Robust Metropolis-Hastings Algorithm for Safe Reversible Markov Chain Synthesis

Mahmoud El Chamie

Abstract—This paper presents a new method to synthesize safe reversible Markov chains via classical Metropolis-Hastings (M-H) algorithm. Classical M-H algorithm does not impose safety constraints on the probability vector for the resulting Markov chain. This paper presents a new M-H algorithm for Markov chain synthesis that handles safety constraints while ensuring reversibility and a desired stationary (steady-state) distribution. Specifically, we provide a convex synthesis algorithm that incorporates the safety constraints via proper choice of the proposal matrix for the M-H algorithm. The resulting proposal matrix is a stochastic matrix that ensures safety for a nominal stationary distribution. Then it is shown that the M-H algorithm with this proposal matrix, robust M-H algorithm, also ensures safety for a well-characterized convex set of stationary distributions, which also includes the nominal stationary distribution. We also present a convex synthesis method for the proposal matrix to maximize the size of this resulting set of feasible stationary distributions for the robust M-H algorithm. Simulation results are also provided to demonstrate that there is no tradeoff between the speed of convergence and the robustness.

I. INTRODUCTION

The Metropolis-Hasting (M-H) algorithm [1], [2], [3] is a method for obtaining random samples from a probability distribution when direct sampling is difficult. The M-H algorithm builds on the theory of Markov processes, Markov-Chain-Monte-Carlo sampling methods [4], [5], [6], [7], [8], [9], graph theory [10], [11], and Lyapunov stability analysis. It is also aided by research in designing fast mixing Markov chains that incorporate constraints on transition probabilities [12], [13], [14]. Given a matrix $K$ (called proposal matrix), this algorithm can be used to design the stochastic transition matrix $M$ of a Markov chain to satisfy some system specifications (e.g., a prescribed stationary distribution). The matrix $M$ inherits the key properties of the proposal matrix $K$, such as speed of convergence, while satisfying the system specifications. Thus the M-H is very useful for applications when online Markov chain synthesis is needed because it can be implemented easily and executed very efficiently on real-time processors. However, currently, the M-H algorithm cannot impose safety constraints on the probability distribution vector, i.e., upper bound constraints on the probability distribution. Such constraints appear naturally in many applications, e.g., to bound/minimize the probability of conflicts/collisions among agents in randomized motion planning of swarms [15], which may arise due to clustering during transition to the desired steady-state distribution. These hard constraints on the probability vector of the Markov chain are often referred to as safety constraints [16], [17].

Markov chain propagates the probability distribution (discrete) vector $\mathbf{x}(t) \in \mathbb{R}^m$ as follows,

$$\mathbf{x}(t+1) = M(t)\mathbf{x}(t) \quad t = 0, 1, 2, \ldots$$

(1)

where $M(t)$ is a column stochastic matrix for all time; hence $\mathbf{x}(t)$ stays normalized as $\mathbf{1}^T\mathbf{x}(t) = 1$ for all $t \geq 0$ where $\mathbf{1}$ is the vector of all ones. In many applications, it is desired to design the column stochastic matrix $M$ to satisfy some specifications. For example, in randomized motion planning (RMP) [18], [19], [15], $\mathbf{x}(t)$ describes the probability of an agent (e.g., vehicle) to be in a certain state and $M(t)$ determines the probability distributions for possible transitions. In gossiping and wireless sensor networks [20], [21], (1) is the dynamics for the evolution of an estimate for a relevant physical quantity as temperature, pressure, etc. In voting models [22], $\mathbf{x}(t)$ determines the preference of a group of people towards a given object of interest (e.g., application, leader, etc.). In belief propagation [23], (1) is a model for information dissemination in social networks. In consensus protocols [24], [25], the transition matrix (called weight matrix in the context of consensus protocols) is designed for fast convergence among networked agents for the value of a quantity of interest.

When $\mathbf{x}(t)$ is a discrete probability vector for an underlying Markov chain governed by equation (1) dynamics, $\mathbf{x}(t)$ would satisfy some constraints naturally due to column stochasticity of $M(t)$ such as $\mathbf{x}(t) \geq 0$ and $\mathbf{1}^T\mathbf{x}(t) = 1$ for all $t = 0, 1, \ldots$. There can also be additional constraints characterized by hard safety upper bounds on the probability vector, i.e.,

$$\mathbf{x}(t) \leq \mathbf{d} \quad \text{for all } t \geq 0,$$

(2)

where $\mathbf{d} \in \mathbb{R}^m$ is a constant non-negative vector.

