2017 American Control Conference
Sheraton Seattle Hotel
May 24–26, 2017, Seattle, USA

# Parallel Probabilistic Swarm Guidance by Exploiting Kronecker Product Structures in Discrete-time Markov Chains

King Keung Wu[1], Yeung Yam[1], Helen Meng[2] and Mehran Mesbahi[3]

*Abstract*— **Discrete-time Markov chain has been a popular modeling paradigm for a wide range of applications. In this paper, we study the class of Markov chains that can be characterized by Kronecker products of transition matrices. Such Markov chains exhibit self-replicating property, which is common in both natural and artificial systems. We show that such chains can be decomposed into a system of parallel atomic Markov chains with fewer states. This feature enables analyzing these Markov chains by studying only the properties of smaller chains, that is useful for simplifying the overall analysis and even improving the computational efficiency. We propose an application of the Kronecker-structured Markov chains to probabilistic guidance problem, which is recently developed for decentralized swarm guidance. We then introduce the parallel probabilistic guidance algorithm, which serves as an improved version of the existing algorithms. We demonstrate that our algorithm is scalable for the class of target configurations having Kronecker product structures. Simulations are presented to demonstrate the effectiveness of the proposed method. Our work provides a guideline for the design of swarm configuration by exploiting the Kronecker structure to greatly enhance the computational efficiency.**

## I. INTRODUCTION

Kronecker product (KP) is an important matrix operation which can represent a large block matrix by two or more smaller factor matrices [1]. One remarkable feature is its "fractal"-like structure, which can represent the self-similarity or self-replicating properties appearing in both natural and artificial systems. Recently, this structure has been applied for modeling networks of complex systems that exhibit the hierarchical network structures [2] and also networked control systems [3]. There are some well-known connections between the theory of graphs and Markov chain (MC), such as the correspondence between the consensus protocol over a weighted digraph and the discrete-time evolution of a MC [4]. Hence, the study of KP structures in discrete-time MC (DTMC) is of great interest.

There is an extensive literature on incorporating KP within MCs, but they focus on the continuous-time MC with Kronecker structure described by the sum of KPs [5]–[7], which has a different formulation as the DTMC. In this paper, we propose a new approach to incorporate KPs in DTMCs that is different from the existing ones, by considering the transition matrix of a large MC system as KPs of multiple factor matrices with much fewer states. We show that some properties of the KPs allows the large MC to be separated into several parallel smaller MCs. This in turn provide the tools to analyze the large MC by only analyzing the smaller, atomic chains, which is often useful to simplify the analysis and improve computational efficiency.

To illustrate the effectiveness of the proposed method, we apply it on the probabilistic swarm guidance (PSG) as a case study. The PSG was originally proposed by Açıkmeşe and Bayard [8] as an MC based approach for coordination of autonomous swarm agents. The key idea is to drive the swarm to a target density distribution in the configuration space in decentralized sense. The problem can thus be formulated as synthesizing an appropriate Markov transition matrix given the desired target density distribution. Hence, no communication or collaboration is required among the agents. Besides, it allows automatically damage repairing once the desired distribution is attained, without extra sensing effort.

Açıkmeşe and Bayard proposed two approaches for synthesizing the Markov transition matrix: Metropolis-Hastings (MH) [8] and a linear matrix inequality (LMI) approach [9]. While the MH algorithm is easy to implement and scalable, it does not enable explicit control on convergence rate and fuel use. The LMI approach allows explicit control on the both, but it is not scalable when the number of bins is large.

The contribution of this paper to the probabilistic guidance problem is as follows. We aim to tackle the scalability problem of the LMI approach when the class of target spatial configuration has a self-replicating structure. We show that using the KP property of separating a large MC into smaller parallel ones with fewer states, it is possible to parallelize the LMI optimization into a system of smaller optimizations with much less variables. Therefore, the method is applicable even for extremely large MCs whose desired target distribution exhibits certain repeating properties. We also introduce a parallel probabilistic guidance algorithm (PPGA) for the swarm agents to decide their movements independently. The proposed approach can serve as a guideline for designing complex large target density distributions.

The paper is organized as follows: Section II reviews the basic background. Section III presents the proposed formulation of DTMCs with the KP structure. Section IV presents the application of the KP MC to PSG and introduces the proposed PPGA. Section V provides the simulation results of the proposed approach. Section VI concludes the paper

[1]King Keung Wu and Yeung Yam are with the Department of Mechanical and Automation Engineering, The Chinese University of Hong Kong, Hong Kong, China kkwu@mae.cuhk.edu.hk,yyam@mae.cuhk.edu.hk

[2]Helen Meng is with the Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong, Hong Kong, China hmmeng@se.cuhk.edu.hk

[3]Mehran Mesbahi is with the Department of Aeronautics and Astronautics, University of Washington, Seattle, WA, USA mesbahi@aa.washington.edu

978-1-5090-5992-8/$31.00 ©2017 AACC

and suggests directions for future research.

## II. BACKGROUND

This section provides the background for the proposed PPGA, including the review on MC, KP and the formulation of the PSG problem.

Here we define the notation used in this paper: $v_i$ or $[\mathbf{v}]_i$ denotes the $i$-th element of a column vector $\mathbf{v} \in \mathbb{R}^p$; $m_{ij}$ or $[\mathbf{M}]_{ij}$ denotes the element on the $i$-th row and $j$-th column; $\mathbf{1}_n$ denotes the vector of ones of dimension $n$; $\mathbf{I}_n$ denotes the identity matrix of dimension $n$-by-$n$; $\mathbf{A} \geq 0$ implies that all entries in $\mathbf{A}$ are non-negative; $\mathbf{Q} \succ (\succeq) 0$ implies that $\mathbf{Q}$ is positive (semi-)definite; $X_k$ denotes the state of a MC at time step $k$; $\odot$ denotes the element-wise multiplication; $\mathbf{a} = \mathbf{vec}(\mathbf{A})$ implies that $\mathbf{a}$ is a vector formed by stacking columns of $\mathbf{A}$; $\rho(\mathbf{A})$ is the spectral radius of $\mathbf{A}$, defined as the largest eigenvalue of $\mathbf{A}$; diag $(\mathbf{M})$ denotes a column vector of main diagonal elements of $\mathbf{M}$. $\|\mathbf{M}\|_F$ denotes the Frobenius norm of the matrix $\mathbf{M}$, which is defined by: $\|\mathbf{M}\|_F = \sqrt{\sum_{i=1}^{p_1}\sum_{j=1}^{q_1} m_{ij}^2}$.

