Probabilistic Swarm Guidance using Inhomogeneous Markov Chains

Saptarshi Bandyopadhyay Student Member, IEEE, Soon-Jo Chung Senior Member, IEEE, and Fred Y. Hadaegh Fellow, IEEE

Abstract—Probabilistic swarm guidance involves designing a Markov chain so that each autonomous agent or robot determines its own trajectory in a statistically independent manner. The swarm converges to the desired formation and the agents repair the formation even if it is externally damaged. In this paper, we present an inhomogeneous Markov chain approach to probabilistic swarm guidance algorithms for minimizing the number of transitions required for achieving the desired formation and then maintaining it. With the help of communication with neighboring agents, each agent estimates the current swarm distribution and computes the tuning parameter which is the Hellinger distance between the current swarm distribution and the desired formation. We design a family of Markov transition matrices for a desired stationary distribution, where the tuning parameter dictates the number of transitions. We discuss methods for handling motion constraints and prove the convergence and the stability guarantees of the proposed algorithms. Finally, we apply these proposed algorithms for guidance and motion planning of swarms of spacecraft in Earth orbit.

I. INTRODUCTION

Small satellites are well-suited for formation flying missions, where multiple satellites operate together in a cluster or pre-defined geometry to accomplish the task of a single conventionally large satellite. In comparison with large monolithic satellites, small distributed satellites are modular in nature and offer low development cost by enabling rapid manufacturing using commercial-off-the-shelf components. Application of swarms of hundreds to thousands of femtosatellites (100-gram-class satellites) for synthetic aperture applications has been discussed in [1]. In this paper, we introduce an inhomogeneous Markov chain approach to develop probabilistic swarm guidance algorithms for constructing and reconfiguring a multi-agent network comprised of a large number (1000s–10000s) of autonomous agents or spacecraft.

Analogous to fluid mechanics, the traditional view of guidance of multi-agent systems is Lagrangian, as it deals with an indexed collection of agents [2]–[7]. Note that such deterministic algorithm tend to perform poorly in the presence of large number (100s–1000s) of expendable agents and are not robust to external disturbances. In this paper, we adopt an Eulerian view, as we control the swarm density distribution of a large number of index-free agents over disjoint bins [8], [9]. A centralized approach for controlling the swarm distribution, using optimal control of partial differential equations, is discussed in [10]. Distributed control of the swarm density distribution, using region-based shape controllers or attraction-repulsion forcing functions, are discussed in [11], [12].

Instead of allocating positions to agents a priori, the distributed probabilistic guidance algorithm orders each agent to transition using a synthesized Markov chain so that the steady-state distribution corresponds to the desired formation [13]–[16]. Each autonomous agent or robot independently determines its own trajectory without any communication so that the overall swarm converges to a desired formation. Due to this Markovian approach, the resulting algorithm is robust to external disturbances or damages to the formation. The main limitation of probabilistic guidance algorithm using homogeneous Markov chains, where the Markov matrix \( M \) is fixed over time, is that the agents are not allowed to settle down even after the swarm has reached the desired steady-state distribution resulting in significant wastage of control effort (e.g., fuel). This paper develops probabilistic swarm guidance algorithm using inhomogeneous Markov chains (PSG–IMC) to address this limitation and minimize the number of transitions for achieving and maintaining the formation. Our key concept, which was first presented in [17], is to develop time-varying Markov matrices \( M_k \) with \( \lim_{k \to \infty} M_k = I \) to ensure that the agents settle down after the desired formation has been achieved.

It is necessary that each agent communicates with its neighboring agents to estimate the current swarm distribution. Consensus algorithms [2], [18]–[22] have been studied for formation control, sensor networks, and formation flying applications. In this paper, using a decentralized consensus algorithm on probability distributions [23], the agents reach an agreement on the current estimate of the swarm distribution. In this paper, guidance refers to both motion planning and open–loop control that generates a desired trajectory for each agent [24]. The PSG–IMC algorithm generates the trajectory for each agent from its current location to its final location so that the overall swarm converges to the desired formation. Note that inter-agent collisions have been ignored in this paper, but the guidance trajectories obtained from PSG–IMC have been implemented using suitable navigation and control algorithms [25], [26]. As an illustrative example, the guidance of spacecraft in a swarm orbiting Earth has been presented in this paper.

S. Bandyopadhyay and S.-J. Chung are with the Department of Aerospace Engineering and Coordinated Science Laboratory, University of Illinois at Urbana-Champaign (UIUC), Urbana, IL 61801, USA. (email: bandyop2@illinois.edu and sjchung@illinois.edu). F. Y. Hadaegh is with the Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA 91109, USA (email: fred.y.hadaegh@jpl.nasa.gov).

This research was supported in part by AFOSR grant FA95501210193. This research was carried out in part at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. © 2013 California Institute of Technology.
A. Notation

The time index is denoted by a right subscript and the agent index is denoted by a lower-case right superscript. The symbol \( P(\cdot) \) refers to the probability of an event. The graph \( G_k \) represents the directed time-varying communication network topology at the \( k \)th time instant, where all the agents form the set of vertices \( V \) and \( E_k \) is the set of directed edges. Let \( N_k^j \) denote the neighbors of the \( j \)th agent at the \( k \)th time instant from which it receives information, i.e., \( \ell \in N_k^j \) if and only if \( \ell, j \in V_k \). The set of inclusive neighbors of the \( j \)th agent is denoted by \( J_k^j := N_k^j \cup \{ j \} \).

Let \( \mathbb{N} \), \( \mathbb{Z}^+ \), and \( \mathbb{R} \) be the sets of natural numbers (positive integers), nonnegative integers (\( \mathbb{Z}^+ := \{0\} \cup \mathbb{N} \)), and real numbers respectively. The set of all \( m \) by \( n \) matrices over the field of real numbers \( \mathbb{R} \) is denoted by \( \mathbb{R}^{m \times n} \). Let \( \lambda \) and \( \sigma \) represent the eigenvalue and the singular value of a square matrix. For a column vector \( \alpha \), let \( \text{diag}(\alpha) \) represent the diagonal matrix of appropriate size with \( \alpha \) as its diagonal elements. Let \( \text{min}^+ \) refer to the minimum of the positive elements. Let \( 1 = [1, 1, \ldots, 1]^T \), \( 0 \) and \( \phi \) be the ones (column) vector, the identity matrix, the zero matrix of appropriate sizes and the empty set respectively. Finally, \( \| \cdot \|_p \) represents the \( \ell_p \) vector norm while \( \| \cdot \|_{TV} \) represents the total variation norm of measures. The symbols \( |\cdot| \), \( [\cdot] \), and \( \text{sgn}(\cdot) \) denote the absolute value or the cardinality of a set, ceiling function, and signum function respectively.

II. Problem Statement and Overview of PSG–IMC

Let \( \mathcal{R} \subset \mathbb{R}^{nx} \) denote the \( nx \)-dimensional compact physical space over which the swarm is distributed. The region \( \mathcal{R} \) is partitioned into \( n_{\text{cell}} \) disjoint bins represented by \( R[i], i = 1, \ldots, n_{\text{cell}} \) so that \( \bigcup_{i=1}^{n_{\text{cell}}} R[i] = \mathcal{R} \) and \( R[i] \cap R[q] = \phi \), if \( i \neq q \).

Let \( m \in \mathbb{N} \) agents belong to the swarm. Note that we assume that \( m \gg n_{\text{cell}} \), since we control the swarm density over these bins. Let the row vector \( r_{k}^j \) represent the bin in which the \( j \)th agent is actually present at the \( k \)th time instant. If \( r_{k}^j[i] = 1 \), then the \( j \)th agent is inside the \( R[i] \) bin at the \( k \)th time instant; otherwise \( r_{k}^j[i] = 0 \). The current swarm distribution \( \mathcal{F}_k \) is given by the ensemble mean of actual agent positions, i.e., \( \mathcal{F}_k := \frac{1}{m} \sum_{j=1}^{m} r_{k}^j \). Let us now define the desired formation.

Definition 1. (Desired Formation \( \pi \)) Let the desired formation shape be represented by a probability (row) vector \( \pi \in \mathbb{R}^{n_{\text{cell}}} \) over the bins in \( \mathcal{R} \), i.e., \( \pi 1 = 1 \). Note that \( \pi \) can be any arbitrary probability vector, but it is the same for all agents within the swarm. In the presence of motion constraints, \( \pi \) needs to satisfy Assumption 2 discussed in Section VI. For example, \( \pi \) is the shape of the letter “I” in Fig. 1.

Fig. 1. The PSG–IMC independently determines each agent’s trajectory so that the overall swarm converges to the desired formation (here letter “I”), starting from any initial distribution.

The objectives of probabilistic swarm guidance using inhomogeneous Markov chains (PSG–IMC) running onboard each agent are as follows:

1) Determine the agent’s trajectory using a Markov chain, which obeys motion constraints, so that the overall swarm converges to a desired formation \( (\pi) \).

2) Reduce the number of transitions for achieving and maintaining the formation in order to reduce control effort (e.g., fuel).