The purpose of this paper is to expand the M-H algorithm for the generation of a transition matrix $M(t)$ that is both convergent and incorporates safety constraints for time-varying system specifications (i.e., time-varying desired stationary distributions). In summary, our main contributions are: (i) Providing a new robust LP-based method to synthesize the proposal matrix of the M-H algorithm to satisfy safety constraints of the form given by (2); (ii) Providing an efficient M-H algorithm, generating safe Markov chains, that can handle time-varying desired stationary probability distributions; (iii) Studying the speed of convergence of the resulting transition matrix constructed by the robust
M-H algorithm in the presence of safety constraints. It is important to note that our earlier results [17] can synthesize constant Markov chains with all the constraints considered in this paper via convex optimization methods. The main contribution of this paper is obtaining feasible synthesis via the M-H algorithm, where the proposal matrix is designed such that the resulting Markov chain satisfies the safety constraints for a well-characterized set of stationary distributions. This allows us to quickly adapt to time-varying stationary distribution specifications without recomputing the proposal matrix, which is a very useful property for a number of applications where this adaptation must happen in real-time, see [26] for an example application.

II. FORMULATION OF REVERSIBLE MARKOV CHAIN 
SYNTHESIS WITH SAFETY CONSTRAINTS

A. Notation

In this paper, small bold letters in general signifies a vector (e.g., \( v \) whose elements are indicated as \( v_1, v_2, \ldots \)), capital letters are in general matrices (e.g., \( M \) whose \( i \)-th row \( j \)-th column element is denoted by \( M_{ij} \)). \( 0 \) and \( 1 \) are the matrices/vectors of conformable dimensions with all zeros and ones respectively; \( e_i \) is a vector with its \( i \)-th entry \( +1 \) and others zero; \( I \) is the identity matrix; \( x_t = e_i^T x \) and \( X_{ij} = e_i^T X e_j; Q = Q^T \geq (\geq) 0 \) is a symmetric positive (semi-)definite matrix; \( R \geq (\geq) H \) implies that \( R_{ij} \geq (\geq) H_{ij} \) for all \( i, j \); \( v \) is a probability vector if \( v \geq 0 \) and \( 1^T v = 1 \); \( m \) is the set of probability \( m \times 1 \) vectors, i.e., \( v \geq 0 \) and \( 1^T v = 1 \) for all \( v \in m \). \( \odot \) denotes the point-wise, Hadamard, product; An undirected graph with self-loops included is denoted by \( G = (V, E) \) where \( V = \{1, \ldots, m\} \) is a set of vertices and \( E = \{1, \ldots, q\} \) if the set of edges. The adjacency matrix \( A \) of the graph \( G \) is: \( A_{ij} = 1 \) if \( (i, j) \in E \) and \( A_{ij} = 0 \) otherwise. Let \( C_G \subseteq m^{m \times m} \) be the set of matrices following the graph \( G \) defined as follows: \( X \in C_G \) implies that \( X_{ij} = 0 \) if the link \( (i, j) \notin E \) and \( X_{ij} > 0 \) otherwise.

B. Markov Chain Specifications

We consider a Markov chain describing the evolution of a discrete probability vector \( x(t) \in m^m \) in time given by (1), where \( M(t) \) is a column stochastic matrix for all \( t \); hence the probability vector \( x(t) \geq 0 \) stays normalized as \( 1^T x(t) = 1 \) for all \( t \geq 0 \). Motivated by Markov chain terminology, \( M(t) \) is referred to as the transition matrix where \( M_{ij}(t) \) is the probability of transition from a state \( j \) to state \( i \) at time \( t \).

We first consider the case with a constant transition matrix \( M(t) = M \) for all \( t \). We will discuss the time-varying case after introducing the M-H algorithm to handle time-varying stationary distributions. Specifically our objective is to synthesize an \( m \times m \) transition matrix \( M \) such that the resulting Markov chain via (1) has the following properties:

1) Desired steady state: \( v \in m^m; \lim_{t \to \infty} x(t) = \hat{v} \forall x(0) \in m^m \).
2) Reversibility: \( \hat{v}_i M_{ji} = \hat{v}_j M_{ij}, \ i,j = 1, \ldots, m \).
3) Transition constraints: \( M_{ij} = 0 \) when \( (ij) \notin E \).
4) Safety constraints: \( x(t) \leq d \) for all \( t \geq 0 \), and for a given vector \( 0 \leq d \leq 1 \).