### A. Discrete-time Finite Markov Chains

MC is a popular model for describing a discrete-time stochastic process where the future state depends only on the current state but not on the previous states [10]. The chain is a sequence $\{X_k : k \geq 0\}$ of random variables taking values in a finite state set $S = \{1, \ldots, n\}$. Thus, $X_k$ represents the state of the random development at the $k$-th time step. The MC satisfies the following property: $\mathbb{P}\{X_k = x_k | X_{k-1} = x_{k-1}, \ldots, X_1 = x_1\} = \mathbb{P}\{X_k = x_k | X_{k-1} = x_{k-1}\}$. For a *time-homogeneous* MC, where $\mathbb{P}\{X_{k+1} = j | X_k = i\} = \mathbb{P}\{X_k = j | X_{k-1} = i\}$ for all time steps $k \geq 0$, we define a transition matrix $\mathbf{P} \in [0,1]^{n \times n}$ with the element defined by $p_{ij} = \mathbb{P}\{X_{k+1} = j | X_k = i\}$, which is the probability for the state to transit from State $i$ to State $j$ in the next time step. Note that the transition matrix $\mathbf{P}$ is row-stochastic, where the sum of every row equals 1, that is, $\sum_j^n p_{ij} = 1$ for all $i$, and $0 \leq p_{ij} \leq 1$.

Let $\boldsymbol{\pi}(k) \in [0,1]^n$ denote the *distribution at time* $k$ where the $i$-th element is defined by $\pi_i(k) = \mathbb{P}\{X_k = i\}$. The distribution at time $k$ can be obtained by the following formula: $\boldsymbol{\pi}(k+1)^{\mathrm{T}} = \boldsymbol{\pi}(k)^{\mathrm{T}}\mathbf{P}$. Hence, $\boldsymbol{\pi}(k)^{\mathrm{T}} = \boldsymbol{\pi}(0)^{\mathrm{T}}\mathbf{P}^k$.

The *stationary distribution* $\boldsymbol{\pi}^*$ for the MC is defined as the distribution that satisfies

$$\boldsymbol{\pi}^{*\,\mathrm{T}} = \boldsymbol{\pi}^{*\,\mathrm{T}}\mathbf{P} \quad \text{or} \quad \boldsymbol{\pi}^* = \mathbf{M}\boldsymbol{\pi}^*, \tag{1}$$

where $\mathbf{M} = \mathbf{P}^{\mathrm{T}}$ and hence $m_{ji} = p_{ij}$. This notation will be used interchangeably in this paper.

A MC is *irreducible* if there exists a time step $k$ such that every entry in the $k$-step transition matrix is positive, that is, $\left[\mathbf{P}^k\right]_{ij} > 0$ for all $i, j \in \{1, \ldots, n\}$.

A distribution $\boldsymbol{\pi}^*$ is said to be a *steady-state distribution* of a MC, if for every initial distribution $\boldsymbol{\pi}(0)$, we have $\lim_{k\to\infty} \boldsymbol{\pi}(k) = \boldsymbol{\pi}^*$. It is known that for an *irreducible* MC is also *aperiodic*, if it has an *stationary distribution* $\boldsymbol{\pi}^*$, then it is also a *steady-state distribution*. In this paper, we focus on finite irreducible time-homogeneous MCs.

### B. Kronecker Product

The KP of matrices $\mathbf{A} \in \mathbb{R}^{p_1 \times q_1}$ and $\mathbf{B} \in \mathbb{R}^{p_2 \times q_2}$ is defined as the $p_1p_2$-by-$q_1q_2$ matrix,

$$\mathbf{A} \otimes \mathbf{B} = \begin{pmatrix} a_{11}\mathbf{B} & a_{12}\mathbf{B} & \cdots & a_{1,q_1}\mathbf{B} \\ a_{21}\mathbf{B} & a_{22}\mathbf{B} & \cdots & a_{2,q_1}\mathbf{B} \\ \vdots & \vdots & \ddots & \vdots \\ a_{p_1,1}\mathbf{B} & a_{p_1,2}\mathbf{B} & \cdots & a_{p_1,,q_1}\mathbf{B} \end{pmatrix} \tag{2}$$

We have the following mixed-product property [1]:

$$(\mathbf{A} \otimes \mathbf{B})(\mathbf{C} \otimes \mathbf{D}) = \mathbf{AC} \otimes \mathbf{BD} \tag{3}$$

where $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathbf{B} \in \mathbb{R}^{p \times q}$, $\mathbf{C} \in \mathbb{R}^{n \times r}$ and $\mathbf{D} \in \mathbb{R}^{q \times s}$.

### C. Probabilistic Swarm Guidance

The objective of PSG is to drive a large number of swarming agents to a prescribed region of the configuration space in a certain target density distribution based on probabilistic approach [8]. The idea is to assign the target density distribution of swarms as the *steady-state distribution* $\boldsymbol{\pi}^*$ of a MC, and then obtain the transition matrix of the MC such that the distribution from any initial distribution $\boldsymbol{\pi}(0)$ can converge to $\boldsymbol{\pi}^*$.