3) Maintain the swarm distribution and automatically detect and repair damages to the formation.

The key idea of the proposed PSG–IMC is to synthesize inhomogeneous Markov chains for each agent so that each agent can independently determine its trajectory while the swarm distribution converges to the desired formation. The flowchart for the algorithm is shown in Fig. 2 and its pseudo code is given in Algorithm I.

Fig. 2. Flowchart for PSG–IMC with motion constraint describing the key steps for a single agent in a single time step.

Fig. 2. Flowchart for PSG–IMC with motion constraint describing the key steps for a single agent in a single time step.

Step 1 involves determining the bin in which the agent is located. For example, the \( j \)th agent is in the \( R[i] \) bin at the \( k \)th time instant, i.e., \( r_{k}^j[i] = 1 \). During Step 2, each agent estimates the current swarm distribution \( (\mathcal{F}_{k,i}) \) by first making a localized guess and then reaching an agreement across the network using the consensus algorithm [23]. Step 2 is elucidated in Section III.

Analogous to the random motion of gas molecules which is dictated by the temperature, in this paper we use the difference between the estimated swarm distribution \( (\mathcal{F}_{k,i}) \) and the desired formation \( (\pi) \) to dictate the motion of agents in the swarm. The Hellinger distance (HD) is a symmetric measure of the difference between two probability distributions and it is upper bounded by 1 [27]. As discussed in Section III the tuning parameter \( (\xi_{k}) \) computed in Step 3 is the HD between the current swarm distribution and the desired formation.

It may be desired that the agents in a particular bin can only transition to some bins, while they cannot transition to other bins. During Step 4, each agent checks if it is currently lying within the trapping region \( (\mathcal{T}_{k}^{j}) \) which arises due to the motion constraints. If the agent is actually located in the trapping region \( (R[i] \in \mathcal{T}_{k}^{j}) \), then the agent is sent to bin \( R[\ell] \) which
is best-suited to reach the formation. The concept of trapping region and the method for handling motion constraints are presented in Section IV.

Step 5 involves designing a family of row stochastic Markov transition matrices $M_{k}^{l}$ with $\pi$ as their stationary distributions, which is presented in Section V. When the HD between the estimated swarm distribution and the desired formation is large, each agent propagates its position in a statistically-independent manner so that the swarm asymptotically tends to the desired formation. As this HD decreases, the Markov matrices also tend to an identity matrix and each agent holds its own position. During Step 6, randomly sampling the Markov matrix generates the next location of the agent. The stability and convergence guarantees for PSG–IMC are presented in Section V.

The guidance or motion planning of spacecraft in a swarm is discussed in Section VII. Strategies for implementing PSG–IMC algorithms are demonstrated with numerical examples in Sections VI and VII.

III. Consensus Estimation of Swarm Distribution

As shown in Step 2 of the flowchart in Fig. 2, the agents need to communicate with each other to understand the current swarm distribution and transition accordingly. In this section, we use the decentralized consensus algorithm [23] to estimate the current swarm distribution, as illustrated in Algorithm 1. The objective of the consensus stage is to estimate the current swarm distribution ($\mathcal{F}_{k}^{j}$) and maintain consensus across the network during each time step. This objective is achieved in two steps: (i) each agent locally estimates the swarm distribution, and (ii) the local estimates converge to the global estimate of the current swarm distribution as agents recursively communicate with neighboring agents and update their estimated distribution of the swarm.

Let the row vector $\mathcal{F}_{k,\nu}^{j}$ represent the $j^{th}$ agent’s estimate of the current swarm distribution during the $\nu^{th}$ consensus loop at the $k^{th}$ time instant. At the beginning of the consensus stage of the $k^{th}$ time instant, the $j^{th}$ agent generates a row vector of local estimate of the swarm distribution $\mathcal{F}_{k,0}^{j}$ by only determining its present bin location:

$$\mathcal{F}_{k,0}^{j}[i] = 1 \text{ if } r_{k}^{j}[i] = 1, \text{ otherwise } 0.$$  \hspace{1cm} (1)

In essence, the local estimate at the start of the consensus stage is a discrete representation of the position of the $j^{th}$ agent in the space $\mathcal{R}$, i.e., $\mathcal{F}_{k,0}^{j} = r_{k}^{j}$. Hence the current swarm distribution is also given by $\mathcal{F}_{k}^{j} = \sum_{i=1}^{m} \frac{1}{m} \mathcal{F}_{k,0}^{j}$, which is equal to the ensemble mean of actual agent positions $\{r_{k}^{j}\}_{j=1}^{m}$ over the bins in $\mathcal{R}$.

During the consensus stage, the agents recursively combine and update their local distributions to reach an agreement across the network. The Linear Opinion Pool (LinOP) of probability measures, which has been used for combining subjective distributions [28], [29], is given by:

$$\mathcal{F}_{k,\nu}^{j} = \sum_{\ell \in \mathcal{J}} a_{k,\nu-1}^{j,\ell} \mathcal{F}_{k,\nu-1}^{j,\ell}, \forall j, \ell \in \{1, \ldots, m\}, \forall \nu \in \mathbb{N},$$  \hspace{1cm} (2)

where $\sum_{\ell \in \mathcal{J}} a_{k,\nu-1}^{j,\ell} = 1$. The updated distribution $\mathcal{F}_{k,\nu}^{j}$ after the $\nu^{th}$ consensus stage is a weighted average of the distributions of the inclusive neighbors $\mathcal{F}_{k,\nu-1}^{j,\ell}, \forall \ell \in \mathcal{J}$ at $k^{th}$ time instant.

We introduce Lemma 1 to show that pointwise convergence of the probability distributions is the sufficient condition for convergence of the induced measures in total variation. Let $\mathcal{A}$ be the $\sigma$-algebra of $\mathcal{R}$ and $\mathcal{A}$ be any set in $\mathcal{A}$. If $\mu_{f}(\mathcal{A}) = \sum_{i=1}^{\ell,m} R[i] \in \mathcal{A} f[i]$ for any set in $\mathcal{A}$, then $\mu_{f}$ is defined as the measure induced by the function $f$ on $\mathcal{R}$. The measure $\mu_{f}$ is defined to converge to the measure $\mu_{f}$ in total variation if $\|\mu_{f} - \mu_{f}\|_{TV} = \sup_{\mathcal{A} \in \mathcal{A}} |\mu_{f}(\mathcal{A}) - \mu_{f}(\mathcal{A})|$ goes to zero.

**Lemma 1.** Let $\{f_{n}\}_{n=1}^{\infty}$, $f^{*}$ denote real-valued measurable (probability mass) functions on $\mathcal{R}$ and let $\mu_{f_{n}}$ and $\mu_{f}$ denote the respective induced measures on $\mathcal{A}$. If $f_{n}$ converges to $f^{*}$ pointwise, i.e., $\lim_{n \rightarrow \infty} f_{n} = f^{*}$ pointwise; then the measure $\mu_{f_{n}}$ converges in total variation to the measure $\mu_{f^{*}}$, i.e., $\lim_{n \rightarrow \infty} \mu_{f_{n}}_{TV} \rightarrow \mu_{f^{*}}$.

**Proof:** The proof is similar to that of Scheffe’s theorem [30, pp. 84]. Since all the probability mass functions are bounded by 1, the dominated convergence theorem (cf. Theorem 1.5.6, pp. 23) implies:

$$\lim_{n \rightarrow \infty} \mu_{f_{n}}(\mathcal{A}) = \lim_{n \rightarrow \infty} \sum_{R[i] \in \mathcal{A}} f_{n}[i] = \sum_{R[i] \in \mathcal{A}} \lim_{n \rightarrow \infty} f_{n}[i] = \sum_{R[i] \in \mathcal{A}} f^{*}[i] = \mu_{f^{*}}(\mathcal{A}), \forall \mathcal{A} \in \mathcal{A}.$$  \hspace{1cm} (3)

This relation between measures implies that $\|\mu_{f_{n}} - \mu_{f}\|_{TV} = 0$ and $\lim_{n \rightarrow \infty} \mu_{f_{n}}_{TV} \rightarrow \mu_{f^{*}}$.

Let $W_{k,\nu} = [\mathcal{F}_{1,k,\nu}, \ldots, \mathcal{F}_{m,k,\nu}]$ be a row vector of pmf functions of the agents after the $\nu^{th}$ consensus loop. The LinOP [23] can be expressed concisely as $W_{k,\nu} = W_{k,\nu-1} P_{k,\nu-1}, \forall \nu \in \mathbb{N}$, where $P_{k,\nu-1}$ is a matrix with entries $P_{k,\nu-1}[j, i] = a_{k,\nu-1}^{j,\ell}$.

**Assumption 1.** The communication network topology of the multi-agent system $G(k)$ is strongly connected (SC). The weighting factors $a_{k,\nu-1}^{j,\ell}, \forall j, \ell \in \{1, \ldots, m\}$ and the matrix $P_{k,\nu-1}$ have the following properties: (i) the weighting factors are the same for all consensus loops within each time instant; (ii) the matrix $P_{k}$ conforms with the graph $G(k)$; (iii) the matrix $P_{k}$ is column stochastic; and (iv) the weighting factors $a_{k,\nu-1}^{j,\ell}$ are balanced.

