2011 50th IEEE Conference on Decision and Control and
European Control Conference (CDC-ECC)
Orlando, FL, USA, December 12-15, 2011

Aggregation and Rendezvous in an Unbounded Domain without a Shared Coordinate System

Bruce Rogers, Gregory Fricke, Devendra P. Garg

***Abstract*—Many swarm robotics problems focus on the details of robotic dynamics, while ignoring certain other practical issues such as the boundedness of the exploration space, the probability of unsuccessful communication or sensing between agents, knowledge of a common coordinate system, or the initial distribution of robots. A two-phase control is devised which, under appropriate individual bounds, almost surely leads to successful rendezvous. We examine a likely and practical initial distribution, which leads to an exploration protocol wherein an agent is obligated to stay within a certain range of its initial position. Furthermore, the rate of convergence is primarily limited to the velocity limitations of the agents. We solve the aggregation and rendezvous problem with random initial conditions in an arbitrary, unbounded 2D domain without an *a priori* share coordinate system. We examine rates of successful aggregation under certain initial distributions, and rates of convergence under uncertain communication.**

I. Introduction

The aggregation and rendezvous problems are basic hurdles in swarm robotics and distributed control [1]–[3]. In aggregation, the goal is to develop a distributed control that allows an arbitrary number of agents to act together to achieve some task [4], [5]. For the rendezvous problem, the goal is for the agents to converge to a common position. We assume that the agents do not have a shared coordinate system and operate in an unbounded two-dimensional domain. Also, we assume communication constraints between the agents that are realistic for current technology. However, the control developed here is high level and rather abstract; we are not concerned with issues like obstacle avoidance, and the agents are assumed to be point masses operating in discrete time.

In contrast to our assumptions, rendezvous is relatively simple with a shared coordinate system and no bounds on communication. Each agent can broadcast its position to the others; they calculate the average position and move to that point. If communication is only local, averaging algorithms may produce many clusters of agents instead of a single swarm. Without shared coordinates, robots must first sense the position of neighbors to develop a shared map. In this paper, we explore a two stage control protocol, first utilizing random motion that allows the agents to develop shared coordinates. The agents then use local averaging to solve the rendezvous problem under some realistic constraints.

Communication restrictions between agents depend on mission specific parameters: the sensors of the robots, the size of the domain, etc. We make the following assumptions on the communication abilities of the agents and their ability to infer others' positions.

- **No shared map:** Agents must meet or sense each other in order to share coordinates.
- **Sensing radius:** Each robot is equipped with a sensor, such as a camera or laser range finder. Most commonly, such sensors have a sweep of less than 360 degrees, and exhibit sensing errors.
- **Simultaneous communication:** Each agent can maintain at most k simultaneous communications. For example, standard Bluetooth technology allows $k = 7$ simultaneous communications. It's important to note that it is not necessary to keep the same k partners over time.
- **Probabilistic communication radius:** Robots have a limited energy source and may be examining an area much larger than the power of the antennae can cover. In general, the probability of a successful communication decreases with distance. In what follows, we simplify matters in one of two ways: by assuming a fixed probability p_d of the failure of each communication attempt, or by assuming that the probability of a successful communication decays with distance.

We develop a control where agents first move randomly to create a cache of comrades with common coordinates. Each agent then selects some small number (bounded by k) of individuals from its cache and moves to the average position of those individuals. In Section II we give a random motion paradigm that supplies the agents with common coordinates and provides bounds on the time through simulations. In Section III we discuss the global dynamics of the averaging portion of the control. Section IV proposes future simulations and experiments and offers some concluding remarks.

II. Finding other agents in the field

The agents in our system wish to aggregate, but there is one major impediment: They do not have a shared coordinate system. They must sense each other in the field before communicating their positions to each other; it would be meaningless otherwise. Upon sensing each other, each agent records the sensed position of the newly met agent; in the following communication step, the agents share their own beliefs of their current and starting positions within their own coordinate frame. In this way, the coordinate frames of all agents become united. In fact, upon complete successful aggregation, each agent has a complete map of the starting points of the entire agent set relative to its own coordinate frame.

