2015 American Control Conference
Palmer House Hilton
July 1-3, 2015. Chicago, IL, USA

Probabilistic Density Control for Swarm of Decentralized ON-OFF Agents with Safety Constraints

Nazlı Demir and Behçet Açıkmeşe

Abstract— **This paper presents a Markov chain based approach for the probabilistic density control of a swarm of autonomous "ON-OFF" agents. The proposed approach specifies the time evolution of the probabilistic density distribution by using a Markov chain, which guides the swarm to a desired steady-state final distribution, while satisfying the prescribed ergodicity and safety constraints. Prior research has developed a Markov chain based approach to control swarms of agents with full mobility. The main contribution of the current paper is generalizing this approach to a swarm of ON-OFF agents with limited mobility. We define ON-OFF agents as having limited mobility in the following sense: The agent either conforms to the motion induced by the environment or it remains motionless. This means that an ON-OFF agent has two possible actions, either accept the environmentally induced motion, "ON", or stop all the motion, "OFF". By using these binary control actions at the agent level, we develop a decentralized control architecture and algorithms that guide the swarm density distribution to a desired probabilistic density in the operational space. The agents make statistically independent probabilistic decisions on choosing to be "ON" or "OFF" based solely on their own states to achieve a desired swarm density distribution. The probabilistic approach is completely decentralized and does not require communication or collaboration between agents. Of course, any collaboration can be leveraged for better performance, which is the subject of future work. There are two new algorithms developed: An online ON-OFF policy computation method to generate a Markov matrix with the ergodicity and motion constraints but without the safety constraints, which can be viewed as generating a Markov matrix via the Metropolis-Hastings (M-H) algorithm for a given proposal matrix. The second algorithm generates, offline, an ON-OFF policy that also ensures the safety constraints together with the ergodicity and motion constraints. The incorporation of the safety constraints is enabled by our recent result that convexifies the Markov chain synthesis with these constraints.**

I. Introduction

This paper introduces a probabilistic density control method applicable to a swarm of autonomous ON-OFF agents. The proposed method produces commands for each agent to determine its own trajectory such that the overall swarm density converges to a desired distribution in the configuration space. The main novelty is that the control law is probabilistic in nature, and ensures the desired spatial probability density distribution for the swarm rather than the exact desired paths of the individual agents. The formulation of this problem and preliminary control laws for swarms with fully mobile agents appeared in [1], [2], [13]. The current paper focuses on an extension of the core idea to a swarm of partially controlled "ON-OFF" agents. This problem is inherently more challenging since the agents have very limited mobility. More precisely, they have control over a binary decision variable, which determines whether they are "ON" or "OFF". If they are ON, it means that they move based on the environmental forces acting on them, such as, winds, sea currents, gravity, and so on. For example, a swarm of autonomous sailboats, where being OFF means the agent does not have its sail deployed, and being ON means that the sail is deployed. Once the agent is ON, it is assumed that there is an existing force field that induces motion on the agent. The force field is assumed to be probabilistic in nature over the region of operation. In the example of sailboats, the force field is due to the wind, which can have a probabilistic characterization, i.e., the probability distribution of the wind direction and magnitude is given as a function of location. After partitioning the configuration space into a finite number of partitions (generating a discrete state-space), this wind field can be described by a Markov chain that describes the probabilities of transitions between partitions if the agent is ON. The only control available is to accept or reject the motion imposed by the nature, i.e., being ON or OFF.

Existing control methods for distributed systems allocate agent positions ahead of time [32], [35], [34], [20], [16], [5], [31]. Ref. [28] also aims to control the probabilistic distribution of the population over an operating space for large-scale robotic populations with limited mobility. However, the approach is fundamentally different from ours since it describes the evolution of the probabilistic states via partial differential equations without considering the motion or safety constraints. Various different schemes applicable to swarm guidance have appeared in the literature [6], [18], [30], [21], [26], [25], [10], [29]. A recent paper [9] uses Markov chains for "swarm self-organization", and appears similar to this paper's usage of Markov chains for swarm control. However, [9] controls the swarm using a probabilistic "disablement" approach based on ideas taken from [22]. Nevertheless, we acknowledge that the basic idea of using Markov chains for controlling swarms is also contained in recent papers.

Our main contribution is extending the Markov chain based swarm density control to decentralized ON-OFF agents, which has several difficulties. First there are inherent motion constraints embedded in the natural force field, which limits the control performance. We cannot determine the transitional probabilities when an agent is ON, we can only accept or reject the motion. In this sense, the proposed approach has connections with the celebrated Metropolis-Hastings algorithm [15], [7], [23] synthesizing the desired

Authors are with The University of Texas at Austin, Texas, USA. Senior Member, IEEE, behcet@austin.utexas.edu.

978-1-4799-8686-6/$31.00 ©2015 AACC

Markov matrices for density control. More precisely, under some assumptions, the design of ON-OFF policy becomes equivalent to designing the *acceptance matrix* in the M-H algorithm. Indeed we prove that, in the absence of the safety constraints and when reversibility of the chain is enforced, the ON-OFF policy must have the form of the general acceptance matrix in M-H algorithm. When we have safety constraints, the resulting ON-OFF policy can be seen as a new M-H algorithm with the safety constraints. It is a new M-H algorithm because the classical algorithm cannot ensure the safety constraints. It is also important to note that the ON-OFF policies synthesized can be implemented in a completely decentralized manner. Our recent results on the convex optimization based synthesis of safety constrained Markov chains [12], [3] now allow us to impose these difficult constraints. The main point of imposing safety constraints is to mitigate the probability of conflicts/collisions between agents. Clearly swarm density control produces high level commands and there must be low level motion planning to eliminate conflicts/collisions. Safety constraints are aimed to aid the lower level motion planning to avoid collisions by providing proper higher level motion commands, i.e., they simplify the job of the lower-level motion planning. Reactive collision avoidance strategies using model predictive control [4] at lower level of control, as appeared in [33] in the context of collaborating spacecraft, and connections to high level density control will be the subject of a future paper.

The paper is organized as follows: Section II summarizes the formulation of the density control problem for ON-OFF agents, this section also describes the ergodicity, safety and motion constraints; Section III presents an analytical solution for the acceptance matrix without safety constraints; Section IV formulates the safety constraints as LMI's on the Markov matrix; Section V proposes two methods for the synthesis of ON-OFF policy: Online synthesis without safety constraints and offline synthesis with safety constraints and Section VI has an illustrative example which uses the Markov matrix synthesized offline with the safety constraints.