Since $m \gg n_{cell}$ and multiple agents are within the same bin, we feel that establishing a SC balanced network is not difficult. For example, if all communication links are bidirectional then the Metropolis weights can be used by each agent to establish a SC balanced network [20].

Let $\theta_{k,\nu} = [\theta_{k,\nu}^{1}, \ldots, \theta_{k,\nu}^{m}]$ be the disagreement vector, where $\theta_{k,\nu}^{j}$ is the $L_{1}$ distances between $\mathcal{F}_{k,\nu}^{j}$ and $\mathcal{F}_{k}^{j}$, i.e., $\theta_{k,\nu}^{j} = \sum_{R[i] \in \mathcal{R}} |\mathcal{F}_{k,\nu}^{j}[i] - \mathcal{F}_{k}^{j}[i]|$. Since the $L_{1}$ distances between pmfs is bounded by 2, the $\ell_{2}$ vector norm $\|\theta_{k,\nu}\|_{2}$ is upper bounded by $2\sqrt{m}$.

**Theorem 2.** (Consensus using the LinOP on SC Balanced Digraphs) **Under Assumption 1** using the LinOP
each $\tilde{F}_k^{j,v}$ globally exponentially converges to $F_k^j = \sum_{i=1}^{m} \frac{1}{m} \tilde{F}_k^{j,0}$ pointwise with a rate faster or equal to the second largest singular value of $F_k$, i.e., $\sigma_{m-1}(F_k)$. Hence the induced measures converge in total variation globally exponentially, i.e., $\lim_{n \to \infty} \mu_{\tilde{F}_k^{j,v}} \Rightarrow \mu_{F_k^j}$, if for some quantity $\varepsilon_{\text{consensus}} > 0$, the number of consensus loops within each consensus stage is at least $n_{\text{loop}} \geq \left\lceil \frac{\ln(\varepsilon_{\text{consensus}}/(2\sqrt{m}))}{\ln \sigma_{m-1}(F_k)} \right\rceil$, then the $\ell_2$ norm of the disagreement vector at the end of the consensus stage is less than $\varepsilon_{\text{consensus}}$, i.e., $\|\theta_{k,n_{\text{loop}}}\|_2 \leq \varepsilon_{\text{consensus}}$.

It follows from Theorem 2 that each agent has a good estimate of the current swarm distribution if $n_{\text{loop}}$ is sufficiently large. Note that, in order to transmit the pmf $\tilde{F}_k^{j,v}$ to another agent, the agent needs to transmit $n_{\text{cell}}$ real numbers bounded by $[0, 1]$. The feasibility of executing multiple consensus loops within each time step, for a practical scenario involving spacecraft swarms, is discussed in Section VII.

Next, we compute the tuning parameter ($\xi_k^j$) described in Step 3 of the flowchart in Fig. 2.

**Definition 2. (Hellinger Distance based Tuning Parameter $\xi_k^j$)**

The HD is a symmetric measure of the difference between two probability distributions and it is upper bounded by $1$ [27]. Each agent chooses the tuning parameter ($\xi_k^j$), based on the HD, using the following equation:

$$
\xi_k^j = D_H(\pi, \hat{F}_k^{j,n_{\text{loop}}}) = \frac{1}{2} \sqrt{\sum_{i=1}^{n_{\text{cell}}} \left( \sqrt{\pi[i]} - \sqrt{\hat{F}_k^{j,n_{\text{loop}}}[i]} \right)^2},
$$

where $\hat{F}_k^{j,n_{\text{loop}}}$ is the estimated current swarm distribution and $\pi$ is the desired formation. \hfill \Box

**IV. FAMILY OF MARKOV TRANSITION MATRICES**

The key concept of PSG-IMC is that each agent can independently determine its trajectory from the evolution of an inhomogeneous Markov chain with a desired stationary distribution. In this section, we design the family of Markov transition matrices for a desired stationary distribution, using the tuning parameter from Definition 2. The proposed solution is much simpler than the recursive Metropolis–Hastings algorithm [16].

Let $x_k^j \in [0, 1]^{n_{\text{cell}}}$ denote the row vector of probability mass function (pmf) of the predicted position of the $j$th agent at the $k$th time instant, i.e., $x_k^j 1 = 1$. The $i$th element ($x_k^j[i]$) is the probability of the event that the $j$th agent is in $R[i]$ bin at the $k$th time instant:

$$
x_k^j[i] = \mathbb{P}(r_k^j[i] = 1), \quad \forall i \in \{1, \ldots, n_{\text{cell}}\}.
$$

The elements of the row stochastic Markov transition matrix $M_k^j \in [0, 1]^{n_{\text{cell}} \times n_{\text{cell}}}, \forall j \in \{1, \ldots, m\}$ are the transition probabilities of the $j$th agent at the $k$th time instant:

$$
M_k^j[i, \ell] := \mathbb{P}(r_{k+1}^j[\ell] = 1 | r_k^j[i] = 1).
$$

In other words, the probability that the $j$th agent in $R[i]$ bin at the $k$th time instant will transition to $R[\ell]$ bin at the $(k + 1)^{\text{th}}$ time instant is given by $M_k^j[i, \ell]$. The Markov transition matrix $M_k^j$ determines the time evolution of the pmf vector $x_k^j$ by:

$$
x_{k+1}^j = x_k^j M_k^j, \quad \forall k \in \mathbb{Z}^+.
$$

Step 5 in Fig. 2 involves designing a family of Markov transition matrices for each agent $M_k^j$, with $\pi$ as their stationary distributions. The following theorem is used by each agent to find these Markov matrices at each time instant.

**Theorem 3. (Family of Markov transition matrices for a desired stationary distribution)**

Let $\alpha_k^j \in [0, 1]^{n_{\text{cell}}}$ be a nonnegative bounded column vector with $\|\alpha_k^j\|_\infty \leq 1$. For given $\xi_k^j$ from (3), the following parametrized family of row stochastic Markov matrices $M_k^j$ have $\pi$ as their stationary distribution (i.e., $\pi M_k^j = \pi$):

$$
M_k^j = \text{diag}(\alpha_k^j) 1 - \frac{\xi_k^j}{\pi \alpha_k^j} \pi \text{diag}(\alpha_k^j) + I - \xi_k^j \text{diag}(\alpha_k^j),
$$

where $\pi \alpha_k^j \neq 0$ and $\sup \xi_k^j \|\alpha_k^j\|_\infty \leq 1$.

**Proof:** For a valid first term in (7), we need $\pi \alpha_k^j \neq 0$. We first show that $M_k^j$ is a row stochastic matrix. Right multiplying both sides of (7) with $1$ gives:

$$
M_k^j 1 = \xi_k^j \text{diag}(\alpha_k^j) \frac{\pi \alpha_k^j}{\pi \alpha_k^j} 1 - 1 - \xi_k^j \text{diag}(\alpha_k^j) = 1.
$$

Next, we show that $M_k^j$ is a Markov matrix with $\pi$ as its stationary distribution, as $\pi$ is the left eigenvector corresponding to its largest eigenvalue 1, i.e., $\pi M_k^j = \pi$. Left multiplying both sides of (7) with $\pi$ gives:

$$
\pi M_k^j = \frac{\pi \alpha_k^j}{\pi \alpha_k^j} \pi \xi_k^j \text{diag}(\alpha_k^j) + \pi - \pi \xi_k^j \text{diag}(\alpha_k^j).
$$

In order to ensure that all the elements in the matrix $M_k^j$ are nonnegative, we enforce that $1 - \xi_k^j \text{diag}(\alpha_k^j) \geq 0$ which results in the condition $\sup \xi_k^j \|\alpha_k^j\|_\infty \leq 1$.

The additional degrees of freedom due to $\alpha_k^j$ vector allows us to capture the physical distance between bins while designing the Markov matrix and increase the probability of transitioning to nearby bins.

**Definition 3. (Physical distance based $\alpha_k^j$ vector)**

In this paper, if the $i^{\text{th}}$ agent is actually located in $R[i]$ bin at the $k^{\text{th}}$ time instant, then each element of the $\alpha_k^j$ vector is determined using the physical distance between bins in the following manner $\forall \ell \in \{1, \ldots, n_{\text{cell}}\}$:

$$
\alpha_k^j[i] = 1 - \frac{\text{dis}(R[i], R[\ell])}{\max_{q \in \{1, \ldots, n_{\text{cell}}\}} \text{dis}(R[q], R[\ell]) + 1},
$$

where $\text{dis}(R[i], R[\ell])$ is the $\ell_1$ distance between the bins $R[i]$ and $R[\ell]$. If $\kappa[i] \in [0, 1]^{n_{\text{cell}}}$ denotes the location of the centroid of the bin $R[i]$, then $\text{dis}(R[i], R[\ell]) = \|\kappa[i] - \kappa[\ell]\|_1$. Irrespective of the distribution of $\pi$, the condition $\pi \alpha_k^j \neq 0$ is satisfied because $\alpha_k^j[i] > 0, \forall i \in \{1, \ldots, n_{\text{cell}}\}$. Moreover, the condition $\sup \xi_k^j \|\alpha_k^j\|_\infty \leq 1$ is satisfied as $\xi_k^j = D_H(\pi, \hat{F}_k^{j,n_{\text{loop}}}) \leq 1$ in (3) and $\|\alpha_k^j\|_\infty = 1$ in (5).