Let $x_i(t) \in \mathbb{R}^2$ be the position of agent i at time t. If there are N agents, the state space of the system is a subset of $\mathbb{R}^{2\times N}$. In applications, the state vector may include additional coordinates, e.g. velocity or orientation; however, without loss of generality, we treat aggregation as a purely positional problem.

We suppose that each agent has some open, bounded, simply connected sensing domain $D_i(t) \subset \mathbb{R}^2$, and that the shape of the D_i is the same for each i and all times. Since our agents are point masses, we assume without loss of generality that $i \in D_i(t)$ for all i and t. We also suppose the sensors are less than perfect; there is a failure probability p_s at each sensing opportunity. If $x_i(t) \in D_j(t)$ at time t, agent j senses i's presence with probability $1 - p_s$. If $x_i(t) \in D_j(t)$ *and* $x_j(t) \in D_i(t)$ at some time t, then agents i and j exchange coordinate systems and add each other's communication address (e.g., IP address) to their phone book with probability $(1-p_s)^2$. Also, since the agents can remember all agents they have previously encountered, two agents exchange not just their own coordinates upon sensing, but also the entire contents of their address and phone books.

At the abstract level, we need a protocol for agents to wander about and find each other. However, if the environment is unbounded and the agents move randomly (e.g., Brownian or Lévy motion), it's possible some of the agents will wander away from the rest. To prevent that possibility, we give the agents a wandering radius W_r. The agents' motions are restricted so that $\|x_i(0) - x_i(t)\| < W_r$ for all i, which is a simple control to implement since each agent has its own relative coordinate system. Let B_i be the ball of radius W_r centered at $x_i(0)$; this is agent i's wandering domain. Also, let D be the support of the initial distribution of agents, so that $x_i(0) \in D \subset \mathbb{R}^2$ for all i.

There is a natural graph theoretic exposition of the control. For basic graph theory definitions, see [6] for a classical treatment or [7]–[11] for an engineering viewpoint. We consider two different networks. First, at each time t, there is a sensing network $\mathcal{G}_t = (V, E_t)$ where the agents are the set of nodes and i is connected to j if and only if $x_i(t) \in D_j(t)$ and $x_j(t) \in D_i(t)$. Because distant agents may communicate if they have met previously, the union of all previous connections is key to aggregation. Then for a fixed time $T > 0$, set $\mathcal{G}(T) = (V, \bigcup_t^T E_t)$. That is, $i \sim j$ in $\mathcal{G}(T)$ if $i \sim j$ in $\mathcal{G}_t$ for any $t \leq T$. The idea of accumulating a set of changing edges is present in [14] and [12]; however, in those works the topology of the network does not explicitly depend on the geometry of the state space. Here, the cumulative sensing graph $\mathcal{G}(T)$ depends on how the wandering domains B_i overlap since two agents i and j can sense each other only if the distance between B_i and B_j is small enough (and certainly if $B_i \cap B_j \neq \emptyset$).

With this in mind, we next define the network $\mathcal{G}_B$ where the nodes are the sets B_i, and two wandering domains are connected if $B_i \cap B_j \neq \emptyset$. $\mathcal{G}_B$ is sometimes called the *nerve* of the cover $\bigcup B_i$. The following proposition essentially follows from the definitions.

Proposition 1: Suppose the dynamics of the individual agents are given by independent stationary ergodic stochastic processes such that the support of the stationary distribution of $x_i(t)$ is B_i. Then if the network of the wandering domains $\mathcal{G}_B$ is connected, for any $\epsilon > 0$ there is a (random) finite time τ such that the probability the cumulative sensing network $\mathcal{G}(\tau)$ is connected is $1 - \epsilon$.

Proof: It suffices to show that two agents in intersecting regions will sense each other in finite time. This is bound to happen by the recurrence property of ergodicity. Given any disc in the intersection of the regions, there is a finite time where both agents will be in any disc. Make the disc sufficiently small so that the agents must sense each other. If $p_s > 0$, the probability of sensing goes 1 as the two agents recurrently meet in the disc. ■

The "almost surely" in the proposition is with respect to the stationary measure. Also, the wandering domains do not need to be discs, but we use discs below because the agents have random initial orientations. In what follows, each agent i picks a point uniformly at random from B_i and moves there, and then repeats. These dynamics clearly meet the hypotheses of the Proposition. In the next subsection, we show by simulation that connection can be achieved in a reasonable amount of time.