Notation

The following is a partial list of notation used: $\mathbf{0}$ is the zero matrix of appropriate dimensions; e_i is a vector of appropriate dimension with its ith entry $+1$ and its other entries zeros; $x[i] = \mathrm{e}_i^T x$ for any $x \in \mathbb{R}^n$ and $A[i,j] = \mathrm{e}_i^T A \mathrm{e}_j$ for any $A \in \mathbb{R}^{n\times m}$; $Q = Q^T \succ (\succeq) 0$ implies that Q is a symmetric positive (semi-)definite matrix; $R > (\geq) H$ implies that $R[i,j] > (\geq) H[i,j]$ for all i,j; $R > (\geq) \mathbf{0}$ implies that R a positive (non-negative) matrix; $v \in \mathbb{P}^n$ is said to be a *probability vector* if $v \geq \mathbf{0}$ and $\mathbf{1}^T v = 1$; matrix $M \in \mathbb{P}^{m\times m}$ is a Markov matrix if $M \geq \mathbf{0}$ and $\mathbf{1}^T M = \mathbf{1}^T$; prob denotes probability of a random variable; $\mathbb{R}^n$ is the n dimensional real vector space; $\emptyset$ denotes the empty set; I is the identity matrix; $\mathbf{1}$ is the matrix of ones with appropriate dimensions; $(v_1, v_2, ..., v_n)$ represents a vector obtained by augmenting vectors $v_1, \ldots, v_n$ such that $(v_1, v_2, ..., v_n) \equiv \begin{bmatrix} v_1^T & v_2^T & \ldots & v_n^T \end{bmatrix}^T$ where v_i have arbitrary dimensions; $\mathrm{diag}(A) = (A[1,1], \ldots, A[n,n])$ for matrix A; $\otimes$ denotes the Kronecker product; $\odot$ represents the Hadamard (Schur) product; $\mathtt{i}(A)$ is the indicator matrix for any matrix A, whose entries are given by $\mathtt{i}(A)[i,j] = 1$ if $A[i,j] \neq 0$ and $\mathtt{i}(A)[i,j] = 0$ otherwise. $\oslash$ denotes element-wise matrix division defined for non-negative matrices by $A = B \oslash C$ means that $A[i,j] = B[i,j]/C[i,j]$ if $C[i,j] > 0$ and $A[i,j] = 0$ if $B[i,j] = C[i,j] = 0$. So $\oslash$ is well-defined if $B[i,j] = 0$ when $C[i,j] = 0$. Further $A \oslash A = \mathtt{i}(A)$, $A = B \oslash C$ implies that $C = B \oslash A$ and $B = A \odot C$.

II. Density Control Problem for ON-OFF Agents

This section summarizes the formulation of the density control problem for swarm of autonomous ON-OFF agents. The swarm agents are distributed over the configuration space $\mathcal{R}$, which is is partitioned into m disjoint subregions R_i, $i = 1, \ldots, m$, such that $\mathcal{R} = \bigcup_{i=1}^{m} R_i$, and $R_i \cap R_j = \emptyset$ for $i \neq j$. The subregions R_i are referred to as *bins*. This partition is used to define a discrete density distribution vector $x(t) \in \mathbb{P}^m$ where $t \in I\!N_+$ denotes a time instant index and

$$x[i](t) := \mathrm{prob}(r(t) \in R_i) \tag{1}$$

where $r(t)$ is the state vector of an agent at time index t. We refer to $x(t)$ as the *swarm density distribution* or the *swarm distribution*. Similarly we also define the probability of finding kth agent in ith bin as follows

$$x_k[i](t) := \mathrm{prob}(r_k(t) \in R_i), \quad i = 1, ..., m, \; k = 1, ..., N. \tag{2}$$

$x_k(t)$ and $x(t)$ are not necessarily the same in general. As it will become apparent, since each agent acts independently of the other agents by using the same decision-making policy, condition (1) holds for N separate events and $x(t) = x_k(t)$ for all time and $k = 1, ..., N$.

The swarm density distribution evolves according to the following Markov chain when all agents are ON,

$$x(t+1) = G x(t) \tag{3}$$

where $G \in \mathbb{P}^{m\times m}$ defines the probability of state transitions

$$G[i,j](t) := \mathrm{prob}(r_k(t+1) \in R_i | r_k(t) \in R_j, \mathrm{ON}), \; k = 1, ..., N. \tag{4}$$

Since each agent's state evolves independently based on the same Markov matrix G, the uncontrolled swarm density evolves according to (3) when all the agents are ON.

Next the density control problem for ON-OFF agents can be described as follows. At any time instance, each agent can measure its current state $r_k(t)$ and predict its next state $r_k(t+1)$ if it were ON at that moment. In other words, we only know G matrix in advance, but not the particular outcome at a given time instance if the agent is ON in a given bin, which becomes known during runtime. Hence, the agent can predict what will its next bin be if it is ON in runtime. Clearly $r_k(t+1) = r_k(t)$ if the agent is OFF. Given this information, each agent independently decides whether it should be ON or OFF. We consider a probabilistic ON-OFF decision policy defined by a matrix $K \in \mathbb{R}^{m\times m}$ where

$$\mathbf{0} \leq K \leq \mathbf{1}\mathbf{1}^T, \quad \mathrm{diag}(K) = \mathbf{1}, \tag{5}$$

where $K[i,j]$ is the probability of being ON if the agent would make a state transition from jth to ith bin when it is ON, i.e., the acceptance probability of the environmentally induced motion. Letting M be the effective Markov matrix when the ON-OFF policy is active, then the probability of the transition from jth to ith bin, $M[i,j]$, is given as follows: For $i,j = 1,...,m$,

$$M[i,j] = G[i,j]K[i,j], \quad i \neq j,$$
$$M[i,i] = 1 - \sum_{k \neq i}^{n} G[i,k]K[i,k].$$

In more compact notation, the above relationship is equivalent to

$$M = G \odot K + \operatorname{diag}(\mathbf{1}^T - \mathbf{1}^T(G \odot K)). \tag{6}$$

It is important that, for given G, the matrix $M \in \mathbb{P}^{m\times m}$ depends linearly on the matrix K.

Lemma 1: M in (6) satisfies that $M \in \mathbb{P}^{m\times m}$.