The evolution of the agent’s location during each time step is based on random sampling of the Markov transition matrix. An agent is said to have undergone a transition if it jumps
from bin \(R[i]\) to bin \(R[\ell]\), \(\ell \neq i\) during a given time step. If \(\xi_k^j = 0\) in (7), then \(M_k^j = \mathbf{I}\) and no transitions occur as the transition probabilities from a bin to any bin (other than itself) goes to zero. If \(\xi_k^j\) is large (close to 1), the agents vigorously transition from one bin to another. In this paper, we seek to minimize unnecessary bin-to-bin transitions during each time step while maintaining or reconfiguring the formation.

In [13], [16], the same homogeneous Markov transition matrix is used by all agents for all time, i.e., \(M_k^j = M\), \(\forall j, \forall k\). Since there is no inter-agent communication, the agents do not realize whether the desired formation has already been achieved and hence continue to transition for all time steps. In contrast, in this paper, each agent executes a different time-varying Markov matrix \(M_k^j\) at each time step. We show that the inhomogeneous Markov chain not only guides the agents so that the swarm distribution converges the desired steady-state distribution but also reduces the number of transitions. Finally, we solve the guidance (motion-planning) problem by realizing a trajectory from initial to terminal positions.

V. CONVERGENCE ANALYSIS OF INHOMOGENEOUS MARKOV CHAINS

In this section, we study the stability and convergence characteristics of the proposed PSG–IMC algorithm without motion constraints. Theorem 4 states that each agent’s pmf vector \(x_k^j\) asymptotically converges pointwise to the desired formation \(\pi\) while Theorem 5 states that the swarm distribution \(F_k^j\) also asymptotically converges pointwise to the desired formation \(\pi\), when the number of agents tends to infinity. Finally, Theorem 7 states that if the number of consensus loops executed by each agent tends to infinity, then the Markov transition matrix \(M_k^j\) asymptotically converges to an identity matrix. Therefore, the first two objectives of PSG–IMC, stated in Section II, are achieved.

The time evolution of the pmf vector \(x_k^j\), defined in (4), from an initial condition \((x_0^j)\) to the \(k^\text{th}\) time instant is given by the inhomogeneous Markov chain:

\[
x_k^j = x_0^j M_0^j M_1^j \cdots M_{k-2}^j M_{k-1}^j, \quad \forall k \in \mathbb{Z}^+,
\]

where each \(M_k^j\) is a \(n_{\text{cell}} \times n_{\text{cell}}\) row stochastic matrix obtained using Theorem 3. Let \(U_{s,t}^j\) denote the matrix product defined by the forward matrix multiplication:

\[
U_{s,t}^j = M_s^j M_{s+1}^j \cdots M_{s+t-1}^j, \quad \forall s \in \mathbb{Z}^+, \forall t \in \mathbb{N}.
\]

Since the overall time evolution of the pmf \(x_k^j\) expressed by (9) can be written concisely as \(x_k^j = x_0^j U_{0,k}^j\). We first focus on the convergence of each agent’s pmf vector.

**Theorem 4.** (Convergence of inhomogeneous Markov chains) Each agent’s time evolution of the pmf vector \(x_k^j\), from any initial condition \(x_0^j \in \mathbb{R}^{n_{\text{cell}}\times 1}\), is given by the inhomogeneous Markov chain (9). If each agent executes the PSG–IMC algorithm, then \(x_k^j\) asymptotically converges pointwise to the desired stationary distribution \(\pi\), i.e., \(\lim_{k \to \infty} x_k^j = \pi\) pointwise for all \(j \in \{1, \ldots, m\}\).

**Proof:** According to Theorem 3, each matrix \(M_k^j\) is a function of the stochastic parameter \(\xi_k^j\) which is determined by \(3\) and bounded by \([0,1]\). Let us define the bins with non-zero probabilities in \(\pi\) as recurrent bins. According to the design of the Markov matrix, if \(\xi_k^j > 0\), then an agent can enter these recurrent bins from all other bins. The bins which are not recurrent are called transient bins and they have zero probabilities in \(\pi\). The agents can only leave the transient bins when \(\xi_k^j < 0\), but they can never enter a transient bin from any other bin. In this proof, we first consider the special case where all the bins are recurrent and show that each agent’s pmf vector converges to \(\pi\). We next consider the general case where transient bins are also present and show that this situation converges to the special case geometrically fast.

**Case 1:** All bins are recurrent, i.e., \(\pi[i] > 0\), \(\forall i \in \{1, \ldots, n_{\text{cell}}\}\).

For \(\xi_k^j > 0\), Theorem 3 guarantees that the matrix \(M_k^j\) is nonnegative and row stochastic. Since \(\pi[i] > 0\), \(\forall i \in \{1, \ldots, n_{\text{cell}}\}\), \(M_k^j\) is strongly connected and also irreducible by [32] Theorem 6.2.24, pp. 362]. As all the diagonal entries of \(M_k^j\) are positive, then \(M_k^j > 0\) by [32] Lemma 8.5.5, pp. 517]. Finally, the primitive matrix theorem [32] Theorem 8.5.2, pp. 516] implies that \(M_k^j\) is a primitive matrix.

We next prove that \(\lim_{k \to \infty} U_{s,t}^j\) is also a primitive matrix. Forward multiplication of two row stochastic Markov matrices \((M_j, M_{j+1}^j\) with \(\xi_j^j, \xi_{j+1}^j > 0\) and \(s \in \mathbb{Z}^+\) obtained using Theorem 3 yields the nonnegative, strongly connected, row stochastic matrix \(M_j M_{j+1}^j\). Similar to the above case, \(M_j M_{j+1}^j\) is irreducible, \(M_j M_{j+1}^j > 0\) and the primitive matrix theorem [32] Theorem 8.5.2, pp. 516] implies that \(M_j M_{j+1}^j\) is a primitive matrix. Hence, the matrix product of Markov matrices, obtained using Theorem 3 with non-zero tuning parameters, always results in a primitive matrix.

If \(\xi_k^j = 0\), then \(M_k^j = \mathbf{I}\) from (7). The matrix product \(U_{s,t}^j\) from (10) can be decomposed into two parts; (a) the tuning parameter \(\xi_s^j\) of the matrices in the first part is always zero and (b) the second part contains the remaining Markov matrices with \(\xi_s^j > 0\).

\[
U_{s,t}^j = \left(\mathbf{I} \cdots \mathbf{I}\right) \cdot \left(M_s^j M_{s+2}^j \cdots M_{s+t-3}^j M_{s+t-1}^j\right).
\]

Since the first part results in an identity matrix, the matrix product \(U_{s,t}^j\) can be completely defined by the second part containing Markov matrices with \(\xi_s^j > 0\). In the matrix product \(\lim_{k \to \infty} U_{0,k}^j\), the tuning parameters \(\xi_k^j\) for the first few Markov matrices are non-zero because the swarm starts from initial conditions that are different from the desired formation. Hence there exists at least one matrix in the second part of \(\lim_{k \to \infty} U_{0,k}^j\). Thus we can prove (by induction) that the matrix product \(\lim_{k \to \infty} U_{0,k}^j\) is a primitive matrix.

Next, we prove that the matrix product \(\lim_{k \to \infty} U_{0,k}^j\) is asymptotically homogeneous and strongly ergodic. Due to the quantization of the pmf by the number of agents \(m\) and the square root function in the Hellinger distance in (3), the smallest positive tuning parameter \(\xi_{\text{min}}\) is given by:

\[
\xi_{\text{min}} = \frac{1}{\sqrt{m}} \leq \min_{j \in \{1, \ldots, m\}, k \in \mathbb{Z}^+} \xi_k^j.
\]

The smallest positive element in the \(\alpha_k^j\) vector in (8) is given...
by:

$$\alpha_{\min} = \left(1 - \frac{\max_{i,q \in \{1, \ldots, n_{\text{cell}}\}} \text{dis}(R[q], R[i])}{\max_{i,q \in \{1, \ldots, n_{\text{cell}}\}} \text{dis}(R[q], R[i]) + 1}\right).$$  \(12\)

Finally, the smallest positive element in the stationary distribution \(\pi\) is given by \(\pi_{\min} = \left(\min_{i \in \{1, \ldots, n_{\text{cell}}\}} \pi[i]\right).\) The diagonal and off-diagonal elements of the Markov matrix \(M_k^j\) designed using (7) are given by:

$$M_k^j[i, i] = 1 - \xi_k^j \alpha_k^j[i] + \frac{\xi_k^j}{\pi_k^j} (\alpha_k^j[i])^2 \pi[i],$$  \(13\)

$$M_k^j[i, \ell] = \frac{\xi_k^j}{\pi_k^j} \alpha_k^j[i] \pi[j] \alpha_k^j[\ell],$$  \(14\)

Irrespective of the choice of \(\xi_k^j\), the diagonal elements of \(M_k^j\) will not tend to zero. But the off-diagonal elements of \(M_k^j\) will tend to zero if \(\xi_k^j \to 0\). Moreover, the largest possible value of \(\pi \alpha_k^j[i]\) is 1, when \(\alpha_k^j[i] = 1\). Hence the smallest non-zero element in \(M_k^j\) is lower bounded by the smallest possible values of the terms in (13).