In the problem, the physical space $\mathcal{R}$ where the swarm distributed is partitioned into $n$ disjoint sub-regions $R_i$ for $i = 1, \ldots, n$ such that $\mathcal{R} = \bigcup_{i=1}^{n} R_i$ and $R_i \cap R_j = \emptyset$ for $i \neq j$. The sub-regions $R_i$ are usually referred as *bins*.

Consider that a swarm agent is located at $\mathbf{r}(k)$ at time step $k$, then let $\boldsymbol{\pi}$ be the probability vector whose $i$-th element $\pi_i$ represents the probability that the agent is in $x_i$ at time step $k$, $\pi_i(k) = \mathbb{P}\{\mathbf{r}(k) \in R_i\}$. Assume that a swarm is comprised of $N$ agents, where agent in the swarm acts statistically independent of each other. Let $\mathbf{r}_j(k)$ be the position of the $j$-th agent at time $k$, denote $\mathbf{x}(k)$ as the actual distribution of the $N$ agents at time step $k$. Thus, its $i$-th element $x_i(k)$ is the average number of agents in $R_i$ at time $k$, that is, $x_i(k) = \#\{\mathbf{r}_j(k) \in R_i\}/N$, where $\#\mathcal{A}$ denotes the cardinality of the set $\mathcal{A}$. When the number of agents is extremely large $N \to \infty$, then $x_i(k) \to \pi_i(k)$ for $i = 1, \ldots, n$, due to the law of large numbers.

The distribution guidance problem is thus formulated as follows: Suppose $N$ is very large; given any initial distribution $\mathbf{x}(0)$, we desire to guide the swarm toward a target steady-state distribution described by a probability vector $\boldsymbol{\pi}^*$, that is, $\lim_{k\to\infty} x_i(k) = \pi_i(k)$ for $i = 1, \ldots, n$. As the number of swarm agents is assumed to be large, we consider the swarm distribution at time $k$ is $\boldsymbol{\pi}(k)$ in the following formulation. Therefore, the goal is to synthesize a transition matrix $M \in [0,1]^{n \times n}$ such that

$$\boldsymbol{\pi}^* = \mathbf{M}\boldsymbol{\pi}^*. \tag{4}$$

Note that $\mathbf{M}$ is a column-stochastic matrix, and $\boldsymbol{\pi}(k+1) = \mathbf{M}\boldsymbol{\pi}(k)$ describes the time evolution of the probability vector $\boldsymbol{\pi}$. If the transition matrix $\mathbf{M}$ is obtained, as the movement of agents are independent from each other, the transition of each agent just follows the MC specified by the same transition matrix $\mathbf{M}$.

There are two broad approaches to synthesize the matrix $\mathbf{M}$. The first one is by applying Metropolis-Hastings algorithm, which is a Markov Chain Monte Carlo (MCMC) method for obtaining a sequence of random samples by propagating a special MC [8]. Although this approach is scalable and easy to implement, it lacks the explicit control on the convergence rate and explicit consideration of energy use by the swarm.

The second approach generates the matrix $\mathbf{M}$ with explicit control on convergence rate and energy use by solving a linear matrix inequality (LMI) problem [9]. The LMI is formulated as follows:

$$\begin{aligned} \min_{\mathbf{M},\mathbf{P}} \quad & \mathbf{1}^{\mathrm{T}}\left(\mathbf{1}-\operatorname{diag}\left(\mathbf{M}\right)\right) \\ \text{subject to} \quad & \mathbf{P}\succ 0,\ \mathbf{M}\geq \mathbf{0},\ \mathbf{1}^{\mathrm{T}}\mathbf{M}=\mathbf{1}^{\mathrm{T}}, \\ & \mathbf{M}\boldsymbol{\pi}^{*}=\boldsymbol{\pi}^{*},\ (\mathbf{1}\mathbf{1}^{\mathrm{T}}-\mathbf{A}^{\mathrm{T}})\odot\mathbf{M}=\mathbf{0} \text{ and} \\ & \begin{pmatrix} \lambda^{2}\mathbf{P} & (\mathbf{M}-\boldsymbol{\pi}^{*}\mathbf{1}^{\mathrm{T}})^{\mathrm{T}}\mathbf{G}^{\mathrm{T}} \\ \mathbf{G}(\mathbf{M}-\boldsymbol{\pi}^{*}\mathbf{1}^{\mathrm{T}}) & \mathbf{G}+\mathbf{G}^{\mathrm{T}}-\mathbf{P} \end{pmatrix}\succeq 0 \end{aligned} \tag{5}$$

where $\lambda\in[0,1)$ denotes the decay rate and $\mathbf{G}+\mathbf{G}^{\mathrm{T}}\succ 0$. If the solution $\mathbf{M}$ exists, the convergence of the corresponding MC is guaranteed as proved in [9]. Note that the formulation (5) considers a motion constraint specified by an adjacency matrix $\mathbf{A}\in\{0,1\}^{n\times n}$ whose element $A_{ij}$ indicates that the transition from bin $i$ to bin $j$ when it equals one, and zero otherwise. By choosing specific $\lambda$ and $\mathbf{G}$, the minimization problem (5) can be solved to obtain $\mathbf{M}$ that satisfies the requirement for probabilistic guidance. However, the size of $\mathbf{M}$ is determined by the number of bins $n$, which could be over thousands, depending on the spatial resolution. Hence, when $n$ is large, it is rather computationally expensive to solve the overarching LMI problem using the available numerical convex optimization packages.

The convergence of the obtained MC characterized by $\mathbf{M}$ is guaranteed by the following theorem:

**Theorem 1.** *(Theorem 2 in [9]) For any initial probability vector $\boldsymbol{\pi}(0)\in[0,1]^{n}$, it follows that $\lim_{k\to\infty}\boldsymbol{\pi}(t)=\boldsymbol{\pi}^{*}$ for the swarm system if and only if $\rho(\mathbf{M}-\boldsymbol{\pi}^{*}\mathbf{1}^{\mathrm{T}})<1$.*

It is shown in [9] that the last inequality constraint in (5) is equivalent to $\rho(\mathbf{M}-\boldsymbol{\pi}^{*}\mathbf{1}^{\mathrm{T}})<1$. Hence, the solution of (5) guarantees the convergence of swarm to the desired distribution, regardless of the initial distribution.