Proof: • $M \geq 0$: Letting $S := G \odot K$, since $0 \leq G, K \leq \mathbf{1}\mathbf{1}^T$, we have $\mathbf{1}^T S = \mathbf{1}^T(G \odot K) \leq \mathbf{1}^T G = \mathbf{1}^T$. Hence $\mathbf{1}^T - \mathbf{1}^T S \geq \mathbf{0}$, which implies that $M = S + \operatorname{diag}(\mathbf{1}^T - \mathbf{1}^T S) \geq \mathbf{0}$.
• $\mathbf{1}^T M = \mathbf{1}^T$: Let $\xi^T := \mathbf{1}^T G \odot K$. Then $\mathbf{1}^T M = \xi^T + \mathbf{1}^T \operatorname{diag}(\mathbf{1}^T - \xi^T) = \xi^T + \mathbf{1}^T(I - \operatorname{diag}(\xi)) = \xi^T + \mathbf{1}^T - \mathbf{1}^T \operatorname{diag}(\xi) = \xi^T + \mathbf{1}^T - \xi^T = \mathbf{1}^T$. ■

The following equation describes the transition probabilities when the ON-OFF decision policy is active: For $i,j = 1,...,m$,

$$M[i,j] = \operatorname{prob}(r_k(t+1) \in R_i | r_k(t) \in R_j), \quad k = 1,...,N. \tag{7}$$

Since all agents propagate their states based on matrix M, we have

$$x(t+1) = Mx(t), \quad t = 0,1,\ldots \tag{8}$$

With this background, the following algorithm describes how the ON-OFF policy given by the matrix K is used to control the probabilistic density distribution of the swarm.

ON-OFF Policy for Swarm Density Control

1) Each agent determines its current bin, $r_k(t) \in R_j$.
2) Each agent determines the next bin, given that it is ON, $r_k(t+1) \in R_j$.
3) Each agent generates a random number z_k that is uniformly distributed in $[0,1]$.
4) The agent choses to be ON if $z_k \in [0, K[i,j]]$, and OFF otherwise.

Given the density evolution (8), when the ON-OFF policy is active, the design problem is to synthesize the matrix M by a proper choice of the matrix K. The resulting matrix M and the corresponding Markov chains for all possible $x(0) \in \mathbb{P}^m$ must satisfy the following specifications.

a) Ergodicity: The agent distribution $x(t)$ must converge as close to a desired distribution $v_d \in \mathbb{P}^m$ as possible in time. Since M is a Markov matrix, a column stochastic matrix, we would impose this requirement by ensuring that there exists some $v \in \mathbb{P}^m$ such that v is as close to v_d as possible via the choice of K,

$$Mv = v \quad \text{and} \quad \lim_{t\to\infty} x(t) = v \quad \forall x(0) \in \mathbb{P}^m. \tag{9}$$

b) Safety Constraints: We consider two types of density safety constraints: (i) Density upper bound constraints; (ii) Density rate constraints. Density upper bound constraint ensures that the density of each bin stays below a prescribed value, that is,

$$x(t) \leq d, \quad t = 1,2,\ldots \tag{10}$$

where $\mathbf{0} \leq d \leq \mathbf{1}$ defines the density upper bounds in each bin and it is assumed that $x(0) \leq d$. This constraint can be used to limit the number of agents in each bin during the transitions to v, hence it mitigates probability of collisions/conflicts. The density rate constraint is used to avoid excessive flow in and out of the bins, hence to help mitigate the collisions/conflicts,

$$-q \leq x(t+1) - x(t) \leq q, \quad t = 0,1,\ldots, \tag{11}$$

where $q \geq 0$ bounds the flow rate.

c) Motion Constraints: In our earlier papers [12], [13], we impose constraints on the physically realizable state transitions by specifying some entries of the matrix M as zeros. In the case of ON-OFF agents, these constraints are automatically ensured by the matrix G: If a state transition is simply not possible naturally, then the corresponding entry in the matrix G is zero, which implies that it is also zero in the matrix M, i.e., $M[i,j] = 0$ if $G[i,j] = 0$. If additional transitions are also needed to be eliminated, we can impose additional constraints beyond the ones that are naturally imposed by matrix G.

III. An Analytical Solution for K Without Safety Constraints

This section introduces an analytical synthesis method for the matrix K to make connections with the celebrated Metropolis-Hastings (M-H) algorithm that is used extensively in Monte Carlo Markov Chain (MCMC) field [7], [27], [14]. To this end, we impose an additional condition on the Markov chain, called the *reversibility* of the stationary distribution v:

$$M[i,j]v[j] = M[j,i]v[i] \qquad \forall i,j = 1,\ldots,m.$$

The reversibility condition is also known as *detailed balance* condition and having reversibility of v implies the stationarity of v [19]. But having v the stationary distribution of M does not necessarily imply its reversibility. So reversibility is a stronger condition, and we impose this condition to facilitate an analytical approach to synthesize the matrix K (hence matrix M). The following matrix form of the reversibility condition is more conducive for analysis,

$$M\operatorname{diag}(v) = \operatorname{diag}(v)M^T. \tag{12}$$

Note that the expression clearly shows the well known fact that the reversibility implies the stationarity of v as follows

$$\mathbf{1}^T M\operatorname{diag}(v) = \mathbf{1}^T\operatorname{diag}(v)M^T \quad \text{where} \quad \mathbf{1}^T\operatorname{diag}(v) = v^T$$
$$\Rightarrow \quad \mathbf{1}^T\operatorname{diag}(v) = v^T M^T \;\Rightarrow\; v^T = v^T M^T \;\Rightarrow\; Mv = v.$$

The following theorem characterizes all feasible ON-OFF policies, K, with the reversibility constraint satisfied.

Theorem 1: Consider the Markov matrix, M, given by (6) where $\mathtt{i}(G\operatorname{diag}(v)) = \mathtt{i}(\operatorname{diag}(v)G^T)$, with $v \in \mathbb{P}^m$, $v > \mathbf{0}$, and $\operatorname{tr}(G) > 0$. A control policy matrix K such that $\mathtt{i}(K) = \mathtt{i}(K^T)$ satisfies the ergodicity constraint (9) with reversibility, $M\operatorname{diag}(v) = \operatorname{diag}(v)M^T$, *if and only if* $\mathtt{i}(G \odot K)$ corresponds to the adjacency matrix of a strongly connected graph and there exists a matrix $L = L^T$ that satisfies the following conditions:

$$\begin{aligned} &\mathbf{0} \le L \le \mathbf{1}\mathbf{1}^T, \quad K = L \odot \min(\mathbf{1}\mathbf{1}^T, R) \quad \text{where} \\ &R := (\operatorname{diag}(v)\,G^T) \oslash (G\operatorname{diag}(v)). \end{aligned} \tag{13}$$

Note that Theorem 1 recovers the M-H algorithm from our point of view, which is proved (see the Appendix) by a novel use of the algebraic operation $\oslash$ on nonnegative matrices.