Thus we get a positive \(\gamma\) that is independent of \(k\), i.e., \(\gamma = \min \alpha_k^j \min \pi \alpha_k^j\). If \(\xi_k^j \to 0\) then, \(\lim_{i,j \in \{1, \ldots, n_{\text{cell}}\}} M_k^j[i, \ell] = 1 > \gamma.\) Since \(M_k^j\) is row stochastic, \(M_k^j[i, \ell] \leq 1, \forall i, \ell \in \{1, \ldots, n_{\text{cell}}\}, \forall k \in \mathbb{Z}^+.\) Thus each Markov matrix satisfies the condition (32) in Lemma [9] given in Appendix.

All the Markov transition matrices, designed using Theorem 3, have \(\pi\) as their left eigenvector for the eigenvalue 1. Hence \(\psi_0 = \pi\) is a set of absolute probability vectors for \(\lim_{k \to \infty} U_{0,k}^j\). According to Lemma 10 given in Appendix, \(\psi_0 = \pi\) is the unique set of absolute probability vectors for \(\lim_{k \to \infty} U_{0,k}^j\) and \(\psi_0 = \pi\) in (30). Hence the individual pmfs asymptotically converge to:

$$\lim_{k \to \infty} x_k^j = \lim_{k \to \infty} x_0^j U_{0,k}^j = x_0^j \pi = \pi.$$  \(15\)

By Lemma [1] the measure induced by \(x_0^j\) on \(\mathbb{B}\) converges in total variation to the measure induced by \(\pi\) on \(\mathbb{B}\), i.e.,

$$\lim_{k \to \infty} \mu_{x_0^j} \xrightarrow{T.V.} \mu_{\pi}.$$  ■

**Case 2: Both transient and recurrent bins are present.**

Without loss of generality, let us reorder the bins such that the first \(n_{\text{rec}}\) bins are recurrent, and the remaining bins are transient. Hence, \(\pi[i] > 0, \forall i \in \{1, \ldots, n_{\text{rec}}\}\) and \(\pi[i] = 0, \forall i \in \{n_{\text{rec}} + 1, \ldots, n_{\text{cell}}\}\). It is known that the agents leave the transient bins geometrically fast [34, Theorem 4.3, pp. 120]. Once all the agents are in the recurrent bins, then the situation is similar to case 1 with \(n_{\text{rec}}\) bins. Then, it follows from case 1 that \(\lim_{k \to \infty} x_k^j = \pi\).

Since each agent’s pmf vector converges to \(\pi\), we now focus on the convergence of the current swarm distribution.

**Theorem 5.** (Convergence of swarm distribution to desired formation) If the number of agents, executing the PSG–IMC algorithm, tends to infinity; then the current swarm distribution \((F_k^j = \frac{1}{n} \sum_{i=1}^{n} r_k[i])\) asymptotically converges pointwise to the desired stationary distribution \(\pi\), i.e., \(\lim_{k \to \infty} \lim_{n \to \infty} F_k^j = \pi\) pointwise.

**Proof:** Let \(X_k^j[i]\) denote the independent Bernoulli random variable representing the event that the \(j\)th agent is actually located in bin \(R[i]\) at the \(k\)th time instant, i.e., \(X_k^j[i] = 1\) if \(r_k[i] = 1\) and \(X_k^j[i] = 0\) otherwise. Let \(X_k^j[i]\) denote the random variable \(\lim_{k \to \infty} X_k^j[i]\). Theorem 4 implies that the success probability of \(X_k^j[i]\) is given by \(P(X_k^j[i] = 1) = \lim_{k \to \infty} X_k^j[i] = \pi[i].\) Hence \(E[X_k^j[i]] = \pi[i] \cdot 1 + (1 - \pi[i]) = 0 = \pi[i],\) where \(E[i]\) is the expected value of the random variable. If \(S_k[i] = X_k^j[i] + \ldots + X_k^m[i],\) then the strong law of large numbers (c.f. [35, pp. 85]) states that:

$$P\left(\lim_{m \to \infty} \lim_{k \to \infty} S_k[i] = \pi[i]\right) = 1.$$  \(16\)

The current swarm distribution is also given by \(F_k^j[i] = \frac{1}{n} \sum_{i=1}^{n} r_k[i] = \frac{S_k[i]}{m}.\) Hence (16) implies that \(\lim_{m \to \infty} \lim_{k \to \infty} F_k^j = \pi\) pointwise. By Lemma [1] the measure induced by \(F_k^j[i]\) on \(\mathbb{B}\) converges in total variation to the measure induced by \(\pi\) on \(\mathbb{B}\), i.e., \(\lim_{k \to \infty} \mu_{F_k^j} \xrightarrow{T.V.} \mu_{\pi}.\) ■

In practical scenarios, the number of agents is finite, hence we need to specify a convergence error threshold. The following theorem gives the minimum number of agents needed to establish \(\epsilon\)-convergence of the swarm.

**Theorem 6.** For some acceptable convergence error \(\epsilon_{\text{conv}} > 0,\) if the number of agents is at least \(m \geq \frac{1}{4\epsilon_{\text{conv}}}\) then the pointwise error probability for each bin is bounded by \(\epsilon_{\text{conv}},\) i.e., \(P\left(\lim_{k \to \infty} \frac{S_k[i]}{m} - \pi[i] \geq \epsilon_{\text{conv}}\right) \leq \epsilon_{\text{conv}}, \forall i \in \{1, \ldots, n_{\text{cell}}\}\). ■

**Proof:** The variance of the independent random variable from Theorem 5 is \(Var(X_k^j[i]) = \pi[i](1 - \pi[i]),\) hence \(Var(\lim_{k \to \infty} S_k^j[i]) = \frac{\pi[i](1 - \pi[i])}{m}.\) The Chebychev’s inequality (cf. [30, Theorem 1.6.4, pp. 25]) implies that for any \(\epsilon_{\text{conv}} > 0,\) the pointwise error probability for each bin is bounded by:

$$P\left(\left|\lim_{k \to \infty} \frac{S_k[i]}{m} - \pi[i]\right| > \epsilon_{\text{conv}}\right) = \frac{\pi[i](1 - \pi[i])}{m \varepsilon_{\text{conv}}} \leq \frac{1}{4m \varepsilon_{\text{conv}}}.$$  ■

Hence, the minimum number of agents is given by \(\frac{1}{4\epsilon_{\text{conv}}} \leq n_{\text{conv}}.\)

We now study the convergence of the tuning parameter and the Markov matrix.

**Theorem 7.** (Convergence of Markov matrix to identity matrix) If the number of agents, executing the PSG–IMC algorithm, tends to infinity and the number of consensus loops within each time step also approaches infinity; then the Markov matrix asymptotically converges to the identity matrix, i.e., \(\lim_{m \to \infty} \lim_{n \to \infty} M_k = I.\)

**Proof:** Theorem 2 states that, if \(n_{\text{loop}} \geq \frac{\ln(\varepsilon_{\text{consensus}})/(2\sqrt{\sigma_i})}{\ln(\sigma_i^{-1}(P_x))}\) then \(\|\theta_k, n_{\text{loop}}\|_2 \leq \varepsilon_{\text{consensus}}.\) Therefore if \(n_{\text{loop}} \to \infty,\) then \(\epsilon \to 0,\) i.e., \(\lim_{n_{\text{loop}} \to \infty} F_k^j[n_{\text{loop}}] = F_k^j.\) Theorem 5 implies that the limiting tuning parameter (3) is given by:
lim_{n \to \infty} \lim_{k \to \infty} \lim_{n_{loop} \to \infty} \xi_k^j = \lim_{m \to \infty} \lim_{k \to \infty} \lim_{n_{loop} \to \infty} D_H(\pi, \hat{\xi}_k^j) = \lim_{m \to \infty} D_H(\pi, \hat{\xi}_k^j) = D_H(\pi, \pi) = 0.

Therefore Theorem 8 implies that the Markov transition matrix
\[
M_k^j = \lim_{n \to \infty} \lim_{k \to \infty} \lim_{n_{loop} \to \infty} D_H(\pi, \hat{\xi}_k^j) = \text{I}.
\]

In practical scenarios, \(n_{loop}\) is finite, hence \(\xi_k^j\) may not converge to zero. A practical workaround to avoid transitions due to nonzero \(\xi_k^j\) is to set \(M_k^j = \text{I}\) when \(\xi_k^j\) is sufficiently small and has leveled out. In the next section, we modify the above theorems to prove convergence of the PSG–IMC algorithm with motion constraints.