Once $\mathcal{G}(t)$ is connected, the agents have essentially solved the aggregation problem. There is then a chain of contacts between any two agents, and thus any two agents can pass communications through that chain. Then every agent's address book will contain the phone number and coordinate system of every other agent. One issue is to develop a communication protocol so that the agents' address books are built up over time, and to minimize the adverse effects of call drops. Another issue not addressed in this paper is the required self-localization of each agent relative to its own coordinate frame; there are many methods in the literature addressing this practicality through odometry calibration and estimation [15], SLAM methods [16], [17], or (perhaps most relevant to this discussion) relative localization [18], [19].

A. Simulations for aggregation time

In this section we wish to determine how long it will take to connect the network $\mathcal{G}(t)$ with a realistic number of agents deployed in a realistic initial domain D. We focus on a small team of agents, $10 \leq N \leq 20$ with initial positions drawn from a two-dimensional normal distribution. For $N \to \infty$, the connectivity of $\mathcal{G}_B$ is well understood [13], but very large systems of robots are not feasible at this time.

We first need to determine constraints on the time until connectivity in best conditions before considering less ideal sensing. In the simulations, we assume perfect, omnidirectional sensing. That is, each agent has a sensing radius ρ, and two agents sense each other at time t if $\|x_i(t) - x_j(t)\| < \rho$. This represents a best case scenario of the discussion above. Without loss of generality, set the sensing radius of the agents to unity, $\rho \equiv 1$.

Suppose the initial position of each agent is drawn from a normal distribution with mean at the origin. The standard deviation in the x-coordinate is five times the sensing radius, and the standard deviation in the y-coordinate is the same as the sensing radius. (The covariance between x and y is 0). Such a distribution may be realistic in a scenario where the agents are, say, ground robots deployed by an aircraft in flight; a long, thin initial distribution makes sense. Under these chosen "mission parameters", the wandering radius W_r of each agent is set to 14. This value was chosen so that the wandering domain network ($\mathcal{G}_B$ above) is very likely to be connected. In simulations, the wandering domain networks were *not* connected only 16 times out of 11,000 instantiations. Additional experiments were run for $W_r = 10, \ldots, 20$. Clearly, the larger the radius, the more likely $\mathcal{G}_B$ is to be connected, but also the longer it takes for the cumulative sensing network to become connected because each agent has to search a larger area.

With mission parameters defined, we implement a simple agent dynamics. For each $t > 0$, $x_i(t)$ is chosen uniformly at random from agent i's wandering domain. The index t does not represent time in the sense of a ticking clock, but rather the number of actions the agents perform. These dynamics are chosen because we want the agents to aggregate as quickly as possible, but they can sense each other only if they are near enough to one another; independent draws from a uniform distribution mix the agents quickly. Also, for two agents it is easy compute the mean and variance of the sensing time.[1] For more than two agents, there are dependencies that make calculating the expected connection time of $\mathcal{G}(t)$ difficult.

Each agent moves from $x_i(t)$ to $x_i(t+1)$ in our simulation by utilizing a 2^{nd}-order unicycle model, i.e., one that is appropriate for control of a two-wheeled differential-drive mobile robot. At each simulation step, the proximity to other agents is evaluated. If one or more other agents is within the defined sensing range ρ, the agents are said to have met.

Figure 1 shows simulation results for $N = 10, 11, \ldots, 20$. For the initial distribution given above and each value of N, we calculate the first time t when the cumulative sensing network $\mathcal{G}(t)$ is connected. The times t are random, so we run the simulation 1000 iterations for each value of N with fresh initial conditions every iteration. The figure shows the $50^{th}, 75^{th}, 90^{th}$ and 99^{th} percentiles of the simulated aggregation times, as well as the worst case.

The y-axis in Figure 1 measures time in *number of random positions explored*. We see little improvement in performance as N increases from 10 to 20, and the times are very tightly distributed about the median (bottom line). Very rarely, the agents take longer to aggregate; see the crossed line corresponding to the 99^{th} percentile. This happens when the wandering domains have very small overlap. The results are promising: in 90 percent of the 1000 cases, only about 10 positions are necessary for the communication graph to become connected.