Theorem 1 can be used to construct a Markov matrix M satisfying the ergodicity constraints by choice of the matrix L, which determines K in (13). For example the choice of $L = \alpha\mathbf{1}\mathbf{1}^T$, where $\alpha \in (0,1)$, results in a feasible M. To see this, note that $\mathtt{i}(G \odot K) = \mathtt{i}(\alpha\min(\mathbf{1}\mathbf{1}^T, R)) = \mathtt{i}(R) = \mathtt{i}(\operatorname{diag}(v)G^T \oslash G\operatorname{diag}(v)) = \mathbf{1}\mathbf{1}^T$, which implies the strong connectivity. This also results in the M-H algorithm with an Alpha-Min acceptance matrix.

IV. Convexification of the Synthesis With Safety Constraints

In this section, the ergodicity and safety constraints will be expressed as convex constraints on the Markov matrix M. Since the matrix M depends linearly on the ON-OFF policy matrix K as in (6), this will imply that the constraints will be convex constraints on K, which is our design variable.

The ergodicity constraint requires that the condition (9) is satisfied. In [2], we developed a sufficient condition for ergodicity based on the LMI characterizations of the stability of discrete-time systems in [11], which states that: There exist $P = P^T \succ \mathbf{0}$, $\lambda \in [0,1)$, and G such that

$$\begin{bmatrix} \lambda^2 P & (M - v\mathbf{1}^T)^T G^T \\ G(M - v\mathbf{1}^T) & G + G^T - P \end{bmatrix} \succeq 0. \tag{14}$$

The above condition is an LMI for prescribed values of λ G, which we typically choose to be $G = \operatorname{diag}{v_d}^{-1}$. A more detailed discussion on the selection of G and λ can be found in [3]. It is shown that λ denotes the exponential decay rate of the error, $e(t) = x(t) - v$. Furthermore, if reversibility is also required, then we can state a necessary and sufficient condition [8] for ergodicity constraint as follows:

$$-\lambda I \le Q^{-1}MQ - qq^T \le \lambda I, \tag{15}$$

where $q = (v_1^{1/2}, \ldots, v_m^{1/2})$ and $Q = \operatorname{diag}(q)$.

The safety constraints are ensured by the following theorem, which gives necessary and sufficient conditions for safety as linear inequalities on M (see [13], [3] for a proof).

Theorem 2: Consider the Markov chain $M \in \mathbb{P}^{m\times m}$ given by (8).
(i) $x(t) \le d$, $t = 1, 2, \ldots$, when $x(0) \le d$, i.e. the density upper bound constraint holds, *if and only if* the following condition holds:
There exist $S \in \mathbb{R}^{m\times m}$ and $y \in \mathbb{R}^m$ such that

$$\begin{aligned} &S \ge \mathbf{0}, \quad M + S + y\mathbf{1}^T \ge \mathbf{0}, \\ &y + d \ge \left(M + S + y\mathbf{1}^T\right) d. \end{aligned} \tag{16}$$

(ii) Suppose that $x(t) \le \mathrm{h}$, $t = 1, 2, \ldots$, where $\mathbf{0} \le \mathrm{h} \le \mathbf{1}$. Then $-q \le x(t+1) - x(t) \le q$, $\forall t \ge 0$, holds (the bound on the rate of change of density holds) *if and only if* the following set of inequalities (for $k = 1, 2$) hold: There exist $S_k \in \mathbb{R}^{m\times m}$ and $y_k \in \mathbb{R}^m$, such that

$$\left.\begin{aligned} &S_k \ge \mathbf{0}, \quad (-1)^k(M - I) + S_k + y_k\mathbf{1}^T \ge \mathbf{0}, \\ &y_k + q \ge \left((-1)^k(M - I) + S_k + y_k\mathbf{1}^T\right)\mathrm{h} \end{aligned}\right. \quad k = 1, 2. \tag{17}$$

Note that, since all the unknowns appear linearly in the above inequalities, they are solvable for M; hence for K since M is linear in K.

V. Synthesis of ON-OFF Policy

In this section we give two synthesis methods for K. The first method, M-H like algorithm, is an online method to generate K onboard in realtime, when the safety constraint is not imposed. The second method is an offline method to compute K with the safety constraints.

A. Online Synthesis Without Safety Constraints

We use Theorem 1, which implies the following method of synthesis. The resulting method can easily be implemented online: Choose any $\mathbf{0} \le L = L^T \le \mathbf{1}\mathbf{1}^T$, then

$$K = L \odot \min(\mathbf{1}\mathbf{1}^T, R), \quad R = (\operatorname{diag} v_d G^T) \oslash (G\operatorname{diag}(v_d)), \tag{18}$$

which results in a $G \odot K$ that represents a strongly connected graph. As also noted earlier, a simple choice of $L = \alpha\mathbf{1}\mathbf{1}^T$ where $\alpha \in (0,1)$ accomplishes this goal and the resulting Markov matrix M satisfies the ergodicity constraints.

B. Offline Synthesis With Safety Constraints

Next we combine the results in Section IV to propose the following LMI optimization problem for the synthesis of K for prescribed matrix G and convergence rate $\lambda \in [0,1)$:

$$\begin{aligned} \min_{K,P,S,y,v} \quad & \|v - v_d\|_1 \qquad \text{s.t} \\ & M = G \odot K + \operatorname{diag}(\mathbf{1}^T - \mathbf{1}^T(G \odot K)) \\ & v \ge \mathbf{0}, \quad \mathbf{1}^T v = 1 \\ & S \ge \mathbf{0}, \\ & M + S + y\mathbf{1}^T \ge \mathbf{0}, \\ & y + d \ge \left(M + S + y\mathbf{1}^T\right) d. \\ & k = 1, 2: \quad S_k \ge \mathbf{0}, \\ & (-1)^k(M - I) + S_k + y_k\mathbf{1}^T \ge \mathbf{0}, \\ & y_k + q \ge \left((-1)^k(M - I) + S_k + y_k\mathbf{1}^T\right)\mathrm{h} \\ & \begin{bmatrix} \lambda^2 P & (M - v\mathbf{1}^T)^T G^T \\ G(M - v\mathbf{1}^T) & G + G^T - P \end{bmatrix} \succeq 0 \\ & P = P^T \succ 0, \quad \mathbf{0} \le K \le \mathbf{1}\mathbf{1}^T \end{aligned} \tag{19}$$

Note that a line search can be performed for λ to find its smallest feasible value, i.e., the fastest convergence rate.