VI. Motion Constraints

In this section, we introduce additional constraints on the motion of agents and study its effect on the convergence of the swarm. We first introduce the motion constraints and the corresponding trapping problem. Next, we discuss the strategy for leaving the trapping region and an additional condition on the desired formation. Finally, Theorem 8 shows that each agent’s pmf vector \(\xi_k^j\) asymptotically converges pointwise to the desired formation \(\pi\), if it executes the PSG–IMC algorithm with motion constraints.

The agents in a particular bin can only transition to some bins but cannot transition to other bins because of the dynamics or physical constraints. These (possibly time-varying) motion constraints are specified in a matrix \(A^j_k\) as follows:

\[
A_k^j[i, \ell] = \begin{cases} 1 & \text{if the } j^{th} \text{agent can transition to } R[\ell] \\ 0 & \text{if the } j^{th} \text{agent cannot transition to } R[\ell] \end{cases},
\]

where \(r_k^i = 1\), \(\forall i, \ell \in \{1, \ldots, n_{cell}\}\). (17)

We assume that the graph conforming to the \(A^j_k\) matrix is strongly connected, as the agents should be able to move from any bin to any other bin within the state space. We also assume that an agent can always choose to remain in its present bin, i.e., \(A_k^j[i, i] = 1\), \(\forall k, j\). In [16], the iterative Metropolis–Hastings algorithm is used to design a Markov transition matrix \(M_k^j\) for handling the motion constraints. In this paper, we introduce a much simpler method for handling motion constraints, such that the convergence results in Section V are not affected.

Our key idea is to capture the motion constraints using the \(\alpha_k^j\) vector. For a bin \(R[i]\), let us define \(A_k^j(R[i])\) as the set of all bins that the \(j^{th}\) agent can transition to at the \(k^{th}\) time instant:

\[
A_k^j(R[i]) := \bigcup_{\ell \in \{1, \ldots, n_{cell}\}} \{ R[\ell] : A_k^j[i, \ell] = 1 \}.
\]

Similarly, let us define \(\Pi\) as the set of all bins that have non-zero probabilities in the desired formation (\(\pi\)):

\[
\Pi := \bigcup_{\ell \in \{1, \ldots, n_{cell}\}} \{ R[\ell] : \pi[\ell] > 0 \}.
\]

If the \(j^{th}\) agent is actually located in \(R[i]\) bin at the \(k^{th}\) time instant, then each element of the modified \(\alpha_k^j\) vector that captures this motion constraint is given by:

\[
\alpha_k^j[i, \ell] := \begin{cases} 1 & \text{if } R[i] \in A_k^j(R[i]) \text{ and } R[\ell] \in A_k^j(R[i]) \text{ is best suited to reach } \Pi \\ 0 & \text{otherwise} \end{cases},
\]

(22)

which is used instead of the Markov matrix in (10). Note that this secondary condition would not cause an infinite loop as the graph conforming to the \(A_k^j\) matrix is strongly connected.

It is possible that the set of all bins that have non-zero probabilities in the desired formation (\(\Pi\)) can be decomposed into subsets, such that any agent cannot transition from one subset to another subset without exiting the set \(\Pi\) (for example...
see Fig. 3(a)). Since our proposed algorithm suggests that the agents will always transition within ℐ after entering it, the agents in one such subset will never transition to the other subsets. Hence, the proposed algorithms will not be able to achieve the desired formation, as the agents in each subset of ℐ are trapped within that subset. In order to avoid such situations, we need to follow the assumption on ℐ and \( A_k^j \).

**Assumption 2.** The set of all bins that have non-zero probabilities in the desired formation (ℐ) and the matrix specifying the motion constraints (\( A_k^j \)) are such that each agent can transition from any bin in ℐ to any other bin in ℐ, without exiting the set ℐ while satisfying the motion constraints.

For the \( j \)th agent in bin \( R[i] \), random sampling of the Markov chain selects the bin \( R[q] \) in line (15) of Algorithm 1. Then the time evolution of the pmf vector \( x_k^j \) can be written as:

\[
x_{k+1}^j = x_k^j B_k^j, \text{ where } B_k^j = \begin{cases} C_k^j & \text{if } R[i] \in T_k^j \\ M_k^j & \text{otherwise} \end{cases}.
\]

(23)

Similar to (9), the evolution of the probability vector from initial condition to any \( k \)th time instant is given by the matrix product:

\[
x_k^j = x_0^j B_k^j B_k^{j-1} \cdots B_k^0 I_{s}, \quad \forall k \in \mathbb{Z}^*.
\]

(24)

where each \( B_k^j, \ell = \{0, \ldots, k-1\} \) is either the row stochastic Markov matrix \( M_k^j \) obtained using Theorem 1 or the \( C_k^j \) matrix obtained using (22). Let \( V_k^{j,s} \) denote the row stochastic matrix defined by the backward matrix multiplication:

\[
V_{s,t}^j = B_k^j B_k^{j-1} \cdots B_k^0 I_{s}, \quad \forall s \in \mathbb{Z}^*, \forall t \in \mathbb{N}.
\]

(25)

Hence the overall time evolution of the pmf \( x_k^j \) expressed by (9) can be written concisely as \( x_k^j = x_0^j V_{0,k}^j \).

**Theorem 8.** (Convergence of inhomogeneous Markov chains with motion constraints) Under Assumption 2, each agent’s time evolution of the pmf vector \( x_k^j \), from any initial condition \( x_0^j \in \mathbb{R}_{\geq 0}^{|\text{cell}|} \), is given by the inhomogeneous Markov chain (24). If each agent executes the PSG–IMC algorithm, then \( x_k^j \) asymptotically converges pointwise to the desired stationary distribution \( \pi \), i.e., \( \lim_{k \to \infty} x_k^j = \pi \) pointwise for all \( j \in \{1, \ldots, m\} \).

**Proof:** We only need to show that there are finitely many occurrences of the \( C_k^j \) matrix in the matrix product \( \lim_{k \to \infty} V_{k,0}^j \). Due to the design of the Markov matrix in Theorem 1, the bins in the set ℐ are absorbing; i.e., if an agent enters any of the bins in the set ℐ, then it cannot leave the set ℐ. Next we notice that if \( R[\ell] \notin T_k^j \), then the only possible transitions are to the bins in ℐ. Thus, once the agent is out of the set \( T_k^j \), it cannot enter it again. Finally, the number of steps inside the set \( T_k^j \) is limited by the size of the set \( |T_k^j| < |\text{cell}| \). Hence the \( C_k^j \) can only occur finite number of times in the matrix product \( \lim_{k \to \infty} V_{k,0}^j \). The new initial condition of the agent is obtained by forward multiplying the previous initial condition with the \( C_k^j \) matrices:

\[
x_0^j = x_0^j C_0^j C_1^j \cdots C_{s-1}^j C_s^j.
\]

(26)

where \( s \) is the maximum number of steps that the \( j \)th agent makes in the set \( T_k^j \). Hence the overall time evolution of the pmf \( x_k^j \) can be written as \( \lim_{k \to \infty} x_k^j = \lim_{k \to \infty} x_0^j V_{0,k}^j \).

Once the agent has exited the set \( T_k^j \), the situation is exactly similar to that discussed in Theorem 1. In case 1 of the proof of Theorem 4, the Markov matrix \( M_k^j \) is strongly connected when \( \xi_j > 0 \) due to Assumption 2 and the strongly connectedness of the \( A_k^j \) matrix. Hence it follows from Theorem 4 that \( \lim_{k \to \infty} x_k^j = \pi \) pointwise for all \( j \in \{1, \ldots, m\} \).

Note that Theorems 3 and 7 can be directly applied to satisfy the first two objectives of PSG–IMC, even under motion constraints.

Algorithm 1 illustrates the pseudo code for the PSG–IMC algorithm with motion constraints. Each agent determines its current location, estimates the current swarm distribution, and checks if it is in the trapping region. Each agent then computes the Markov matrix and determines its next location.

