Workshop on Network Embedded
Sensing and Control (NESC)

Notes on Multi-Agent Motion Coordination: Models and Algorithms

Francesco Bullo
Mechanical Engineering Department
Center for Control, Dynamical Systems and Computation
University of California at Santa Barbara
http://motion.mee.ucsb.edu

Extended Abstract
Network Embedded Sensing and Control
University of Notre Dame, October 17-18, 2005

Motion coordination is an extraordinary phenomenon in biological systems such as schools of fish and serves as a remarkable tool for man-made groups of robotic vehicles and active sensors. Although each individual agent has no global knowledge about the group as a whole or about the surrounding environment, complex coordinated behaviors emerge from local interactions. From a scientific point of view, the study of motion coordination poses novel challenges for systems and control theory. A comprehensive understanding of this phenomenon requires the joint ambitious study of mobility, communication, computation, and sensing aspects.

In this brief document, we review some of our recent work on models and algorithms for coordinating the motion of multi-agent networks. In Section 1, we discuss models and classifications for multi-agent networks, i.e., groups of robotic agents that can sense, communicate and take local control actions. For these networks, we introduce basic notions of communication and control algorithms, coordination tasks and time complexity. Earlier efforts in this direction are documented in [1, 2]; our treatment is influenced by [3, 4] and presented in detail in [5].

In Section 2, we discuss various basic algorithms for (i) rendezvous at a point and (ii) deployment over a given region. The proposed control and communication algorithms achieve these various coordination objectives requiring only spatially-distributed information or, in other words, single-hop communication. These rendezvous and deployment scenarios are treated extensively in [6, 7] and in [8, 9, 10], respectively. Early efforts on related problems include [11, 12]. The proposed models and examples shed some light on a novel class of control problems with insightful connections to the disciplines of distributed algorithms, geometric optimization, and algorithmic robotics.

1 Robotic networks and complexity

The global behavior of a robotic network arises from the combination of the local actions taken by its members. Each agent in the network can perform a few basic tasks such as sensing, communicating, processing information and moving according to it. The many ways in which these capabilities can be integrated make a robotic network a versatile and, at the same time, complex system. To understand the trade-offs between performance, reliability and costs, it seems appropriate to propose a modeling framework where the execution of different coordination algorithms can be appropriately formalized, analyzed and compared.

We consider *uniform networks of robotic agents* defined by a tuple $\mathcal{S} = (I, \mathcal{A}, E_{\mathrm{cmm}})$ consisting of a set of unique identifiers $I = \{1, \dots, N\}$, a collection of control systems $\mathcal{A} = \{A^{[i]}\}_{i \in I}$, with $A^{[i]} = (X, U, X_0, f)$, and a map E_{cmm} from X^N to the subsets of $I \times I$ called the *communication edge map*. Here, (X, U, X_0, f) is a control system with state space $X \subset \mathbb{R}^d$, input space U, set of allowable initial states $X_0 \subset X$, and system dynamics $f: X \times U \to X$. An edge between two identifiers in E_{cmm} implies the ability of the corresponding two agents to exchange messages. A *control and communication law* for $\mathcal{S}$ consists of the sets:

1. $\mathbb{T} = \{t_\ell\}_{\ell \in \mathbb{N}} \subset \bar{\mathbb{R}}_+$, an increasing sequence of time instants, called *communication schedule*;
2. L, called the *communication language*, whose elements are called *messages*;
3. W, set of values of some *logic variables* $w^{[i]} \in W$, $i \in I$, and $W_0 \subseteq W$, subset of *allowable initial values*. These sets correspond to the capability of agents to allocate additional variables and store sensor or communication data;

and the maps:

4. $\text{msg}: \mathbb{T} \times X \times W \times I \to L$, called *message-generation function*;
5. $\text{stf}: \mathbb{T} \times W \times L^N \to W$, called *state-transition function*;
6. $\text{ctl}: \bar{\mathbb{R}}_+ \times X \times X \times W \times L^N \to U$, called *control function*.

To implement a control and communication law each agent performs the following sequence or cycle of actions. At each instant $t_\ell \in \mathbb{T}$, each agent i communicates to each agent j such that (i, j) belongs to $E_{\text{cmm}}(x^{[1]}, \dots, x^{[N]})$. Each agent i sends a message computed by applying the message-generation function to the current values of t_ℓ, $x^{[i]}$ and $w^{[i]}$. After a negligible period of time, agent i resets the value of its logic variables $w^{[i]}$ by applying the state-transition function to the current value of $w^{[i]}$, and to the messages $y^{[i]}(t_\ell)$ received at t_ℓ. Between communication instants, i.e., for $t \in [t_\ell, t_{\ell+1})$, agent i applies a control action computed by applying the control function to its state at the last sample time $x^{[i]}(t_\ell)$, the current values of $x^{[i]}$ and $w^{[i]}$, and to the messages $y^{[i]}(t_\ell)$ received at t_ℓ.