VI. Numerical Examples

This section presents an example which demonstrates the PDC algorithm for ON-OFF agents. In this problem, $N = 3000$ agents are assumed to be distributed on a region which is partitioned to 8 equally sized rectangular bins (see Fig 1). It is also assumed that, when all agents are ON, the swarm density evolves according to the following G matrix.

$$\begin{bmatrix} \frac{24}{125} & \frac{57}{2000} & \frac{187}{5000} & 0 & 0 & 0 & 0 & 0 \\ \frac{943}{2500} & \frac{137}{5000} & \frac{973}{2500} & \frac{2763}{10000} & \frac{2763}{10000} & 0 & 0 & 0 \\ \frac{1077}{2500} & \frac{1957}{5000} & \frac{2867}{5000} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{2763}{10000} & 0 & \frac{387}{5000} & \frac{387}{5000} & 0 & \frac{697}{5000} & 0 \\ 0 & \frac{691}{2500} & 0 & \frac{387}{5000} & \frac{387}{5000} & 0 & \frac{697}{5000} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{308}{625} & 0 & \frac{317}{625} \\ 0 & 0 & 0 & \frac{5689}{10000} & \frac{5689}{10000} & \frac{93}{1000} & \frac{57}{625} & \frac{129}{2000} \\ 0 & 0 & 0 & 0 & 0 & \frac{2071}{5000} & \frac{63}{100} & \frac{4283}{10000} \end{bmatrix}$$

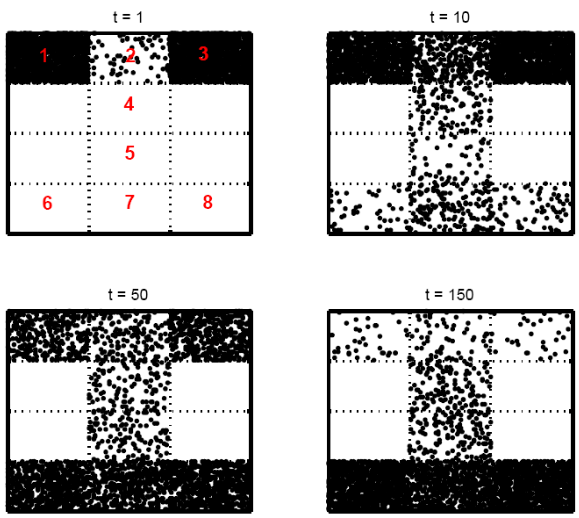

Fig. 1. Evolution of the density distribution with ON-OFF decision policy

This policy leads to the following final distribution:

$$v_n = [0.05 \;\; 0.1 \;\; 0.05 \;\; 0.1 \;\; 0.1 \;\; 0.275 \;\; 0.05 \;\; 0.275]^T.$$

The aim is to design an ON-OFF decision policy, K, such that the resulting Markov chain, M, will lead the swarm to converge to a particular final distribution while satisfying the desired safety upper bound constraints. Simulation parameters are set as follows:

$$x(0) = [0.5 \;\; 0 \;\; 0.5 \;\; 0 \;\; 0 \;\; 0 \;\; 0 \;\; 0]^T, \quad \lambda = 0.975,$$
$$v_d = [0.005 \;\; 0.02 \;\; 0.005 \;\; 0.04 \;\; 0.05 \;\; 0.34 \;\; 0.2 \;\; 0.34]^T,$$
$$d = [1 \;\; 0.15 \;\; 1 \;\; 0.12 \;\; 0.12 \;\; 1 \;\; 0.4 \;\; 1]^T.$$

Since the bins at the corners behave like accumulation points in the initial and final conditions, the safety upper bound constraints are not imposed for these bins, i.e. the corresponding entries of the d vector are set as 1. With the given parameters, the optimization problem given in (19) generates the following K matrix, with YALMIP and SDPT3 [24], [36]:

$$\begin{bmatrix} 1.0000 & 0.4037 & 0.9283 & 0.5000 & 0.5000 & 0.5000 & 0.5000 & 0.5000 \\ 0.0778 & 1.0000 & 0.0776 & 0.2276 & 0.1544 & 0.5000 & 0.5000 & 0.5000 \\ 0.1193 & 0.0086 & 1.0000 & 0.5000 & 0.5000 & 0.5000 & 0.5000 & 0.5000 \\ 0.5000 & 0.6699 & 0.5000 & 1.0000 & 0.6298 & 0.5000 & 0.7507 & 0.5000 \\ 0.5000 & 0.1711 & 0.5000 & 0.7467 & 1.0000 & 0.5000 & 0.9575 & 0.5000 \\ 0.5000 & 0.5000 & 0.5000 & 0.5000 & 0.5000 & 1.0000 & 0.5000 & 0.8349 \\ 0.5000 & 0.5000 & 0.5000 & 0.9773 & 0.8924 & 0.9400 & 1.0000 & 0.9324 \\ 0.5000 & 0.5000 & 0.5000 & 0.5000 & 0.5000 & 0.8112 & 0.3982 & 1.0000 \end{bmatrix}$$

Simulations are done for the cases i) when all agents are ON; ii) with ON-OFF decision policy; which helps to observe effect of ON-OFF decision policy. Figure (2) shows the time histories of the overall density distribution for these two cases. The density goes above the desired upper bound for the bins 2, 4 and 5 when all the agents are ON, i.e. the density evolves according to (3). By using binary control actions, we are able to modify the final distribution and ensure that the density does not go beyond the prescribed upper limit at the expense of convergence rate.

The snapshots of the overall distribution taken at selected time instances for the case with ON-OFF decision policy are shown in Figure (1) which also shows the bin numbers.