The desired steady state \( \hat{v} \) defines a probability distribution over the discrete set of states. In consensus protocols, \( \hat{v} = \frac{1}{m} 1 \) is the uniform distribution. In RMP, it is the desired density distribution of autonomous agents over a given operational area.

Transition constraints are given by an adjacency matrix describing the set of feasible state transitions.\(^1\) Safety constraints bound the probability distribution during both: the transient and the stationary dynamics. For example, in RMP for multi-agent systems, this constraint can prevent overcrowding of agents in subregions.

C. Convex Characterization of the Specifications

This section summarizes our results on convex formulations of the four system specification properties for a time-invariant Markov matrix \( M \). These convex representations allow us to formulate Linear Matrix Inequality (LMI) problems for the synthesis of reversible Markov chains for given steady-state distributions [17].

Stochasticity, Transition, and Reversibility Constraints:
First of all, the Markov matrix must satisfy the column stochasticity constraints described as
\[ M \geq 0 \quad \text{and} \quad 1^T M = 1^T, \]
which are linear equality and inequality constraints on \( M \).

The transition constraints can be expressed via the adjacency matrix \( A \) of the graph \( G \) that represents all feasible transitions \( M \in C_G \):
\[ M_{ij} = \begin{cases} 0 & \text{if} (i, j) \notin E \\ > 0 & \text{if} (i, j) \in E \end{cases} \iff (11^T - A) \odot M = 0 \]
\[ \exists \epsilon > 0 \text{ s.t. } M \geq \epsilon A, \]
which impose linear constraints on \( M \). Here we can choose \( \epsilon \) a sufficiently small positive scalar.

Reversible chains have favorable properties (e.g., detailed balance condition) and are commonly utilized in applications of Markov Chain Monte Carlo (MCMC) methods, e.g., in birth death processes, M/M/1 queues, and symmetric random walks on graphs. A Markov matrix \( M \) is reversible with a stationary distribution \( v \) if and only if [27]
\[ M \text{diag}(v) = \text{diag}(v) M^T. \]
The above equation implies that \( v \) is a steady-state distribution for the resulting Markov chain, which can be obtained by simply multiplying it by \( 1 \). \( Mv = v \). The set of all admissible Markov matrices with a given steady-state distribution, \( v \), and adjacency matrix, \( A \), is:
\[ \mathcal{M}_G(v) := \{ \text{Matrices satisfying (3), (4), and (5)} \}. \]
The results in this paper hold (with minor modifications) when \( v \) has some zero entries. Without loss of generality, we assume in what follows that \( v > 0 \).

\(^1\)In consensus protocols, they represent possible communication links between agents. In RMP, transition constraints represent feasible transitions for an agent as a function of its current state.
We can parameterize all reversible Matrices having $v$ as the steady-state distribution. Let $\Delta$ be an $m \times m$ matrix defined by the stationary distribution $v$ as follows:

$$\Delta_{ij}(v) := \begin{cases} v_i/v_j & \text{if } i < j \\ 1 & \text{else.} \end{cases} \quad (7)$$

Next we give a useful parameterization of all reversible Markov matrices for a given $v$.

**Lemma 1.** $M \in \mathcal{M}_G(v)$ if and only if there exists an $m \times m$ matrix $Y$ such that

$$M = \Delta(v) \circ Y + I - \text{diag}(1^T(\Delta(v) \circ Y)),$$

$$M \geq \epsilon I, \quad Y = Y^T, \quad Y \in C_G$$

where $\epsilon > 0$ is a positive scalar.