[1]The sensing time is a Bernoulli random variable with probability proportional to the areas of overlap.

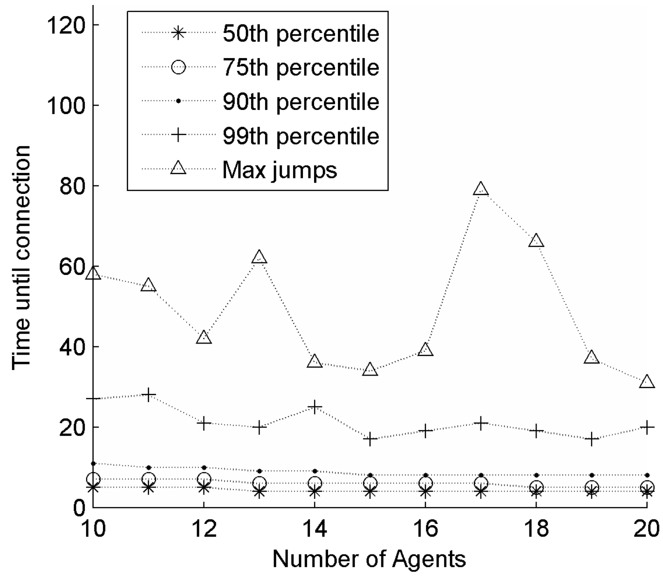

Fig. 1. Time until aggregation for $N = 10$ to 20 agents. The time is random, so 1000 iterations are simulated for each value of N, and percentiles are reported. The time units are number of positions explored.

III. Global Dynamics of Averaging Protocol

We now move to the rendezvous problem with communication constraints. Once the agents have aggregated and can share coordinates, they could broadcast their positions and rendezvous at the mean. However, broadcasting does not necessarily scale with the number of agents, whereas small-scale directed messaging does [20]. If, for example, there are dozens or hundreds of agents, each agent would have to *receive and process* all of those broadcast messages, or there would need to be a protocol to somehow synchronize the messages. Alternatively, if each agent only sends directed messages to a small number of other agents, results from random graph theory confirm that the typical number of incoming requests will be closely related to that small number; thus most agents will have a tenable number of requests to which to respond. As noted in the introduction we assume for practicality reasons that at most k simultaneous communications are possible.

We wish to keep communications to a minimum in order to minimize the detrimental effects of dropped calls. Suppose each agent i nominates exactly two neighbors i_1 and i_2. In this section we consider the global dynamics from the local updating

$$x_i(t+1) = \frac{x_{i_1}(t) + x_{i_2}(t)}{2}. \quad (1)$$

So at each time step, agent i goes to the mean of its two neighbors' previous positions. In the limit as $N \to \infty$, these dynamics are known to contract the unit disc, but the situation is less clear for a fixed finite number of agents.

A key practical concern is how the agents choose their two neighbors. The mathematical ideal is for each agent to choose from the others uniformly without replacement. If the agents have aggregated as in the previous section, they have access to the phone numbers of all the other agents and can choose from them uniformly. Even in the ideal case, there is no mathematical guarantee that the positions of the agents will converge to a common value. However, for large N with initial positions randomly distributed in a convex subset of $\mathbb{R}^2$, there are several heuristic reasons to believe the dynamics in (1) will cause the agents to converge. First, the dynamics define a contraction, so the most distant agents are bound to come closer together. Second, since each agent nominates exactly two neighbors, random graph theory tells us that the network of connections between agents will have a unique giant component for large N. That is, the vast majority of agents will be in the same communication class, so under (1) the positions of these agents will converge to the same value.

Finally, even if the communication network has many separate communication classes, the initial conditions are random, so the law of large numbers says that each communication class will converge to a point near the mean of the distribution. These observations fall well short of rigorous proof, but they are played out in simulation. We ran 1000 realizations of the dynamics with $N = 10, \ldots, 20$ and initial conditions uniformly distributed in a rectangle of size $W = 100$ on each side. These initial conditions are more dispersed than the more realistic distribution described in the previous section, so convergence should be less likely and slower. After 25 time steps of the dynamics from (1) we measured whether or not all the agents settled within a disc of diameter 1. If not, we consider this a failure and record the failure rates in the first row of Table I. For the sake of space, we record just the even numbers of agents. The others are similar.