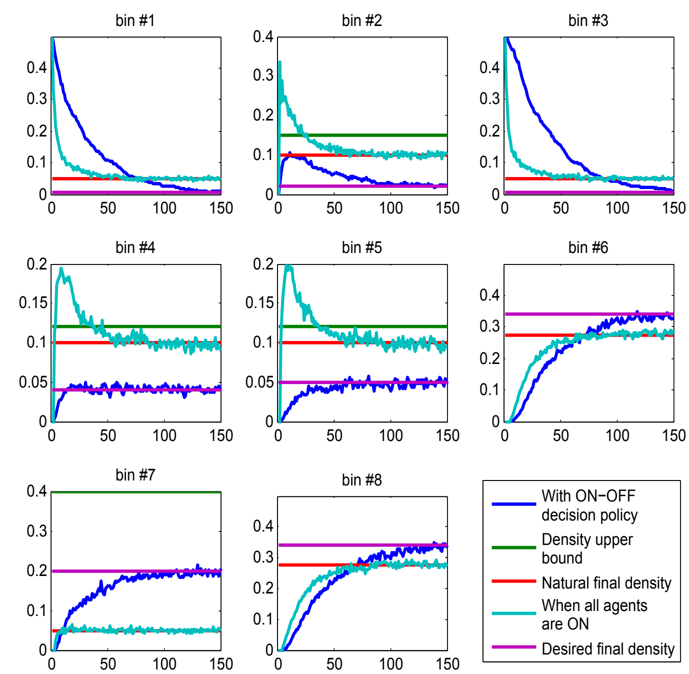

Fig. 2. Time history of the density of each bin

VII. Conclusions

This paper develops a Markov chain based probabilistic density control algorithm for swarm of ON-OFF agents. Here, ON-OFF agents are assumed to have limited mobility, i.e they can only have two actions: accepting the environmentally induced motion,“ON”,’ or having no motion,“OFF”’. The paper’s aim is to synthesize the acceptance matrix, which describes the acceptance probability of the environmentally induced motion such that the resulting Markov chain guides the swarm to a prescribed final distribution while satisfying the mission constraints. Since each agent makes statistically independent decisions according to the given control policy, coordination of swarm is achieved in a decentralized manner. In this study, we first present the analytical solution for the acceptance matrix without the safety constraints, which is inspired by the Metropolis-Hastings algorithm. Then, based on our earlier results, we present an offline synthesis of the acceptance matrix with the safety constraints by using LMI's. The safety constraints are imposed to mitigate any conflicts or collisions between agents that can be caused by excessive agent density and flow rate in a given region. The resulting density control algorithm is illustrated with a numerical example.

ACKNOWLEDGMENTS

The authors gratefully acknowledge David S. Bayard of Jet Propulsion Laboratory for his deep insights and comments. This research was supported in part by Defense Advanced Research Projects Agency (DARPA) Grant No. D14AP00084 and National Science Foundation Grant No. CNS-1446578

VIII. APPENDIX: PROOF OF THEOREM 1

This section presents the proof of Theorem 1 which proposes an analytical solution for the acceptance matrix. The following proof utilizes an element-wise division on positive vectors, which presents is not standard. We believe that this notation (and approach) presents an alternative to the standard notation Markov chain literature and it can be insightful in constructing Markov matrices analytically. Here, $\odot$ represents the Hadamard (Schur) product and $\oslash$ denotes element-wise matrix division defined for non-negative matrices by $A = B \oslash C$ means that $A[i,j] = B[i,j]/C[i,j]$ if $C[i,j] > 0$ and $A[i,j] = 0$ if $B[i,j] = C[i,j] = 0$. So $\oslash$ is well-defined if $B[i,j] = 0$ when $C[i,j] = 0$. Further $A \oslash A = \mathtt{i}(A)$, $A = B \oslash C$ implies that $C = B \oslash A$ and $B = A \odot C$.

Proof: • We first establish the equivalence of (13) to the reversibility. Observe that $M\mathrm{diag}(v) = \mathrm{diag}(v)M^T$ is equivalent to having

$$(G \odot K)\mathrm{diag}(v) = \mathrm{diag}(v)(G^T \odot K^T). \tag{20}$$

This can be established by observing that $D\mathrm{diag}(v) = \mathrm{diag}(v)D$, for $D = \mathrm{diag}(\mathbf{1}^T - \mathbf{1}^T(G \odot K))$, since D is diagonal.

Let $U := K \oslash K^T$, which is well-defined since $\mathtt{i}(K) = \mathtt{i}(K^T)$. The equality (20) implies that

$$\begin{aligned} &(G \odot K^T \odot U)\mathrm{diag}(v) - \mathrm{diag}(v)(G^T \odot K^T) = \mathbf{0} \\ &\Rightarrow (G\mathrm{diag}(v)) \odot U \odot K^T - (\mathrm{diag}(v)G^T) \odot K^T = \mathbf{0} \\ &\Rightarrow \left[(G\mathrm{diag}(v)) \odot U - \mathrm{diag}(v)G^T\right] \odot K^T = \mathbf{0}, \end{aligned} \tag{21}$$

which implies that the reversibility can equivalently be expressed by the equation (21). Noting that $K \geq \mathbf{0}$ and $\mathtt{i}(K) = \mathtt{i}(K^T)$, the equation (21) can be equivalently expressed as

$$\underbrace{\left[(G\mathrm{diag}(v)) \odot U - \mathrm{diag}(v)G^T\right]}_{:=\Phi} \odot \mathtt{i}(K) = \mathbf{0}. \tag{22}$$

The above equality is equivalent to having $\Phi[i,j] = 0$ for $K[i,j] > 0$. Dividing (in the matrix sense) both sides by $G\mathrm{diag}(v)$, which is a well-defined operation since it does not lead to a strictly positive number being divided by zero,

$$\big(U - (\mathrm{diag}(v)\,G^T) \oslash (G\,\mathrm{diag}(v))\big) \odot \mathtt{i}(K) = \mathbf{0}.$$

This equation implies the following condition, which is an equivalent characterization of reversibility in this case,

$$(U - R) \odot \mathtt{i}(K) = \mathbf{0} \quad \text{where} \quad U = K \oslash K^T. \tag{23}$$

The equation (23) can be expanded to have $U \odot \mathtt{i}(K) = R \odot \mathtt{i}(K)$. Since $U = K \oslash K^T$ with $\mathtt{i}(K) = \mathtt{i}(K^T)$, $U \odot \mathtt{i}(K) = U$, which implies that the equation (23) is equivalent to

$$K \oslash K^T = \mathtt{i}(K) \odot R. \tag{24}$$

Now, under the assumptions of the theorem, there is a new equivalent condition to reversibility given by the equation (24). It implies that, since $\mathtt{i}(K)$ is symmetric and $\mathbf{0} \leq K \leq \mathbf{1}\mathbf{1}^T$,

$$\begin{aligned} &K = R \odot \mathtt{i}(K) \odot K^T = R \odot \mathtt{i}(K^T) \odot K^T = R \odot K^T \\ &\Rightarrow\ K \leq \min(\mathbf{1}\mathbf{1}^T, R), \end{aligned}$$

which then implies that there exists some $\mathbf{0} \leq L \leq \mathbf{1}\mathbf{1}^T$ such that $\mathtt{i}(L) = \mathtt{i}(K)$

$$K = L \odot \min(\mathbf{1}\mathbf{1}^T, R) \ \Rightarrow\ K^T = L^T \odot \min(\mathbf{1}\mathbf{1}^T, R^T).$$