Some remarks are appropriate. In our present definition, all agents are identical and implement the same algorithm; in this sense the control and communication law is called *uniform* (or anonymous). If $W = W_0 = \emptyset$, then the control and communication law is *static* (or memoryless) and no state-transition function is defined. It is also possible for a law to be *time-independent* if the three relevant maps do not depend on time. Finally, let us also remark that this is a synchronous model in which all agents share a common clock.

Next, we establish the notion of coordination task and of task achievement by a robotic network. A *(static) coordination task* for a network $\mathcal{S}$ is a map $\mathcal{T}: X^N \to \{\texttt{true}, \texttt{false}\}$. Additionally, let $\mathcal{CC}$ be a control and communication law for $\mathcal{S}$. We say that $\mathcal{CC}$ *achieves* the task $\mathcal{T}$ if for all initial conditions $x_0^{[i]} \in X_0$, the corresponding network evolution $t \mapsto x(t)$ has the property that there exists $T \in \mathbb{R}_+$ such that $\mathcal{T}(x(t)) = \texttt{true}$ for $t \geq T$.

In control-theoretic terms, achieving a task means establishing a convergence or stability result. Beside this key objective, one might be interested in efficiency as measured by required communication service, required control energy or by speed of completion. We focus on the latter notion. The *time complexity to achieve $\mathcal{T}$ with $\mathcal{CC}$* is

$$\text{TC}(\mathcal{T}, \mathcal{CC}) = \sup \left\{ \text{TC}(\mathcal{T}, \mathcal{CC}, x_0) \mid x_0 \in X_0^N \right\},$$

where $\text{TC}(\mathcal{T}, \mathcal{CC}, x_0) = \inf \{\ell \mid \mathcal{T}(x(t_k)) = \texttt{true}\,,\ \forall k \geq \ell\}$, and where $t \mapsto (x(t))$ is the evolution of $(\mathcal{S}, \mathcal{CC})$ from x_0. The *time complexity of $\mathcal{T}$* is

$$\text{TC}(\mathcal{T}) = \inf \{\text{TC}(\mathcal{T}, \mathcal{CC}) \mid \mathcal{CC} \text{ achieves } \mathcal{T}\}\,.$$

Some ideas on how to define meaningful notions of communication complexity are discussed in [5]. In the following discussion, we describe certain coordination algorithms, which have been cast into this modeling framework and whose time complexity properties have been analyzed.

2 Example algorithms and tasks

Key problems in motion coordination include the design of strategies for flocking, motion planning, collision avoidance and others. Numerous such problems remain interesting open challenges as of today. For example, it is still not clear how to prescribe the agents' motion in such a way as to achieve a generic prescribed geometric pattern; note that certain impossibility results are known [1]. Typically, coordination objectives are characterized via appropriate utility functions. We illustrate our approach by discussing two basic types of problems: rendezvous and deployment.

Aggregation algorithms

The rendezvous objective (also referred to as the gathering problem) is to achieve agreement over the location of the agents, that is, to steer each agent to a common location. An early reference on this problem is [11]. We consider two scenarios which differ in the agents' sensing/communication capabilities and the environment to which the agents belong. First [6], we consider the problem of rendezvous for agents equipped with *range-limited communication* in obstacle-free environments. In this case, each agent is capable of sensing its position in the Euclidean space $\mathbb{R}^d$ and can communicate it to any other robot within a given distance r. This communication service is modeled by the r-disk graph, in which two agents are neighbors if and only if their Euclidean distance is less than or equal to r. Second [7], we consider *visually-guided agents*. Here the agents are assumed to belong to a nonconvex simple polygonal environment Q. Each agent can sense within line-of sight any other agent as well as sense the distance to the boundary of the environment. The relationship between the agents can be characterized by the so-called visibility graph: two agents are neighbors if and only if they are mutually visible to each other.

In both scenarios, the rendezvous problem cannot be solved with distributed information unless the agents' initial positions form a connected sensing/communication graph. Arguably, a good property of any rendezvous algorithm is that of maintaining connectivity between agents. This connectivity-maintenance objective is interesting on its own. It turns out that this objective can be achieved through local constraints on the agents' motion. Motion constraint sets that maintain connectivity are designed in [11, 7] by exploiting the geometric properties of disk and visibility graphs.

These discussions lead to the following algorithm that solves the rendezvous problems for both communication scenarios. The agents execute what we shall refer to as the *Circumcenter Algorithm*; here is an informal description. Each agent iteratively performs the following tasks:

```
1: acquire neighbors' positions
2: compute connectivity constraint set
3: moves toward the circumcenter of the point set comprised of its neighbors and of itself,
   while remaining inside the connectivity constraint set.
```

One can prove that, under technical conditions, the algorithm does achieve the rendezvous task in both scenarios; see [6, 7]. Additionally, when $d = 1$, it can be shown that the time complexity of this algorithm is $\Theta(N)$; see [5].