**Proof.** Let $Y$ be a matrix such that (8) is satisfied, then

$$1^T M = 1^T (\Delta(v) \circ Y) + 1^T - 1^T (\Delta(v) \circ Y) = 1^T,$$

and adding that $M \geq \epsilon I \geq 0$ implies (3) is satisfied. For $i < j$, we have $M_{ij} = \Delta_{ij}(v)Y_{ij} = (v_i/v_j)Y_{ij}$, and for $i > j$, we have $M_{ji} = \Delta_{ji}(v)Y_{ji} = Y_{ji}$. Then, adding that $Y$ is a symmetric matrix, we get $M_{ij} = (v_i/v_j)Y_{ij} = (v_i/v_j)M_{ji}$, which implies that (5) is satisfied. Since $Y \in C_G$ and $M_{ij} \geq \epsilon$, then $M \in C_G$ and (4) is satisfied. Therefore $M \in \mathcal{M}_G(v)$. To show the other direction of the lemma, now suppose that $M \in \mathcal{M}_G(v)$, it is sufficient to construct $Y$ from $M$ such that $Y$ satisfies equations (8). It is indeed the case by constructing $Y$ as follows: $Y_{ij} = M_{ij}$ if $i > j$, $Y_{ij} = Y_{ji}$ if $i < j$, $Y_{ij} = \epsilon$ if $i = j$.

Lemma 1 shows that any any matrix $M$ that generates a reversible chain can be parameterized by a set of linear constraints.

**Convergence to Steady-State Distribution:** The following well known result (see for example [18]) shows that asymptotic convergence to $v$ is ensured on matrix $M$ when the spectral radius condition is satisfied,

$$\rho(M - v1^T) < 1. \quad (9)$$

**Lemma 2.** Consider a column stochastic matrix, $M$, such that $Mv = v$. Consider the dynamical system (1). Then, $\lim_{t \to \infty} x(t) = v$ holds for any initial probability vector $x(0)$ if and only if (9) is satisfied.

The following lemma gives a known characterization for the convergence of the system using graph $G$ properties

**Lemma 3.** If $G = (V,E)$ is connected, then

$$\rho(M - v1^T) < 1 \quad \text{for all } M \in \mathcal{M}_G(v).$$

**Proof.** When $G$ is connected, then for any pair of vertices $i$ and $j$, we can find a path $iu_1u_2\ldots u_{l-1}j$, and thus $(M^l)_{ij} \geq M_{iu_1}M_{u_1u_2}\ldots M_{u_{l-1}j} > 0$ (the last inequality is due to (4)). Since $l \leq m$, then $M$ is irreducible as for every pair $i,j$ of its index set, there exists a positive integer $l \equiv l(i,j)$ such that $(M^l)_{ij} > 0$ [28, p. 18]. Moreover, since self-loops belong to $E$, then $M_{ii} > 0$ for all $i$ and thus $M$ is primitive having a unique stationary distribution $v$, so $\lim_{t \to \infty}(M - v1^T)^t = 0$ and the lemma follows.

We will assume that the graph $G$ is connected thereafter. Note that $\mathcal{M}_G(v)$ is a convex set because (3), (4) and (5) are linear (in)equality conditions, which facilitate convex synthesis.

**Safety:** We consider the following safety constraints [16], [29], [17].

$$x(t) \leq d, \quad t = 1, 2, \ldots, \quad \text{if } x(0) \leq d, \quad (10)$$

which impose that the probability vector (density distribution) is bounded by the vector $d$. These constraints can be useful in RMP for collision/conflict mitigation by limiting the densities, hence eliminating crowding during transitions. We have the following recent result [17] on Markov chain synthesis.

**Theorem 1.** Consider the Markov chain dynamics given by (1) with a constant $M$ matrix. Given a scalar $\gamma \in [0,1]$, for any $x(0) \leq d$, the following holds

$$x(t) \leq (1 - \gamma)d \quad \forall t = 1, 2, \ldots$$

if and only if there exist two variables $S \in \mathbb{R}^{m \times m}$ and $y \in \mathbb{R}^m$ such that:

$$S \geq 0, \quad M + S + y1^T \geq 0, \quad (M + S + y1^T)d \leq y + (1 - \gamma)d. \quad (11)$$

**Proof.** This theorem was proved in [17] for $\gamma = 0$. When $\gamma \in [0,1]$, the proof can be refined from [17] by simply having a factor $(1 - \gamma)$ in the derived equations.