Then we have

$$\begin{aligned} K \oslash K^T &= \big(L \odot \min(\mathbf{1}\mathbf{1}^T, R)\big) \oslash \big(L^T \odot \min(\mathbf{1}\mathbf{1}^T, R^T)\big) \\ &= \big(L \odot \min(\mathbf{1}\mathbf{1}^T, R)\big) \oslash \big(L^T \odot \min(\mathbf{1}\mathbf{1}^T, \mathtt{i}(R) \oslash R)\big), \end{aligned}$$

where $R^T = \mathtt{i}(R) \oslash R$ can be derived as follows:

$$\begin{aligned} &R^T = (G\mathrm{diag}(v)) \oslash (\mathrm{diag}(v)G^T) \Rightarrow\ R^T \odot R \\ &= (G\mathrm{diag}(v)) \oslash (\mathrm{diag}(v)G^T) \odot (\mathrm{diag}(v)G^T) \oslash (G\mathrm{diag}(v)) \\ &= (G\mathrm{diag}(v)) \oslash (G\mathrm{diag}(v)) \odot (\mathrm{diag}(v)G^T) \oslash (\mathrm{diag}(v)G^T) \\ &= \mathtt{i}(\mathrm{diag}(v)G^T) \odot \mathtt{i}(G\mathrm{diag}(v)) = \mathtt{i}(R), \end{aligned}$$

which then implies $R^T = \mathtt{i}(R) \oslash R$. Now, since $\mathtt{i}(L) = \mathtt{i}(K)$ (which is symmetric)

$$K \oslash K^T = (L \oslash L^T) \odot \underbrace{\big(\min(\mathbf{1}\mathbf{1}^T, R) \oslash \min(\mathbf{1}\mathbf{1}^T, \mathtt{i}(R) \oslash R)\big)}_{:=\psi}.$$

Clearly ψ is well-defined since $\mathtt{i}(\min(\mathbf{1}\mathbf{1}^T, R)) = \mathtt{i}(\min(\mathbf{1}\mathbf{1}^T, \mathtt{i}(R) \oslash R))$, further

$$\psi[i,j] = \begin{cases} 1/(1/R[i,j]) & \text{if } R[i,j] \geq 1 \\ R[i,j]/1 & \text{if } R[i,j] \in (0,1) \\ 0 & \text{if } R[i,j] = 0 \end{cases} \quad i,j = 1,\ldots,m,$$

which implies that $\psi = R$. This then implies that

$$K \oslash K^T = (L \oslash L^T) \odot R.$$

Using the equation (24), the above implies that

$$(L \oslash L^T) \odot R = \mathtt{i}(K) \odot R \Rightarrow L \oslash L^T = \mathtt{i}(K) \odot R \oslash R = \mathtt{i}(K) \odot \mathtt{i}(R)$$

where $\mathtt{i}(K) = \mathtt{i}(L)$ and $\mathtt{i}(R)$ are symmetric. Since $K \leq \min(\mathbf{1}\mathbf{1}^T, R)$, $R[i,j] = 0$ implies that $K[i,j] = 0$ and hence $\mathtt{i}(K) \odot \mathtt{i}(R) = \mathtt{i}(K) = \mathtt{i}(L)$, which is symmetric. Consequently,

$$L \oslash L^T = \mathtt{i}(L) \ \Rightarrow\ L \oslash \mathtt{i}(L) = L^T \ \Rightarrow\ L = L^T.$$

This concludes the necessity of having symmetric L satisfying (13).

Next we show the sufficiency. Consider a symmetric L as in (13). To show sufficiency, it is enough to show that (24) will be satisfied.

$$K \oslash K^T = \left(L \odot \min(\mathbf{1}\mathbf{1}^T, R)\right) \oslash \left(L^T \odot \min(\mathbf{1}\mathbf{1}^T, R^T)\right).$$

Then, as done above, we can show that $K \oslash K^T = (L \oslash L^T) \odot R$ and since $L = L^T$, $L \oslash L^T = \mathtt{i}(L)$, which concludes the sufficiency proof. This then concludes the proof.

• Next we present the equivalence of the ergodicity with the strong connectivity of the graph of $\mathtt{i}(G \odot K)$. First observe that the ergodicity constraint is equivalent to the primitivity of the matrix M when $v > \mathbf{0}$ (see [1], [2]).

Now we will show that, when $\mathrm{tr}(G) > 0$, the primitivity of M is equivalent to the strong connectivity of $G \odot K$. Since $\mathrm{tr}(G) > 0$, there exists $G[j,j] > 0$ for some index j. Note that $\sum_{i=1}^{m} M[i,j] = M[j,j] + \sum_{i \neq j} M[i,j]$ where $\sum_{i \neq j} M[i,j] \leq \sum_{i \neq j} G[i,j] < 1$ since $G[j,j] > 1$ and $G \in \mathbb{P}^m$. This implies that $\sum_{i \neq j} M[i,j] < 1$, and hence $M[j,j] > 0$, which then implies that $\mathrm{tr}(M) > 0$. Since $\mathrm{tr}(M) > 0$, and since the strong connectivity of the graph of M is same as the graph of $G \odot K$, the strong connectivity of $G \odot K$ is equivalent to the primitivity of M [17]. This concludes the proof. ∎

References

[1] B. Açıkmeşe and D. S. Bayard. A markov chain approach to probabilistic swarm guidance. *American Control Conference, Montreal, Canada*, pages 6300–6307, 2012.

[2] B. Açıkmeşe and D. S. Bayard. Markov chain approach to probabilistic guidance for swarms of autonomous agents. *In Press, Asian Journal of Control*, 2015.

[3] B. Açıkmeşe, N. Demir, B. Açıkmeşe, and M. Harris. Convex necessary and sufficient conditions for density safety constraints in markov chain synthesis. *Accepted to IEEE Trans. on Automatic Control*, 2015.