## III. DISCRETE-TIME MARKOV CHAINS WITH KRONECKER PRODUCT STRUCTURE

In this section, we present the properties of discrete-time finite MCs having a KP structure.

### A. Formulation and Interpretation

Consider two irreducible MCs which are denoted as $\alpha_1=\{X_k : k=0,1,2,\ldots\}$ and $\alpha_2=\{Y_k : k=0,1,2,\ldots\}$. Let the finite state sets of $\alpha_1$ and $\alpha_2$ be $S_1=x_1,\ldots,x_{n_1}$ and $S_2=y_1,\ldots,y_{n_2}$ respectively. We assume that the corresponding transition matrices $\mathbf{P}_1\in[0,1]^{n_1\times n_1}$ and $\mathbf{P}_2\in[0,1]^{n_2\times n_2}$ of $\alpha_1$ and $\alpha_2$ are given, where $[\mathbf{P}_1]_{ij}=\mathbb{P}\{X_{k+1}=x_j|X_k=x_i\}$ and $[\mathbf{P}_2]_{ij}=\mathbb{P}\{Y_{k+1}=y_j|Y_k=y_i\}$. Now consider another MC $\beta=\{Z_k : k=0,1,2,\ldots\}$ with the state set $S$ as the Cartesian product of the states sets of $\alpha_1$ and $\alpha_2$, that is, $S=S_1\times S_2$. Note that $S$ is finite with $n_1n_2$ elements. Now, we aim to combine the two MCs $\alpha_1$ and $\alpha_2$ to generate $\beta$ by assuming the random variables $X_{k+1}$ and $Y_{k+1}$ only depend on $X_k$ and $Y_k$, respectively, that is,

$$\begin{aligned} \mathbb{P}\{Z_{k+1}|Z_k\} &= \mathbb{P}\{(X_{k+1},Y_{k+1})\,|(X_k,Y_k)\} \\ &= \mathbb{P}\{X_{k+1}|X_k\}\mathbb{P}\{Y_{k+1}|Y_k\}. \end{aligned} \tag{6}$$

The corresponding transition matrix $\mathbf{P}$ of $\beta$ can actually be generated by taking the KP of $\mathbf{P}_1$ and $\mathbf{P}_2$, i.e., $\mathbf{P}=\mathbf{P}_1\otimes\mathbf{P}_2$. Note that $\mathbf{P}$ is also a row-stochastic matrix. The above formulation can be easily extended to KPs of multiple transition matrices, and similar arguments are thus applicable in combining multiple MCs.

### B. Basic Properties

Here we state certain properties of DTMCs with KP structures. Given an irreducible MC $\alpha=\{X_k : k=0,1,2,\ldots\}$ characterized by transition matrix $\mathbf{P}=\mathbf{P}_1\otimes\cdots\otimes\mathbf{P}_N$, where $\mathbf{P}_1,\ldots,\mathbf{P}_N$ are the transition matrices corresponding to $N$ different irreducible MCs $\alpha_1,\ldots,\alpha_N$ with smaller number of states.

The following proposition shows that the evolution of the corresponding distribution can be decomposed into that of its factors. We assume the transition matrix is given as $\mathbf{P}=\mathbf{P}_1\otimes\cdots\otimes\mathbf{P}_N$.

**Proposition 1.** (Evolution of distribution)
If the initial distribution of the MC $\alpha$ can be represented by $\boldsymbol{\pi}(0)=\boldsymbol{\pi}_1(0)\otimes\cdots\otimes\boldsymbol{\pi}_N(0)$, where $\boldsymbol{\pi}_1(0),\ldots,\boldsymbol{\pi}_N(0)$ are the initial distributions of the MC $\alpha_1,\ldots,\alpha_N$ with transition matrix $\mathbf{P}_1,\ldots,\mathbf{P}_N$, respectively. Then $\boldsymbol{\pi}(k)=\boldsymbol{\pi}_1(k)\otimes\cdots\otimes\boldsymbol{\pi}_N(k)$.

*Proof.* Given $\boldsymbol{\pi}(0)=\boldsymbol{\pi}_1(0)\otimes\cdots\otimes\boldsymbol{\pi}_N(0)$, as we have $\boldsymbol{\pi}_i(k)^{\mathrm{T}}=\boldsymbol{\pi}_i(0)^{\mathrm{T}}\mathbf{P}_i^{k}$ for $i=1,\ldots,N$, hence by the KP property in (3),

$$\begin{aligned} \boldsymbol{\pi}^{\mathrm{T}}(k) &= \boldsymbol{\pi}^{\mathrm{T}}(0)\mathbf{P}^{k} \\ &= \left(\boldsymbol{\pi}_1^{\mathrm{T}}(0)\otimes\cdots\otimes\boldsymbol{\pi}_N^{\mathrm{T}}(0)\right)\left(\mathbf{P}_1\otimes\cdots\otimes\mathbf{P}_N\right)^{k} \\ &= \left(\boldsymbol{\pi}_1^{\mathrm{T}}(0)\mathbf{P}_1^{k}\right)\otimes\cdots\otimes\left(\boldsymbol{\pi}_N^{\mathrm{T}}(0)\mathbf{P}_N^{k}\right) \\ &= \boldsymbol{\pi}_1^{\mathrm{T}}(k)\otimes\cdots\otimes\boldsymbol{\pi}_N^{\mathrm{T}}(k). \end{aligned}$$

□

**Proposition 2.** (Stationary and steady-state distribution)
If there exist stationary (steady-state) distributions $\boldsymbol{\pi}_1^{*},\ldots,\boldsymbol{\pi}_N^{*}$ for the MC $\alpha_1,\ldots,\alpha_N$, where $\boldsymbol{\pi}_i^{*\,\mathrm{T}}=\boldsymbol{\pi}_i^{*\,\mathrm{T}}\mathbf{P}_i$, for $i=1,\ldots,N$, then $\boldsymbol{\pi}^{*}=\boldsymbol{\pi}_1^{*}\otimes\cdots\otimes\boldsymbol{\pi}_N^{*}$ is the stationary (steady-state) distribution of the MC $\alpha$.