Deployment algorithms

The problem of deploying a group of agents over a given region of interest can be tackled with the following simple heuristic. Each agent iteratively performs the following tasks:

```
1: acquire neighbors' positions
2: compute own dominance region
3: move towards the center of own dominance region
```

This short description can be made accurate by specifying what notions of dominance region and of center are to be adopted. In what follows we mention two examples and refer to [8, 9, 10] for more details.

First, we consider the *area-coverage deployment problem* in a convex polygonal environment. The objective is to maximize the area within close range of the mobile nodes. This models a scenario in which the nodes are equipped with some sensors that take measurements of some physical quantity in the environment, e.g., temperature or concentration. Assume that certain regions in the environment are more important than others and describe this by a density function ϕ. This problems leads to the coverage performance metric

$$\mathcal{H}_{\rm ave}(p_1,\dots,p_N) = \int_Q \min_{i\in\{1,\dots,N\}} f(\|q-p_i\|)\,\phi(q)dq = \sum_{i=1}^N \int_{V_i} f(\|q-p_i\|)\phi(q)dq.$$

Here p_i is the position of the ith node, f measures the performance of an individual sensor, and $\{V_1,\dots,V_N\}$ is the Voronoi partition of the nodes $\{p_1,\dots,p_N\}$. If we assume that each node obeys a first order dynamical behavior, then a simple gradient scheme can be easily implemented in a spatially-distributed manner. Following the gradient of $\mathcal{H}_{\rm ave}$ corresponds, in the previous algorithm, to defining (1) the dominance regions to be the Voronoi cells generated by the agents, and (2) the center of a region to be the centroid of the region (if $f(x)=x^2$). Because the closed-loop system is a gradient flow for the cost function, performance is locally, continuously optimized. As a special case, when the environment is a segment and $\phi=1$, the time complexity of the algorithm can be shown to be $O(N^3\log(N\epsilon^{-1}))$, where ϵ is a threshold value below which we consider the task accomplished; see [5].

Second, we consider the problem of deploying *to maximize the likelihood of detecting a source.* For example, consider devices equipped with acoustic sensors attempting to detect a sound-source (or, similarly, antennas detecting RF signals, or chemical sensors localizing a pollutant source). For a variety of criteria, when the source emits a known signal and the noise is Gaussian, we know that the optimal detection algorithm involves a matched filter, that detection performance is a function of signal-to-noise-ratio, and, in turn, that signal-to-noise ratio is inversely proportional to the sensor-source distance. In this case, the appropriate cost function is

$$\mathcal{H}_{\rm worst}(p_1,\dots,p_N) = \max_{q\in Q} \min_{i\in\{1,\dots,N\}} f(\|q-p_i\|) = \max_{q\in V_i} f(\|q-p_i\|),$$

and a greedy motion coordination algorithm is for each node to move toward the circumcenter of its Voronoi cell. A detailed analysis [10] shows that the detection likelihood is inversely proportional to the circumradius of each node's Voronoi cell, and that, if the nodes follow this algorithm, then the detection likelihood increases monotonically as a function of time.

Acknowledgments

This document summarizes some results of joint work with Anurag Ganguli, Ketan Savla, Jorge Cortés, Emilio Frazzoli, and Sonia Martínez. The author thanks the organizers of the Workshop on Network Embedded Sensing and Control for the warm hospitality throughout the meeting and the opportunity to present this work. The author also gratefully acknowledges the partial support of ONR YIP Award N00014-03-1-0512, NSF SENSORS Award IIS-0330008, and ARO MURI Award W911NF-05-1-0219.

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] N. Santoro, "Distributed computations by autonomous mobile robots," in *SOFSEM 2001: Conference on Current Trends in Theory and Practice of Informatics (Piestany, Slovak Republic)* (L. Pacholski and P. Ruzicka, eds.), vol. 2234 of *Lecture Notes in Computer Science*, pp. 110–115, New York: Springer Verlag, 2001.

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

[4] 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.

[5] 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`.

[6] J. Cortés, S. Martínez, and F. Bullo, "Robust rendezvous for mobile autonomous agents via proximity graphs in arbitrary dimensions," *IEEE Transactions on Automatic Control*, July 2004. To appear.

[7] A. Ganguli, J. Cortés, and F. Bullo, "On rendezvous for visually-guided agents in a nonconvex polygon," in *IEEE Conf. on Decision and Control*, (Seville, Spain), Dec. 2005. To appear.

[8] 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.

[9] 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.

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

[11] 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.

[12] 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.