III. ROBUST METROPOLIS-HASTINGS (M-H) ALGORITHM

This section presents a new M-H synthesis method for the $M(t)$ matrix in (1) as a function of a time-varying desired distribution, $v(t)$. If the desired distribution does not change with time, then the algorithm will generate a constant Markov matrix. However, if $v$ is time-varying, the Markov matrix will be time-varying as well. Our main result adapts the well-known M-H algorithm to handle the safety constraints for a time-varying stationary distribution. In particular, we show that M-H algorithm can be applied for a well-characterized class of stationary distributions by designing the proposal matrix to ensure convergent and safe Markov chains. Beyond having a new M-H result to generate safe Markov chains, the resulting algorithm presents a computationally inexpensive method, which makes it easily adaptable for online computations once a proper proposal matrix is generated offline via using convex optimization techniques. In summary, this section presents a method to synthesize the proposal matrix based on the results of the previous section, and will also characterize the set of stationary distributions for which M-H algorithms generates safe reversible Markov chains.
A. M-H Algorithm

The M-H algorithm [4], [2] is a Markov Chain Monte Carlo (MCMC) method for obtaining a sequence of random samples from a distribution.

**Definition 1. [M-H Algorithm for a given v]** The M-H algorithm produces an $M$ matrix given by:

$$M_{ij} = \begin{cases} K_{ij} F_{ij} & \text{if } i \neq j \\ K_{jj} + \sum_{k \neq j} (1 - F_{kj}) K_{kj} & \text{if } i = j \end{cases} \quad (12)$$

where $K$ is a proposal matrix that satisfies $K \geq 0$ and $1^T K = 1^T$; $v > 0$; and $F$ is an acceptance matrix, which satisfies for $i \neq j$.

$$F_{ij} = \min\left(1, \frac{v_i K_{ji}}{v_j K_{ij}}\right) \quad i, j = 1, \ldots, m. \quad (13)$$

Note that if $K_{ij} = 0$, then $M_{ij} = M_{ji} = 0$. Similarly, having $K_{ij} > 0$ and $K_{ji} > 0$ implies that $M_{ij} > 0$ and $M_{ji} > 0$. Consequently, when $v > 0$ and the matrix $F$ is chosen by (13), we can impose transition constraints, given by (4), on the proposal matrix, $K$, to guarantee that $M$ also satisfies the transition constraints.

**Lemma 4. Consider the M-H algorithm, for some $v > 0$, given by (12). If the proposal matrix $K \in M_G(u)$ for some probability vector $u$, then the resulting Markov matrix $M \in M_G(v)$.**

Proof. The matrix $M$ is by the algorithm’s construction a column stochastic matrix satisfying the reversibility assumption. Since $K \in M_G(u)$, then $K \in C_G$, and thus $M \in C_G$. As a result, $M \in M_G(v)$ and this ends the proof. □

Remark: The M-H can transform any proper proposal matrix into a Markov matrix with a desired stationary probability distribution $v$. Incorporating safety constraints into the resulting transition matrix $M$ for a time-varying distribution $v(t)$ is not straightforward. In the example of multi-agent system motion planning, even if the agents have sufficient computational capabilities (being able to solve optimization problems online), the safety might not be guaranteed. For example, two agents operating via two different safe Markov matrices might lead to an unsafe combined behavior. Therefore, it becomes natural to ask under what conditions, the M-H procedure can be safe? It turns out that if the proposal matrix for a nominal distribution $\hat{v}$ is safe and have some robustness properties, then the M-H procedure can also be safe as long as $v$ is in the neighborhood of $\hat{v}$ as we will discuss next.

B. Robust Proposal for Safe M-H Algorithm

By construction, M-H algorithm can ensure reversible Markov chains with the desired steady-state distribution and transition constraints. However, the safety specification is not necessarily guaranteed. In this section, we study the effect of the proposal matrix $K$ on the safety of M-H algorithm. For this purpose, we introduce the following definitions.

**Definition 2 (S-Safe). M-H algorithm with a given proposal matrix $K$ is called S-Safe if the resulting matrix $M$ leads to Markov chains satisfying the safety constraints (10) for all steady-state distributions $v \in S$.**

**Definition 3. Given a matrix $K \geq 0$ and $\gamma \in [0, 1]$, $V_\gamma(K) \subseteq \mathbb{F}^m$ is defined as follows: $V_\gamma(K) = \{ v \in \mathbb{F}^m : v \text{ satisfies (14)} \}$.**

$$\sum_{k=1}^m \max\{0, v_i K_{ki} - v_k K_{ik}\} \leq \gamma v_i, \text{ for } i = 1, \ldots, m. \quad (14)$$

Note that the set $V_\gamma(K)$ is parameterized by $\gamma$, which can be considered as a robustness parameter as discussed next: The larger $\gamma$ is, the larger is the set for which the M-H algorithm is safe.