*Proof.* It can be shown that

$$\begin{aligned} &\left(\boldsymbol{\pi}_1^{*\,\mathrm{T}}\otimes\cdots\otimes\boldsymbol{\pi}_N^{*\,\mathrm{T}}\right)=\left(\boldsymbol{\pi}_1^{*\,\mathrm{T}}\mathbf{P}_1\right)\otimes\cdots\otimes\left(\boldsymbol{\pi}_N^{*\,\mathrm{T}}\mathbf{P}_N^{k}\right) \\ &=\left(\boldsymbol{\pi}_1^{*\,\mathrm{T}}\otimes\cdots\otimes\boldsymbol{\pi}_N^{*\,\mathrm{T}}\right)\left(\mathbf{P}_1\otimes\cdots\otimes\mathbf{P}_N\right). \end{aligned}$$

Hence, $\boldsymbol{\pi}^{*}=\boldsymbol{\pi}_1^{*}\otimes\cdots\otimes\boldsymbol{\pi}_N^{*}$ is the stationary distribution. Similarly, a parallel statement holds for the steady-state

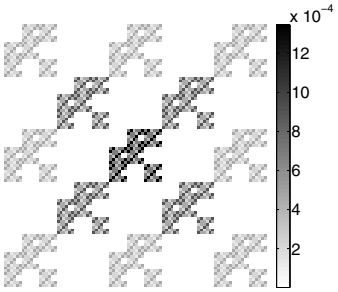



Fig. 1: An example of target spatial configuration with self-replicating structures in $90 \times 90$ bins. The color indicates the relative density, where the density sum over all bins is one.

distribution, as if the steady-state distribution exists, it must also be a stationary distribution. □

Note that Proposition 2 reveals a non-trivial property: when all the Markov sub-chains $\alpha_1, \ldots, \alpha_N$ have steady-state distributions, the MC $\alpha$ will also has a steady-state distribution. By definition, if $\boldsymbol{\pi}^*$ is the steady-state distribution of MC $\alpha$, the distribution will eventually converge to $\boldsymbol{\pi}^*$ from *any* initial distribution $\boldsymbol{\pi}(0)$ which is not necessarily KP of $\boldsymbol{\pi}_1(0), \ldots, \boldsymbol{\pi}_N(0)$. Hence, it is independent of Proposition 1 where $\boldsymbol{\pi}(0) = \boldsymbol{\pi}_1(0) \otimes \cdots \otimes \boldsymbol{\pi}_N(0)$. This property is essential to guarantee the convergence of the proposed parallel swarm guidance algorithm.

## IV. PROBABILISTIC SWARM GUIDANCE VIA KRONECKER PRODUCT MARKOV CHAINS

In this section, we apply the proposed MC framework with the KP structure in Section III for the PSG. As mentioned in Section II-C, this problem is computationally challenging for the LMI approach when the number of bins is large, which may happen when the configuration space is large and is partitioned in high resolution. We aim to reduce the computational cost for solving the corresponding LMI problem by exploring a class of swarm configurations that exhibit the self-replicating structures with the aid of KPs. Self-replicating swarm behaviours and patterns have widely been observed in natural animal communities. Therefore, it is a natural choice for the design of swarm configuration. We show that if the target configuration in swarm guidance problem exhibits repeating patterns, we can apply the separation property of KP MC to break the large system into smaller chains and synthesize the transition matrix $\mathbf{M}$ by synthesizing for the smaller subsystems. This on the other hand can greatly reduce the computational cost. In other words, it is desirable to design the configuration such that it possesses as many repeating features in configuration space as possible.

### A. An Illustrative Example

We show a 2-dimensional example of swarm configuration with self-replicating structures in Fig. 1. Each pixel represents a bin. It is obvious to see the repeating patterns. The colors represent the relative densities of the bins, where the sum of densities over all bins is one. This configuration is actually generated by the KPs of four matrices,

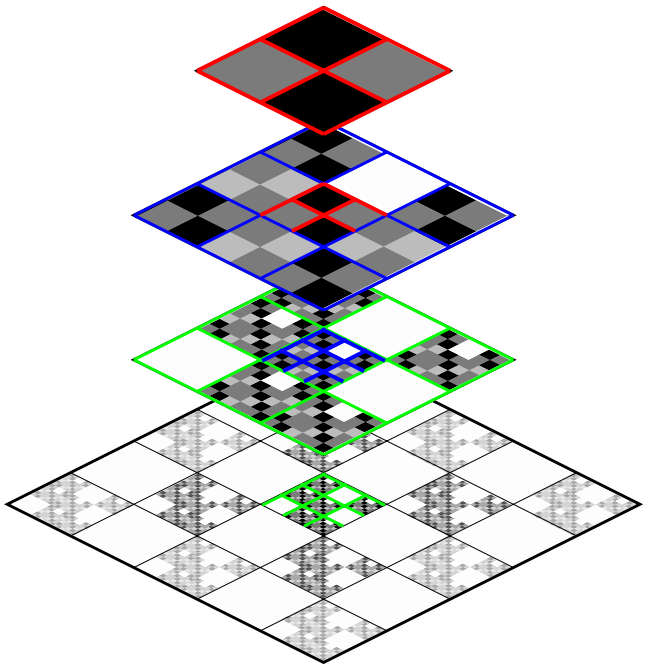


Fig. 2: An illustration on how the KP structures in space corresponding to multi-resolutions. The bins can be decomposed into four layers, represented by $\mathbf{X}^{(1)} \in [0,1]^{5\times5}$, $\mathbf{X}^{(2)} \in [0,1]^{3\times3}$, $\mathbf{X}^{(3)} \in [0,1]^{3\times3}$ and $\mathbf{X}^{(4)} \in [0,1]^{2\times2}$ respectively, from bottom to top.

