control*, vol. 50, no. 3, pp. 387–392, 2005.

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

[45] R. Diestel, *Graph Theory*, vol. 173 of *Graduate Texts in Mathematics*. New York: Springer Verlag, 2 ed., 2000.

[46] C. D. Meyer, *Matrix Analysis and Applied Linear Algebra*. Philadelphia, PA: SIAM, 2001.

[47] P. J. Davis, *Circulant Matrices*. Providence, RI: American Mathematical Society, 2 ed., 1994.

[48] S. Martínez, F. Bullo, J. Cortés, and E. Frazzoli, "Synchronous robotic networks and complexity of control and communication laws," Jan. 2005. Preprint. Available electronically at `http://xxx.arxiv.org/math.OC/0501499`.

[49] D. G. Luenberger, *Linear and Nonlinear Programming*. Reading, MA: Addison-Wesley, 2 ed., 1984.

[50] A. R. Teel, "Nonlinear systems: discrete-time stability analysis." Lecture Notes, University of California at Santa Barbara, 2004.