TABLE I
NUMBER OF FAILURES IN 1000 SIM RUNS.

Number of agents	10	12	14	16	18	20
Case 1	29	32	25	36	21	16
Case 2	0	4	8	2	0	3

The second row shows the same measurement, but with a twist on the dynamics. Each agent picks two agents randomly and follows (1) for 10 time steps. Then each agent picks a fresh pair of neighbors and follows (1) for 10 more time steps. In the first case, agents in over 95 percent of the simulations converge to within 1 unit of each other after 25 time steps. In the second case, agents in more than 99 percent of the simulations converge to within 1 unit of each other after just 20 time steps. As in the previous section, time is measured in calculations performed, and the convergence is fast enough that it is practically limited only by the speed of the robot.

Even when the agents do not converge to a single point within 25 times steps (reported as a failure in the table), they come together into several clusters that lie on a common line in the plane. Even in the cases of "failure" the maximum distance between the agents is less than 3. Also, for $N = 250$ or 500 we found that the dynamics converge to within a disc of diameter 1 in every case out of 2000.

A. Probabilistic communication: Fixed probability

We now consider the case where successful communication during the rendezvous phase is not guaranteed. Intuitively, this should decrease the rate of convergence (and perhaps the likelihood of success). To investigate this, we modified the rendezvous simulation such that with probability p_d a communication attempt at a given time-step will be unsuccessful. In this case, a "communication attempt" is the request from one agent to one of its two nominated neighbors for their current positions. An unsuccessful attempt means that the requested information is not returned. If only one such request is successful, the requesting agent will move to the average of its current position and the responding agent's position. If neither request is successful, the requesting agent does not move. (Note that the responding agents do not take this communication into account in generating their own moves.)

We ran three sets of simulations of 100 runs each for values of $p_d = \{0.1, 0.5, 0.9\}$ and $N = 20$, and the results are shown in Figure 2. The y-axis of the figure is the maximum separation between any two agents. The plotted results give the median distances at the given time-step for each of the three values of p_d. Additionally, for the $p_d = 0.9$ case, the 25th and 75th percentile values are shown.

Astoundingly, even with a failure rate of $p_d = 0.5$ (that is, one out of two communication attempts is unsuccessful), the median max-distance is effectively zero before the 30th time-step. Compare this to the case of the previous subsection, i.e., perfect communication, where we mark the rendezvous successful if the max distance is ≤ 1 at completion of the 25th time-step. As expected, the rendezvous requires more time under decreased probability of successful communication; unexpectedly, the effect seems to be quite mild. Even for the case of $p_d = 0.9$, the rendezvous is successful. The rate is clearly much slower in this case, but successful rendezvous can still be expected. These results illustrate a lower bound for the time required for rendezvous under very ineffective communication.

B. Probabilistic communication: Range-dependence

We continue this investigation by making a more realistic assumption regarding the communication: the probability of successful communication decreases with increased range. We know well that the power level of radio frequency signals drops off as the square of the distance, so we might similarly expect that digital signals are less likely to be successfully received at greater distances.

With this motivation, we define the probability of successful communication as a function of the distance d between

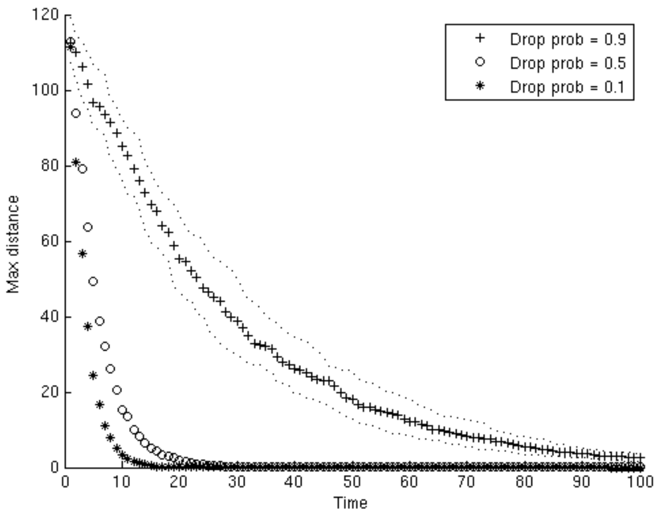

Fig. 2. Convergence rates subjected to various probabilities of successful communication.