$$\mathbf{X}^{(1)} = \frac{1}{19}\begin{pmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 2 & 0 & 2 & 0 \\ 1 & 0 & 3 & 0 & 1 \\ 0 & 2 & 0 & 2 & 0 \\ 1 & 0 & 1 & 0 & 1 \end{pmatrix}, \quad \mathbf{X}^{(2)} = \frac{1}{6}\begin{pmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix},$$

$$\mathbf{X}^{(3)} = \frac{1}{13}\begin{pmatrix} 2 & 1 & 2 \\ 1 & 2 & 0 \\ 2 & 1 & 2 \end{pmatrix} \quad \text{and} \quad \mathbf{X}^{(4)} = \frac{1}{6}\begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix},$$

of which the sums of all entries equal to one. Therefore, one can think of the matrices as the density distributions in different scales. The KP demonstrates the power of capturing the multi-scaling properties inside a configuration. In general, the matrices can be of arbitrary size, and do not necessarily be square. The total number of bins $N$ is 8100, and thus the size of transition matrix $\mathbf{M}$ is $8100 \times 8100$, which is too large for solving the LMI problem in (5). We will show that the size of the LMI problem can be greatly reduced by exploiting the KP structure.

Fig. 2 illustrates how the configuration in Fig. 1 can be decomposed into layers of smaller local configurations by KPs. In the figure, the bottom layer is divided into $5 \times 5$ partitions, and each partition is a weighted copy of the upper layer. Hence, the upper layer represents the repeating local features. Each layer can be expressed by a matrix of the weights, which are values corresponding to the probability densities of finding a swarm agent in the bins.

### B. General Case

Consider the general case where a configuration $\mathbf{X} \in [0,1]^{p_1 \cdots p_L \times q_1 \cdots q_L}$ is composed of the KPs of $L$ matrices, $\mathbf{X}^{(i)} \in [0,1]^{p_i \times q_i}$ for $i = 1, \ldots, L$, that is, $\mathbf{X} = \mathbf{X}^{(1)}, \ldots, \mathbf{X}^{(L)}$, where the sum of all entries in $\mathbf{X}^{(i)}$ is one. It is obvious that the sum of all entries in matrix $\mathbf{X}$ is also one. Let $\mathbf{x}^{(i)} \in [0,1]^{p_i q_i}$ be the vector generated by stacking the columns of $\mathbf{X}^{(i)}$, i.e. $\mathbf{x}^{(i)} = \mathbf{vec}(\mathbf{X}^{(i)})$, that becomes a density vector of layer $i$. Each element can be given an identity number according to its index. Hence, each bin in $\mathbf{X}$ can be labelled by a tuple of $L$ indexes, each corresponding to one layer. For example, the bin at $X_{9,7}$ in Fig. 1 is labelled as $(1,2,2,1)$. However note that in general $\mathbf{x} \neq \mathbf{x}^{(1)} \otimes \cdots \otimes \mathbf{x}^{(L)}$ where $\mathbf{x} = \mathbf{vec}(\mathbf{X})$. In the meantime, the product can be a permutation of $\mathbf{x}$, i.e., $\mathcal{P}\mathbf{x} = \mathbf{x}^{(1)} \otimes \cdots \otimes \mathbf{x}^{(L)}$, where $\mathcal{P}$ is certain permutation matrix. As the re-ordering labels of bins or the entries in the density vector does not affect the physical meaning in the probabilistic guidance problem, we will use this fact to develop a parallel algorithm for the probabilistic guidance.

The following provides the definition of *Kronecker-structured density vector*:

**Definition 1.** (Kronecker-structured density vector)
We call a density vector $\mathbf{x}$ the *Kronecker-structured density vector*, if there exists a permutation $\mathcal{P}$ that satisfies $\mathcal{P}\mathbf{x} = \mathbf{x}^{(1)} \otimes \cdots \otimes \mathbf{x}^{(L)}$, and the corresponding vectors $\mathbf{x}^{(i)}$ for $i = 1, \ldots, L$ is called the *factor density vectors*.

### C. Parallel Probabilistic Guidance Algorithm

We propose a parallel probabilistic guidance algorithm based on the properties of KP structure. Let $\boldsymbol{\pi}$ be the target configuration of swarm which is a *Kronecker-structured density vector*, $\mathcal{P}\boldsymbol{\pi} = \boldsymbol{\pi}^{(1)} \otimes \cdots \otimes \boldsymbol{\pi}^{(L)}$ for a certain permutation matrix $\mathcal{P}$ and a finite integer $L > 1$. The key idea of parallel probabilistic guidance is to find the MC with column-stochastic transition matrix $\mathbf{M}$ which can be expressed as $\mathbf{M} = \mathbf{M}_1 \otimes \ldots \otimes \mathbf{M}_L$.

The following theorem lays the foundation for our parallel algorithm. For simplicity, we will use $\boldsymbol{\pi}$ instead of $\boldsymbol{\pi}^*$ to represent the steady-state distribution which is also the target swarm distribution when it does not cause any ambiguity.