[4] B. Acikmese, J. M. Carson, and D.S. Bayard. A robust model predictive control algorithm for incrementally-conic uncertain/nonlinear systems. *International Journal of Robust and Nonlinear Control*, 21:563–590, 2011. Published online in Wiley InterScience (www.interscience.wiley.com), July 2010.

[5] B Açıkmeşe, Fred Y Hadaegh, Daniel P Scharf, and Scott R Ploen. Formulation and analysis of stability for spacecraft formations. *IET Control Theory & Applications*, 1(2):461–474, 2007.

[6] S. Berman, A. Halasz, M. A. Hsieh, and V. Kumar. Optimized stochastic policies for task allocation in swarms of robots. *IEEE Trans. on Robotics*, 25(4):927–937, 2009.

[7] L. J. Billera and P. Diaconis. A geometric interpretation of the Metropolis-Hastings algorithm. *Statistical Science*, 16(4):335–339, 2001.

[8] S. Boyd, P. Diaconis, P. Parillo, and L. Xiao. Fastest mixing Markov Chain on graphs with symmetries. *SIAM Jrnl. of Optimization*, 20(2):792–819, 2009.

[9] I. Chattopadhyay and A. Ray. Supervised self-organization of homogeneous swarms using ergodic projections of markov chains. *IEEE Trans. on Systems, Man, and Cybernetics*, 39(6):1505–1515, 2009.

[10] J. Cortés, S. Martínez, and F. Bullo. Robust rendezvous for mobile autonomous agents via proximity graphs in arbitrary dimensions. *IEEE Trans. on Automatic Control*, 51(8):1289–1298, 2006.

[11] M.C. de Oliveira, J. Bernussou, and J.C. Geromel. Anew discrete-time robust stability condition. *Systems and Control Letters*, pages 261–265, 1999.

[12] N. Demir, B. Açıkmeşe, and M. Harris. Convex optimization formulation of density upper bound constraints in Markov chain synthesis. *American Control Conference*, pages 483– 488, 2014.

[13] N. Demir, B. Açıkmeşe, and C. Pehlivanturk. Density control for decentralized autonomous agents with conflict avoidance. *IFAC World Congress, South Africa*, pages 11715–11721, 2014.

[14] P. Diaconis. The Markov chain Monte Carlo revolution. *Bulletin of the American Mathematical Society*, 46:179–205, 2009.

[15] P. Diaconis and L. Saloff-Coste. What do we know about the Metropolis algorithm? *Jrnl. of Computer and System Sciences*, 57:20–36, 1998.

[16] F.Y. Hadaegh, B. Açıkmeşe, D.S. Bayard, G. Singh, M. Mandic, S. Chung, and D. Morgan. Guidance and control of formation flying spacecraft: From two to thousands. *Festschrift honoring John L. Junkins, Tech Science Press*, pages 327–371, 2013.

[17] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, 1985.

[18] A. Jadbabaie, G. J. Lin, and A. S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Trans. on Automatic Control*, 48(6):988–1001, 2003.

[19] F. P. Kelly. *Reversibility and Stochastic Networks*. John Wiley & Sons, Inc., 1979.

[20] Y. Kim, M. Mesbahi, and F. Hadaegh. Multiple-spacecraft reconfiguration through collision avoidance, bouncing, and stalemate. *Jrnl. of Optimization Theory and Applications*, 122(2):323–343, 2004.

[21] E. Klavins, R. Ghrist, and D. Lipsky. A grammatical approach to self organizing robotic systems. *IEEE Trans. on Automatic Control*, 51(6):949–962, 2006.

[22] M. Lawford and W. Wonham. Supervisory control of probabilistic discrete event systems. *Proc. 36th Midwest Symp. Circuits Syst.*, pages 327–331, 1993.

[23] J. S. Liu. Metropolized independent sampling with comparisons to rejection sampling and importance sampling. *Statistics and Computing*, 6:113–119, 1996.

[24] J. Löfberg. YALMIP: A toolbox for modeling and optimization in MATLAB. In *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.

[25] S. Martínez, F. Bullo, J. Cortés, and E. Frazzoli. On synchronous robotic networks-part ii: Time complexity of rendezvous and deployment algorithms. *IEEE Trans. on Automatic Control*, 52(12):2214–2226, 2007.

[26] S. Martínez, J. Cortés, and F. Bullo. Motion coordination with distributed information. *IEEE Control System Magazine*, pages 75–88, 2007.

[27] N. Metropolis and S. Ulam. The Monte-Carlo method. *American Statistical Association*, 44:335–341, 1949.

[28] D. Milutinovic and P. Lima. Modeling and optimal centralized control of a large-size robotic population. *IEEE Transactions on Robotics*, 22(6):1280 – 1285, 2006.

[29] D. Morgan, S. Chung, L. Blackmore, B. Acikmese, D. Bayard, and F. Hadaegh. Swarm-keeping strategies for spacecraft under j2 and atmospheric drag perturbations. *Journal of Guidance, Control, and Dynamics*, 35(5):1492–1506, 2012.

[30] R. Olfati-Saber. Flocking for multi-agent dynamic systems: Algorithms and theory. *IEEE Trans. on Automatic Control*, 51(3):410–420, 2006.

[31] J. Ramirez, M. Pavone, E. Frazzoli, and D. W. Miller. Distributed control of spacecraft formations via cyclic pursuit: Theory and experiments. *AIAA Jrnl. of Guidance, Control, and Dynamics*, 33(5):1655–1669, 2010.

[32] A. Richards, T. Schouwenaars, J. How, and E. Feron. Spacecraft trajectory planning with avoidance constraints using mixed-integer linear programming. *AIAA Jrnl. of Guidance, Control, and Dynamics*, 25(4):755–764, 2002.

[33] D. Scharf, B. Açikmese, and F. Ploen, S.and Hadaegh. A direct solution for fuel-optimal reactive collision avoidance of collaborating spacecraft. In *American Control Conference, 2006*, pages 6–pp. IEEE, 2006.

[34] D. P. Scharf, F. Hadaegh, and S. R. Ploen. A survey of spacecraft formation flying guidance and control (part i): guidance. *In Proc. of American Control Conference*, 2003.

[35] M. Tillerson, G. Inalhan, and J. How. Coordination and control of distributed spacecraft systems using convex optimization techniques. *International Jrnl. of Robust Nonlinear Control*, 12(1):207–242, 2002.

[36] R. H. Tutuncu, K. C. Toh, and M. J. Todd. Solving semidefinite-quadratic-linear programs using sdpt3. *Mathematical Programming*, 95(2):189–217, 2003.