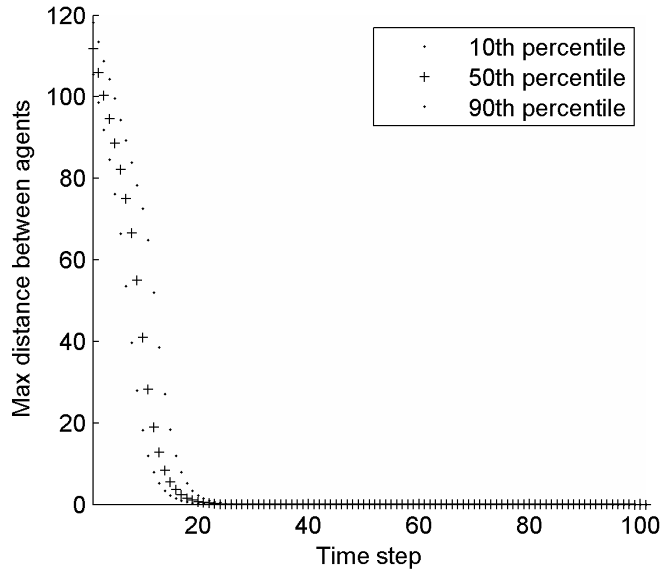

Fig. 3. Convergence rates subjected to the probability of successful communication defined as a function of distance.

any two agents attempting to communicate. For simplicity, we choose the sigmoid function shown in (2), where C_r is the communication radius and a is a shaping parameter that defines the sharpness of the dropoff.

$$p_d(d) = 1 - \frac{1}{1 + e^{a/2 - d*a/C_r}} \tag{2}$$

Choosing $W = 100$ (as previously), $C_r = 0.75W$ and $a = 5$, we have $p_d(0) = 0.924$ and $p_d(\sqrt{2}W) = 0.000979$ (where $\sqrt{2}W$ is the max separation any two agents could experience in this scenario).

We repeated the simulations for convergence using this function for communication probability. In 10,000 iterations, only two instantiations failed to converge within 100 time steps. The results of these simulations are given in Fig. 3 with the crossed points indicating the median max-separation between any two agents, and the dotted lines indicating the 10^{th} and 90^{th} percentiles. From the figure, it is clear that the convergence rates under this assumed function maintain the convergence rates at roughly those of the case where *all* communication is subjected to a 50% probability of failure (regardless of distance). Furthermore the results are very promising in the very tight distribution of convergence rate. It is likely that the shape of this convergence is highly dependent on the shape of the communication function (2), specifically on the sharpness of the cutoff.

IV. Discussion and Future Directions

We have considered distributed controls that are functions only of an agent's position. Real robots move in continuous time and continuous space, have limitations on their ability to dead-reckon their own positions, and have limitations in their ability to consistently and accurately measure the positions of other agents. In the future we seek to further refine our simulations to address these issues in more realistic and practical terms.

A serious challenge is to rendezvous a relatively small number of agents in a perhaps vast environment. We have given a two stage process where agents first find a shared coordinate systems and are able to communicate with each other. Then, each agent moves directly between two other agents in order to rendezvous at a common point. We choose only two others in order to limit dropped communications due to technology and protocol constraints.

We have been silent on how the agents know when to switch from one dynamics protocol to the other. One way is to suppose the agents know how many other agents there are, so each one has an address book with $N - 1$ slots. When an agent's address book becomes full, it votes to switch from the random dynamics of Section II to the deterministic dynamics of Section III. Once the vote is unanimous, the switch is made. This, however, adds requirements to the communication protocol.