**Algorithm 1** Probabilistic swarm guidance algorithm using inhomogeneous Markov chains (PSG–IMC)

1: (one cycle of \( j \)th agent during \( k \)th time instant)
2: Agent determines its present bin, e.g., \( r_k^j[i] = 1 \)
3: Set \( n_{\text{loop}} \), the weighting factors \( a_k^j \)
4: for \( \nu = 1 \) to \( n_{\text{loop}} \)
5: if \( \nu = 1 \) then Set \( F_{k,0}^j \) from \( r_k^j \) end if
6: Transmit the pmf \( F_{k,\nu-1}^j \) to other agents
7: Obtain the pmfs \( F_{k,\nu-1}^j, \forall \ell \in J_k^j \)
8: Compute the new pmf \( F_{k,\nu}^j \), using LinOP
9: end for
10: Compute the tuning parameter \( \xi_k^j \)
11: if \( R[\ell] \in T_k^j \) then Go to bin \( R[\ell] \)
12: else Compute the \( \alpha_k^j \) vector
13: Compute the Markov matrix \( M_k^j \)
14: Generate a random number \( z \in \text{unif}[0;1] \)
15: Go to bin \( R[q] \) such that \( \sum_{\ell=1}^{q-1} M_k^j[i, \ell] \leq z < \sum_{\ell=1}^{q} M_k^j[i, \ell] \)
16: end if

**A. Numerical Example**

In this example, the PSG–IMC algorithm with motion constraints is used to guide a swarm containing \( m = 2000 \) agents to the desired formation \( \pi \) associated with the UIUC logo shown in Fig. 3(b). Monte Carlo simulations were performed
and the cumulative results from 50 runs are shown in Fig. 4. As shown in Fig. 3 at $k = 1$, each simulation starts with the agents uniformly distributed across $R \subset \mathbb{R}^2$, which has been partitioned into $30 \times 30$ bins. Each agent independently executes the PSG–IMC with motion constraints, illustrated in Algorithm 1. During the consensus stage, each agent is allowed to communicate with those agents which are at most 10 steps away. If $\kappa[i] = (x[i], y[i])$ denotes the location of the bin $R[i]$ in the $30 \times 30$ grid, then the communication matrix $P_k$ is written as follows:

$$P_k[\ell, j] = \begin{cases} 
\frac{1}{\max(|x[j]|, |y[j]|)} & \text{if } j \neq \ell, r_k[i] = 1, r_k[q] = 1, \\
1 - \sum_{s \neq j} a_{k,\ell-1} & \text{if } j = \ell \\
0 & \text{otherwise}
\end{cases}$$

\begin{equation}
\forall j, \ell, s \in \{1, \ldots, m\}, \forall i, q \in \{1, \ldots, n_{cell}\}, \forall k \in \mathbb{N}.
\end{equation}

Note that $P_k$ is symmetric and stochastic, hence it satisfies Assumption 1. Moreover, each agent is allowed to transition to only those bins which are at most 5 steps away. Hence the motion constraint matrix is written as follows:

$$A_k[i, q] = \begin{cases} 
1 & \text{if } |\kappa[i] - \kappa[q]|_1 \leq 5 \\
0 & \text{otherwise}
\end{cases}$$

\begin{equation}
\forall j \in \{1, \ldots, m\}, \forall i, q \in \{1, \ldots, n_{cell}\}, \forall k \in \mathbb{N}.
\end{equation}

As shown in the HD graph in Fig. 4(a), the desired formation is almost achieved within the first 20 time steps for all simulation runs. After the 20th time step, the swarm is externally damaged by eliminating approximately 350±35 agents from the middle section of the formation. This can be seen by comparing the images for the 20th and 21st time step in Fig. 5. Note that the swarm always recovers from this damage and attains the desired formation within another 10 time steps. Thus the third objective of PSG–IMC, stated in Section III, is also achieved. From the correlation between the plots of HD and number of transitions in Fig. 4(b), we can infer that the agents transition only when necessary. It is evidently from the cumulative results of the 50 simulation runs in Fig. 4 that the PSG–IMC algorithm works reliably well and achieves the desired goals for all simulations runs.

VII. GUIDANCE AND MOTION PLANNING OF SWARMS OF SPACECRAFT

The PSG–IMC algorithm determines the desired trajectory of each agent from its current location to the final location so that the overall swarm converges to the desired formation. Since it is technically feasible to develop and deploy swarms (100s–1000s) of femtosatellites, in this section we solve the guidance problem for such swarms of spacecraft in Earth orbit.

Assume that the spacecraft are located in the local coordinate frame rotating around Earth. For some initial conditions and no further control input, the solution to the linear Hill–Clohessy–Wiltshire (HCW) equations, for each deputy spacecraft orbiting the chief spacecraft, is a closed elliptical trajectory in the local relative frame as shown in Fig. 6(a). Note that the semi-major axis along $\hat{x}$ axis is twice that of the semi-minor axis along $\hat{y}$ axis.

Motion constraints could arise from the spacecraft dynamics, as it might be infeasible for the spacecraft in a certain bin to transition to another bin within a single time step due to the large distance between these bins. Some constraints could also arise from a limit on the amount of fuel that can be consumed in a single time step. If time-varying bins $(R_k[i], \forall i \in \{1, \ldots, n_{cell}\})$ are designed so that the spacecraft continues to coast along the elliptical HCW solution in the local frame, as shown in Fig. 6(a), then no control input will be required to transition from $R_k[i]$ to $R_{k+1}[i]$. Moreover, if there is an upper limit on the fuel consumed during each time step, then the spacecraft in $R_k[i]$ bin can only transition to the light blue cells in Fig. 6(b) during the $(k+1)$th time instant. Thus a time-varying motion constraints matrix $A_k[i]$ can be designed to handle motion constraints due to spacecraft dynamics. We have already shown that if each spacecraft executes the PSG–IMC algorithm, then the swarm converges to the desired formation and the Markov matrices converge to the identity matrix.

We now extend the example discussed in Section VI-A to a swarm of spacecraft in Earth orbit. If $\tilde{x}[i], \tilde{y}[i]$ denotes the location of the bin $R[i]$ in the $30 \times 30$ grid, then the time-varying location of the centroid of the bin in local frame is given by:
\[ \kappa_k[i] = \left( \frac{1}{2}(1 + \frac{1}{15} x[i]) \sin \left( \frac{x}{10} k + \frac{300}{200} y[i] \right) \right) \left( 1 + \frac{1}{15} x[i] \right) \cos \left( \frac{x}{10} k + \frac{300}{200} y[i] \right). \] (29)

Fig. 6(b) shows the locations of the time-varying bins at different time steps along with the respective swarm distributions. Similar to the previous example, the swarm converges to the desired formation and the spacecraft settles down after the final formation has been achieved.

In order to obtain an estimate of the communication load, let us assume that during each of the 20 consensus loops, each agent needs to transmit a fraction \( F_{j, \nu} \) of its estimated pmf at 100 Kbps XBee radio [37] to 100 other agents and receives pmf estimates from 100 neighboring agents. The total transmission time using a 250 Kbps XBee radio [37] is \( \frac{900 \times 5.2 \times 100 \times 20}{250 \times 10^6} \approx 2 \) minutes. This is significantly less than the time step of 8 – 10 minutes used for such missions, hence we conclude that executing multiple consensus loops within each time step is feasible.

VIII. CONCLUSIONS

This paper presents a new approach to shaping and reconfiguring a large number of autonomous agents in a probabilistic framework. In the proposed PSG–IMC algorithm, each agent independently determines its own trajectory so that the overall swarm asymptotically converges to the desired formation, which is robust to external disturbances or damages. Compared to prior work using homogeneous Markov chains for all agents, the proposed algorithm using inhomogeneous Markov chains helps the agents avoid unnecessary transitions when the swarm converges to the desired formation. We present a novel technique for constructing inhomogeneous Markov matrices in a distributed fashion, thereby achieving and maintaining the desired swarm distribution while satisfying motion constraints. Using a consensus algorithm along with communication with neighboring agents, the agents estimate the current swarm distribution and transition so that the Hellinger distance between the estimated swarm distribution and the desired formation is minimized. The application of PSG–IMC algorithm to guide a swarm of spacecraft in Earth orbit has been discussed. Results from multiple simulation runs demonstrate the properties of self-repair capability, reliability and improved convergence of the proposed PSG–IMC algorithm.

APPENDIX

In this section, we introduce some key definitions and lemmas that are used in the proof of Theorem 4.

**Definition 5. (Strong Ergodicity and Asymptotic Homogeneity)** The matrix product \( U_{s,t}^j \) is defined to be strongly ergodic if
\[ \lim_{t \to \infty} U_{s,t}^j = \begin{cases} w^j_s, & \forall t \in \mathbb{N}, \forall j \in \{1, \ldots, m\} \end{cases} \] (30)
where \( w^j_s \) is a row probability vector that will depend on \( s \) (i.e., \( w^1_s = 1 \)).

A Markov chain [9] is defined to be asymptotically homogeneous (with respect to \( \psi \)) if there exists a row probability vector \( \psi \) (i.e., \( \psi 1 = 1 \)) such that [34, pp. 92]:
\[ \lim_{k \to \infty} \psi M_k^j = \psi, \quad \forall j \in \{1, \ldots, m\} \] (31)

where \( M_k^j 1 = 1 \) and \( \psi 1 = 1 \).

**Lemma 9.** [34, pp. 97] (Asymptotic Homogeneity implies Strong Ergodicity) If the matrix product \( \lim_{t \to \infty} U_{s,t}^j \) is a primitive matrix and there exists a positive \( \gamma \) independent of \( k \) such that the following condition holds \( \forall k \in \mathbb{Z}^+ \) and \( \forall j \in \{1, \ldots, m\} \):
\[ 0 < \gamma \leq \min_{i, \ell \in \{1, \ldots, n_{cell}\}} M_k^j[i, \ell], \quad \max_{i, \ell \in \{1, \ldots, n_{cell}\}} M_k^j[i, \ell] \leq 1, \quad \forall k \in \mathbb{Z}^+, \] (32)
then asymptotic homogeneity is necessary and sufficient for strong ergodicity of \( \lim_{t \to \infty} U_{s,t}^j \).