**Theorem 2.** *Given a Kronecker-structured density vector $\boldsymbol{\pi}$ where $\mathcal{P}\boldsymbol{\pi} = \boldsymbol{\pi}^{(1)} \otimes \cdots \otimes \boldsymbol{\pi}^{(L)}$ with certain permutation matrix $\mathcal{P}$ and a finite integer $L > 1$. If there exists matrices $\mathbf{M}_i$ that satisfy*

$$\mathbf{M}_i \boldsymbol{\pi}^{(i)} = \boldsymbol{\pi}^{(i)},\ \mathbf{M}_i \geq 0 \text{ and } \mathbf{1}^{\mathrm{T}} \mathbf{M}_i = \mathbf{1}^{\mathrm{T}} \tag{7}$$

*for $i = 1, \ldots, L$, then the matrix $\mathbf{M} = \mathbf{M}_1 \otimes \cdots \otimes \mathbf{M}_L$ must satisfy*

$$\mathbf{M}\mathcal{P}\boldsymbol{\pi} = \mathcal{P}\boldsymbol{\pi},\ \mathbf{M} \geq 0 \text{ and } \mathbf{1}^{\mathrm{T}} \mathbf{M} = \mathbf{1}^{\mathrm{T}}. \tag{8}$$

*Proof.* Note that

$$\begin{aligned} &\mathbf{M}\mathcal{P}\boldsymbol{\pi} = (\mathbf{M}_1 \otimes \cdots \otimes \mathbf{M}_L) \left( \boldsymbol{\pi}^{(1)} \otimes \cdots \otimes \boldsymbol{\pi}^{(L)} \right) \\ &= \mathbf{M}_1 \boldsymbol{\pi}^{(1)} \otimes \cdots \otimes \mathbf{M}_L \boldsymbol{\pi}^{(L)} = \boldsymbol{\pi}^{(1)} \otimes \cdots \otimes \boldsymbol{\pi}^{(L)} = \mathcal{P}\boldsymbol{\pi} \end{aligned}$$

and it is well-known that KPs of column-stochastic matrix is also column-stochastic. □

Based on Theorem 2, we propose a parallel LMI formulation to solve the guidance problem. The idea is based on [9]: Given a *Kronecker-structured density vector* $\boldsymbol{\pi}$, where $\mathcal{P}\boldsymbol{\pi} = \boldsymbol{\pi}^{(1)} \otimes \cdots \otimes \boldsymbol{\pi}^{(L)}$, we synthesize $\mathbf{M}_1, \ldots, \mathbf{M}_L$ by solving the following optimization problem:

$$\begin{aligned} &\min_{\mathbf{M}_i, \mathbf{P}_i} \ \mathbf{1}^{\mathrm{T}} \left( \mathbf{1} - \mathrm{diag}\left( \mathbf{M}_i \right) \right) \\ &\text{subject to } \ \mathbf{P}_i \succ 0,\ \mathbf{M}_i \geq \mathbf{0},\ \mathbf{1}^{\mathrm{T}} \mathbf{M}_i = \mathbf{1}^{\mathrm{T}}, \mathbf{M}_i \boldsymbol{\pi}^{(i)} = \boldsymbol{\pi}^{(i)}, \\ &\text{and } \begin{pmatrix} \lambda^2 \mathbf{P}_i & (\mathbf{M}_i - \boldsymbol{\pi}^{(i)} \mathbf{1}^{\mathrm{T}})^{\mathrm{T}} \mathbf{G}_i^{\mathrm{T}} \\ \mathbf{G}_i (\mathbf{M}_i - \boldsymbol{\pi}^{(i)} \mathbf{1}^{\mathrm{T}}) & \mathbf{G}_i + \mathbf{G}_i^{\mathrm{T}} - \mathbf{P}_i \end{pmatrix} \succeq 0 \end{aligned} \tag{9}$$

where $\lambda \in [0,1)$ denotes the decay rate and $\mathbf{G}_i + \mathbf{G}_i^{\mathrm{T}} \succ 0$, for $i = 1, \ldots, L$.

Note that the above LMI problem is the parallel version of (5) without the motion constraint. If the adjacency matrix $\mathbf{A}$ that characterizes the motion constraint also has the similar KP structure as the transition matrix $\mathbf{M}$, i.e., $\mathbf{A} = \mathbf{A}_1 \otimes \cdots \otimes \mathbf{A}_L$ where $\mathbf{A}_i \in \{0,1\}^{p_i q_i \times p_i q_i}$ has the same size as $\mathbf{M}_i$, then the motion constraints characterized by $\mathbf{A}_i$ can also fit into the above system (9). However, we only concern the case without motion constraint in this paper for simplicity.

We describe the parallel probabilistic guidance algorithm (PPGA) as follows. As the Markov sub-chains generated by the above optimization problems are independent from each other, we can consider them as parallel MCs so that we do not need to explicitly compute or store the large transition matrix $\mathbf{M}$. Each agent has a copy of all the factor transition matrices $\mathbf{M}_i$ for $i = 1, \ldots, L$. The swarm then propagates its position based on the following Markov process. Note that every bin is labelled by a tuple of length $L$, with each element representing the position in terms of the corresponding layer. First, each agent detects its current bin, labelled as $(i_1, \ldots, i_L)$. Then, each agent generates vector of random numbers $\mathbf{z} \in [0,1]^L$ whose values are uniformly distributed in $[0,1]$. Then, each agent moves to bin labelled as $(j_1, \ldots, j_L)$ if

$$\sum_{k=1}^{j_l - 1} M_{k, i_l} \leq z_l \leq \sum_{k=1}^{j_l} M_{k, i_l} \tag{10}$$

for $l = 1, \ldots, L$. Obviously, the overall convergence rate will be limited by the layer that converges with the slowest speed.

Note that if the solutions of (9) exist, Theorem 1 ensures the convergence of all the sub-chains to the desired *steady-state distributions* . Hence, by Proposition 2, the MC with transition matrix $\mathbf{M} = \mathbf{M}_1 \otimes \cdots \otimes \mathbf{M}_L$ also converges to the *steady-state distribution* which guarantees the convergence of the proposed PPGA.