A simple alternative method of deciding when to switch to the rendezvous protocol is to capitalize on the results of Section II. Specifically, we see that, for 10 or more agents utilizing this protocol, arbitrarily setting the time-to-switch to, say, $t = 100$ means that with high confidence the address books of all agents will be full. As the number of agents increases within the same distribution (i.e., the initial density increases) the number of time-steps additionally decreases. Clearly then, and especially within the mindset of swarming robotics wherein exactness is of less importance than generalized robustness, a time could be chosen within which some minimum number of agents (with a perhaps larger set of deployed agents) aggregate and rendezvous with a dictated likelihood appropriate for the given mission.

V. ACKNOWLEDGMENTS

The authors gratefully acknowledge the support of the Army Research Office under grant number W911NF-08-1-0106.

REFERENCES

[1] F. Bullo, J. Cortés and S. Martínez, *Distributed Control of Robotic Networks*. Princeton University Press, 2009.

[2] W. Burgard, M. Moors, C. Stachniss, and F.E. Schneider, "Coordinated multi-robot exploration," *IEEE Transactions on Robotics*, vol. 21, no. 3, pp. 376–386, 2005.

[3] A. Ganguli, S. Susca, S. Martinez, F. Bullo, and J. Cortes, "On collective motion in sensor networks: sample problems and distributed algorithms", in in *Proceedings of 44th IEEE Conference on Decision and Control and 2005 European Control Conference. (CDC-ECC '05)*, Dec. 2005, pp. 4239–4244.

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

[5] V. Gazi, "Swarm aggregations using artificial potentials and sliding-mode control", *IEEE Transactions on Robotics*, vol. 21, no. 6, pp. 1208–1214, 2005.

[6] B. Bollobás, *Modern Graph Theory*. Springer-Verlag, 1998.

[7] M. Mesbahi, "State-dependent graphs," in *Proceedings of 42nd IEEE Conference on Decision and Control (CDC2003)*, vol. 3, Dec. 2003, pp. 3058–3063.

[8] ——, "On state-dependent dynamic graphs and their controllability properties," in *43rd IEEE Conference on Decision and Control, 2004*, vol. 3, Dec. 2004, pp. 2473–2478.

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

[10] ——, "On the factorization, observability, and identification of the agreement protocol," in *Proceedings of the 16th Mediterranean Conference on Control and Automation*, Jun. 2008, pp. 1610 –1615.

[11] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods in Multiagent Networks*. Princeton University Press, 2010.

[12] L. Moreau, "Stability of Multiagent Systems With Time-Dependent Communication Links," *IEEE Transaction on Automatic Control*, vol. 50, no. 2, pp. 169-181, 2005.

[13] M. Penrose, *Random Geometric Graphs*. Oxford University Press, 2003.

[14] A. Tahbaz-Salehi and A. Jadbabaie, "A Necessary and Sufficient Condition for Consensus Over Random Networks, *IEEE Transactions on Automatic Control*, vol. 53, no. 3, pp. 791–795, 2008.

[15] G. Antonelli, S. Chiaverini and G. Fusco, "A calibration method for odometry of mobile robots based on the least-squares technique: theory and experimental validation," *IEEE Transaction on Robotics*, vol. 21, no. 5, pp. 994–1004, 2005.

[16] S. Thrun, W. Burgard and D. Fox, *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*, The MIT Press, 2005.

[17] A.I. Eliazar and R. Parr, "DP-SLAM 2.0," in *Proceedings of the 2004 IEEE International Conference on Robotics and Automation (ICRA'04)*, vol. 2, Apr. 2004, pp. 1314–1320.

[18] S.I. Roumeliotis and G.A. Bekey, "Distributed multirobot localization," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 781–795, 2002.

[19] J. Pugh, X. Raemy, C. Favre, R. Falconi and A. Martinoli, "A Fast Onboard Relative Positioning Module for Multirobot Systems", *IEEE/ASME Transactions on Mechatronics*, vol. 14, no. 2, pp. 151–162, 2009.

[20] W. Agassounon, "Distributed information retrieval and dissemination in swarm-based networks of mobile, autonomous agents," in *Proceedings of the 2003 IEEE Swarm Intelligence Symposium (SIS'03)*, Apr. 2003, pp. 152–159.