**Proposition 1. Consider a proposal matrix $K \in M_G(\hat{v})$ used in M-H algorithm that results in a reversible Markov chain. Then $V_\gamma(K)$ is a nonempty convex set for any $\gamma \in [0, 1]$.**

Proof. $V_\gamma(K)$ is nonempty because by considering $\hat{v}$, the stationary distribution of $K$, the following equations hold $\hat{v}_i K_{ki} = \hat{v}_k K_{ik}$ for all $i$ and $k$. Therefore the left hand side of (14) is zero and the equations are satisfied for any $\gamma \in [0, 1]$. In particular, if $\gamma = 0$, then $\hat{v}$ is the unique value of $v$ that satisfies the inequalities and in this case $V_\gamma$ is a singleton. If $\gamma = 1$, then $V_\gamma$ is the set of all probability vectors ($V_\gamma = \mathbb{F}^m$) since $\sum_{k=1}^m \max\{0, v_i K_{ki} - v_k K_{ik}\} = \sum_{k=1}^m v_i K_{ki} = v_i$.

$V_\gamma$ is a convex set of probability vectors $v$ because of the following argument: given a matrix $K$, then $v_i K_{ki} - v_k K_{ik}$ is a convex function of $v$ (a linear function of $v$), $\max\{0, v_i K_{ki} - v_k K_{ik}\}$ is convex (the maximum of convex functions), and $\gamma v_i$ is convex (sum of convex functions), the set defined by $\sum_{k=1}^m \max\{0, v_i K_{ki} - v_k K_{ik}\} - \gamma v_i \leq 0$ is a convex set (sub-level set of a convex function), and finally $V_\gamma$ is a convex set (intersection of convex sets). Moreover, $V_\gamma$ defines a neighborhood around $\hat{v}$ because for any $\gamma > 0$, a small enough perturbation around $\hat{v}$ would satisfy equations (14). □

We can now give the main technical result of this paper.

**Theorem 2. Suppose that there exist variables $Y \in \mathbb{R}^{m \times m}$, $S \in \mathbb{R}^{m \times m}$, $\epsilon > 0$, and $y \in \mathbb{R}^m$ such that the proposal matrix $K$ satisfies the following conditions**

$$K = \Delta(\hat{v}) \circ Y + I - \text{diag}(1^T (\Delta(\hat{v}) \circ Y)), \quad (15)$$

$$\begin{align*}
K &\geq \epsilon I, \\
Y &= Y^T, \\
Y &\in C_G \\
(K + S + y1^T) d &\leq y + (1 - \gamma) d, \\
S &\geq 0, \\
K + S + y1^T &\geq 0,
\end{align*}$$

where $\Delta(\hat{v})$ is given by (7) and $\gamma \in [0, 1]$. Then the M-H algorithm with the proposal matrix $K$ is $V_\gamma$-Safe.

Proof. The proof is provided in the Appendix. □

It is important to note that, the M-H algorithm is safe for any desired distribution if $\gamma = 1$. Also note that
Theorem 2 results in a Markov matrix $K$ with reversible chains and the set $\mathcal{V}_\gamma$ is a convex set containing $\hat{\nu}$ (via Proposition 1). Fig. 1 illustrates the size of the set $\mathcal{V}_\gamma$ as a function of the robustness parameter $\gamma$. It demonstrates that, as $\gamma$ approaches 1, the size of the set becomes larger and approaches the sample space, i.e., the simplex defining the probability distributions. This implies that maximizing $\gamma$ maximizes the size of the set $\mathcal{V}_\gamma$, which is the basis of the following LP-based synthesis for the proposal matrix $K$.

\[
\begin{align*}
\text{maximize} & \quad \gamma \\
\text{subject to} & \quad (15).
\end{align*}
\]

Remark: The robustness parameter $\gamma$ and the matrix $K$ are not independent, i.e., fixing one of them and solving for the other can render the problem infeasible. Nonetheless, this, (16) provides the maximum possible value for $\gamma$ for which a feasible proposal matrix $K$ can be found.