**Lemma 10.** [33] (Uniqueness of Absolute Probability Vectors) A set of absolute probability vectors for the matrix product \( \lim_{t \to \infty} U_{s,t}^j \) is defined to be a sequence \( \{w^j_s\} \) of probability (row) vectors such that \( \forall j \in \{1, \ldots, m\} \):
\[ \lim_{t \to \infty} w^j_{s,t} U_{s,t}^j = w^j_s, \quad \forall s \in \mathbb{Z}^+, \forall t \in \mathbb{N} \] (33)

The matrix product \( \lim_{t \to \infty} U_{s,t}^j \) is strongly ergodic if and only if there is only one set of absolute probability vectors \( \{w^j_s\} \).

ACKNOWLEDGMENT

The authors thank D. Bayard, B. Açıkmese, D. Morgan, and A. Gupta for stimulating discussions on this paper.

REFERENCES

[1] F. Y. Hadaegh, S.-I. Chung, and H. M. Manohara, “On development of 100-gram-class spacecraft for swarm application,” IEEE Syst. J., 2013.

[2] S.-J. Chung, S. Bandyopadhyay, I. Chang, and F. Y. Hadaegh, “Phase synchronization control of complex networks of Lagrangian systems on adaptive digraphs,” Automatica, vol. 49, pp. 1148–1161, May 2013.

[3] M. Mesbah and F. Y. Hadaegh, “Formation flying control of multiple spacecraft via graphs, matrix inequalities, and switching,” AIAA J. Guidance, Control, and Dynamics, vol. 24, no. 2, pp. 369–377, 2001.

[4] S. Martinez, J. Cortes, and F. Bullo, “Motion coordination with distributed information,” IEEE Control Syst. Mag., vol. 27, no. 4, pp. 75–88, 2008.

[5] G. Arslan, J. R. Marden, and J. S. Shamma, “Autonomous vehicle-target assignment: a game theoretical formulation,” ASME J. Dynamic Systems, Measurement and Control, vol. 129, pp. 584–596, September 2007.

[6] B. J. Julian, M. Angermann, M. Schwager, and D. Rus, “Distributed robotic sensor networks: An information-theoretic approach,” Int. J. of Robotics Research, vol. 31, no. 10, pp. 1134–1145, 2012.

[7] A. Richards, T. Schouwenaars, J. How, and E. Feron, “Spacecraft trajectory planning with avoidance constraints using mixed-integer linear programming,” AIAA J. Guidance, Control, and Dynamics, vol. 25, no. 4, pp. 755–764, 2002.

[8] P. Kingston and M. Egerstedt, “Index-free multiagent systems: An Eulerian approach,” in 2nd IFAC Workshop on Distributed Estimation and Control in Networked Systems, (Annecy, France), pp. 215–220, 2010.

[9] P. K. Menon, G. D. Sweriduk, and K. D. Bilimoria, “New approach for modeling, analysis, and control of air traffic flow,” AIAA J. Guidance, Control, and Dynamics, vol. 27, no. 5, pp. 737–744, 2004.

[10] D. Mitulinić and P. Lima, “Modeling and optimal centralized control of a large-size robotic population,” IEEE Trans. Robotics, vol. 22, no. 6, pp. 1280–1285, 2006.

[11] C. C. Cheah, S. P. Hou, and J. J. E. Slotine, “Region-based shape control of a large-size robotic population,” IEEE Control Syst. Mag., vol. 27, no. 4, pp. 75–86, 2007.

[12] C. M. Cheah, S. P. Hou, and J. J. E. Slotine, “Region-based shape control of a swarm of robots,” Automatica, vol. 45, no. 10, pp. 2406–2411, 2009.

[13] S. Zhao, S. Ramakrishnan, and M. Kumar, “Density-based control of multiple robots,” in Amer. Control Conf. (San Francisco, USA), 2011.

[14] A. R. Mesquita, J. P. Hespanha, and K. Åström, “Omotactic: A stochastic multi-agent on site optimization procedure,” in Proc. Hybrid Systems: Computation and Control, 2008.
S. Berman, A. Halasz, M. A. Hsieh, and V. Kumar, “Optimized stochastic policies for task allocation in swarms of robots,” IEEE Trans. Robotics, vol. 25, no. 4, pp. 927–937, 2009.

I. Chattopadhyay and A. Ray, “Supervised self-organization of homogeneous swarms using ergodic projections of Markov chains,” IEEE Trans. Systems, Man, and Cybernetics, vol. 39, no. 6, pp. 1505–1515, 2009.

B. Ackimense and D. S. Bayard, “A Markov chain approach to probabilistic swarm guidance,” in Amer. Control Conf., pp. 6300–6307, June 2012.

S. Bandyopadhyay, S.-J. Chung, and F. Y. Hadaegh, “Inhomogeneous Markov chain approach to probabilistic swarm guidance algorithm,” in 5th Int. Conf. Spacecraft Formation Flying Missions and Technologies, (Munich, Germany), 2013.

R. Olfati-Saber and R. Murray, “Consensus problems in networks of agents with switching topology and time-delays,” IEEE Trans. Autom. Control, vol. 49, no. 9, pp. 1520 – 1533, 2004.

A. Jadbabaie, J. Lin, and A. S. Morse, “Coordination of groups of mobile autonomous agents using nearest neighbor rules,” IEEE Trans. Autom. Control, vol. 48, no. 6, pp. 988 – 1001, 2003.

L. Xiao and S. Boyd, “Fast linear iterations for distributed averaging,” Syst. Control Lett., vol. 53, pp. 65 – 78, 2004.

J. S. Shamma, Cooperative Control of Distributed Multi-Agent Systems. Chichester, West Sussex, England: John Wiley & Sons, 2007.

W. Ren and R. W. Beard, “Consensus seeking in multiagent systems under dynamically changing interaction topologies,” IEEE Trans. Autom. Control, vol. 50, pp. 655 – 661, May 2005.

S. Bandyopadhyay and S.-J. Chung, “Distributed estimation using Bayesian consensus filtering,” IEEE Trans. Autom. Control, 2013. under review (http://arxiv.org/abs/1403.3117).

D. P. Scharf, F. Y. Hadaegh, and S. R. Ploen, “A survey of spacecraft formation flying guidance and control (part I): Guidance,” in Proc. Amer. Control Conf., (Colorado, US), pp. 1733–1739, 2003.

D. Morgan, S.-J. Chung, and F. Y. Hadaegh, “Model predictive control of swarms of spacecraft using sequential convex programming,” AIAA J. Guidance, Control, and Dynamics, 2014. to appear.

D. Morgan, G. P. Subramanian, S. Bandyopadhyay, S.-J. Chung, and F. Y. Hadaegh, “Probabilistic guidance of swarms using sequential convex programming,” in IEEE/RSJ Inter. Conf. Intelligent Robots and Systems, 2014. submitted.

E. Torgerson, Comparison of Statistical Experiments. Cambridge University Press, 1991.

M. H. DeGroot, “Reaching a consensus,” J. Amer. Statistical Assoc., vol. 69, no. 345, pp. 688 – 704, 1960.

C. Genest and J. V. Zidek, “Combining probability distributions: A critique and an annotated bibliography,” Statistical Sci., vol. 1, no. 1, pp. 114 – 135, 1986.

R. Durrett, Probability: Theory and Examples. Thomson Brooks, 2005.

S. Cha, “Comprehensive survey on distance/similarity measures between probability density functions,” Int. J. Math. Models and Methods in Appl. Sci., vol. 1, no. 4, pp. 300–307, 2007.

R. A. Horn and C. R. Johnson, Matrix Analysis. Cambridge, England: Cambridge University Press, 1985.

D. Blackwell, “Finite non-homogeneous chains,” Ann. of Math., vol. 46, pp. 594−599, Oct. 1945.

E. Seneta, Non-negative Matrices and Markov Chains. New York, NY: Springer-Verlag, 2006.

P. Billingsley, Probability and Measure. New York: J. Wiley & Sons, 1995.

D. Morgan, S.-J. Chung, L. Blackmore, B. Ackimense, D. Bayard, and F. Y. Hadaegh, “Swarm-keeping strategies for spacecraft under J2 and atmospheric drag perturbations,” AIAA J. Guidance, Control, and Dynamics, vol. 35, no. 5, pp. 1492 – 1506, 2012.

K. Brumbaugh, “The metrics of spacecraft design reusability and cost analysis as applied to CubeSats,” Master’s thesis, The University of Texas at Austin, Austin, TX., 2012.

S. Chatterjee and E. Seneta, “Towards consensus: Some convergence theorems on repeated averaging,” J. Appl. Probability, vol. 14, no. 1, pp. 89 − 97, 1977.