## V. NUMERICAL EXPERIMENTS

In this section, we present the computational simulation results to demonstrate the efficacy of the proposed algorithm. We use the spatial configuration in Fig. 1 as input, which

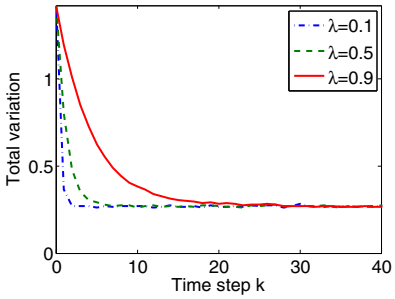
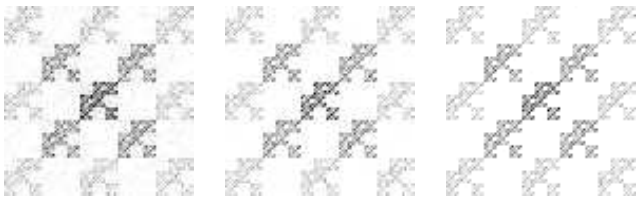



Fig. 3: The plot showing the changes in total variation for $\lambda = 0.1, 0.5, 0.9$, respectively.

(a) $k = 0$ (b) $k = 2$ (c) $k = 4$



(d) $k = 8$ (e) $k = 16$ (f) $k = 32$

Fig. 4: Evolution of the distribution and the target configuration

has 8,100 bins in total. The swarm contains $N = 20000$ autonomous agents that are guided by the PPGA to form a distribution sequence over time. The swarm is distributed uniformly across the space at time step $k = 0$. We solved the optimization problem in (9), and each agent moves independently based on the PPGA. We apply *total variation* $T(k)$ at time $k$ to monitor the convergence: $T(k) = \sum_j^{pq} |x_j(k) - \pi_j|$ where $p = p_1 \cdots p_L$ and $q = q_1 \cdots q_L$.

Fig. 3 shows the total variations change in time for different $\lambda$ settings. It reveals that the selection of $\lambda$ actually affects the convergence rate. Smaller $\lambda$ leads to faster convergence. For $\lambda = 0.9$, the distribution converges at around $k = 20$.

The evolution of the distribution when $k = 0, 2, 4, 8, 16, 32$ for $\lambda = 0.9$ is shown in Fig. 4. It reveals that the convergence is in parallel across different layers. The simulation is run on a personal computer; it takes less than a minute to solve the corresponding LMI problems. However, if we try to solve the original optimization in (5) which has 65,610,000 variables for $\mathbf{M}$, it is impossible for a personal computer to solve it due to the limitation on memory and computational power. Our method can even lead to a parallel computation framework, where each optimization can be solved separately in different cores of computing units.

## VI. CONCLUSIONS

This paper explored the power of Kronecker product (KP) for the analysis and synthesis of discrete-time Markov chains (MC). In particular, we studied the MC that has KP structure and showed that it is equivalent to a system of parallel MCs with less number of states. We then demonstrated the advantages of using KP-structured MC on the application of probabilistic swarm guidance by greatly reducing the computational cost of finding the guidance rule for the target configuration that exhibits repeating spatial patterns. In addition, we proposed the parallel probabilistic guidance algorithm that can drive the agents independently to the desired swarm distribution. The search for suitable guidance rule can be done off-line using a personal computer with inexpensive configuration. The pre-computed rule is then implemented to the swarm agents. This rule preserves the essence of the original probabilistic guidance idea proposed by Açıkmeşe and Bayard in [8]. Here we focused on the design perspective where the target configuration is chosen to possess the repeating patterns. It is worthwhile to study the appropriate approximation of certain desired density distribution that is not exactly Kronecker-structured but also exhibits self-replicating patterns. Moreover, incorporating motion constraints to the proposed hierarchical framework will also be studied in our future work.

## ACKNOWLEDGMENT

The research of K.K. Wu and H. Meng is affiliated with the Stanley Ho Big Data Decision Analytics Research Centre of The Chinese University of Hong Kong. The research of M. Mesbahi has been supported by U.S. Army Research Office grant W911NF-13-1-0340 and U.S. Office of Naval Research grant N00014-12-1-1002. The authors acknowledge the discussions with Behçet Açıkmeşe on probabilistic guidance, Airlie Chapman, and the RAIN Laboratory at the University of Washington for hosting the first author.

## REFERENCES

[1] C. F. Van Loan, "The ubiquitous Kronecker product," *Journal of Computational and Applied Mathematics*, vol. 123, pp. 85–100, 2000.
[2] J. Leskovec, D. Chakrabarti, J. Kleinberg, C. Faloutsos, and Z. Ghahramani, "Kronecker graphs: An approach to modeling networks," *The Journal of Machine Learning Research*, vol. 11, pp. 985–1042, 2010.
[3] A. Chapman and M. Mesbahi, "Kronecker product of networked systems and their approximates," in *21st International Symposium on Mathematical Theory of Networks and Systems*, 2014, pp. 1426–1431.
[4] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods in Multiagent Networks*. Princeton University Press, 2010.
[5] T. Dayar, *Analyzing Markov Chains using Kronecker Products: Theory and Applications*. Springer Science & Business Media, 2012.
[6] ——, "Analyzing Markov chains based on Kronecker products," *MAM*, pp. 279–300, 2006.
[7] Y. Bao, İ. N. Bozkurt, T. Dayar, X. Sun, and K. S. Trivedi, "Decompositional analysis of Kronecker structured Markov chains," *Electronic Transactions on Numerical Analysis*, vol. 31, pp. 271–294, 2008.
[8] B. Açikmeşe and D. S. Bayard, "A Markov chain approach to probabilistic swarm guidance," in *2012 American Control Conference (ACC)*. IEEE, 2012, pp. 6300–6307.
[9] B. Açıkmeşe and D. S. Bayard, "Markov chain approach to probabilistic guidance for swarms of autonomous agents," *Asian Journal of Control*, vol. 17, no. 4, pp. 1105–1124, 2015.
[10] J. R. Norris, *Markov Chains*. Cambridge University Press, 1998.