C. Robustness and Convergence Speed

Note that, in the optimization problem (16), the speed at which the proposal matrix $K$ converges to its stationary distribution $\hat{\nu}$ is not taken into account by the constraints. Though the convergence is guaranteed (due to Lemma 3), the speed at which the system converges can be arbitrarily slow. Therefore, it is desirable to generate proposal matrices having both fast speed of convergence and robustness properties. It is well-known that the speed of convergence of a Markov matrix to its stationary distribution is governed by the second largest eigenvalue in magnitude $|\lambda_2|$. Since $K$ is searched over Markov matrices for reversible chains, the fastest converging proposal matrix can be computed via solving the following Semi-Definite Programming (SDP) problem [30] [19] for a prescribed robustness measure $\gamma$:

\[
\begin{align*}
\text{minimize} & \quad \lambda \\
\text{subject to} & \quad -\lambda I \preceq Q^{-1}KQ - qq^T \leq \lambda I \quad \text{Eq. (15)}
\end{align*}
\]

where $q = \hat{\nu}^{1/2}$ is the element-wise square root of $\hat{\nu}$ and $Q = \text{diag}(q)$. Note that $\gamma$ is not a variable in this formulation but a given value. The above SDP has a feasible solution for $[0, \gamma_{\text{max}}]$ where $\gamma_{\text{max}}$ is computed via (16). Then a proportional measure of convergence speed is the spectral gap, defined as $s_q = 1 - \lambda^*$ where $\lambda^*$ is the minimal value of the SDP (17). Since any feasible solution from (15) for a given $\gamma_1$ is also a feasible solution for $\gamma_2$ where $\gamma_2 \leq \gamma_1$, we expect the convergence speed to be a non-increasing function of $\gamma$ when $\gamma$ varies from 0 to $\gamma_{\text{max}}$. It turns out that in fact, the speed of convergence has low sensitivity to the robustness parameter. In other words, we can have both fast convergence and robustness at the same time. Fig. 2 demonstrates this observation via simulation on an example with 20 states, i.e., $x(t) \in \mathbb{P}^{20}$.

IV. Conclusions

A robust M-H algorithm is introduced that ensures the satisfaction of safety upper-bound constraints for the synthesized Markov chains. The proposal matrix of the M-H algorithm is designed offline using a linear programming formulation to maximize the set of robust desired stationary distributions that the M-H algorithm can handle safely. This solution method is further improved to achieve fast convergence using a semidefinite programming formulation. Our results show that both, speed and robustness, can be achieved simultaneously.

---

Footnote:

---

Note that, in the optimization problem (16), the speed at which the proposal matrix $K$ converges to its stationary distribution $\hat{\nu}$ is not taken into account by the constraints. Though the convergence is guaranteed (due to Lemma 3), the speed at which the system converges can be arbitrarily slow. Therefore, it is desirable to generate proposal matrices having both fast speed of convergence and robustness properties. It is well-known that the speed of convergence of a Markov matrix to its stationary distribution is governed by the second largest eigenvalue in magnitude $|\lambda_2|$. Since $K$ is searched over Markov matrices for reversible chains, the fastest converging proposal matrix can be computed via solving the following Semi-Definite Programming (SDP) problem [30] [19] for a prescribed robustness measure $\gamma$:
APPENDIX

PROOF OF THEOREM 2

The first two conditions in Equation (15) ensure that $K \in \mathcal{M}(\hat{v})$ by Lemma 1. The last two conditions in Equation (15) ensure that $K$ satisfies the conditions of Theorem 1 and thus $K$ is safe. Moreover, using Lemma 4, the resulting transition matrix $M$ from the M-H algorithm belongs to $\mathcal{M}_G(\hat{v})$. It remains to prove that $M$ is safe when $\hat{v} \in \mathcal{V}_\gamma$. Note that the last two lines are equivalent to the following expression (Theorem 1),

$$Kx \leq (1 - \gamma)d, \quad \text{for all } 0 \leq x \leq d, x^T1 = 1. \quad (18)$$

To show that $M$ is safe, we need to show that for any $0 \leq x \leq d, x^T1 = 1$, we have that $MX \leq d$, or equivalently

$$\sum_k M_{ik}x_k \leq d_i, \quad \text{for } i = 1, \ldots, m.$$ 

Notice that from the M-H algorithm (depending on the ratio $C := \frac{v_{ik}K_{ik}}{v_i}$) either (a) $C \leq 1$, $M_{ik} = K_{ik}$, $M_{ki} = \frac{v_k}{v_i}K_{ik}$ or (b) $C \geq 1$, $M_{ik} = \frac{v_i}{v_k}K_{ki}$, $M_{ki} = K_{ki}$. Thus we can divide the neighbors of $i$ in $\mathcal{G}$, $N_i$, into two mutually exclusive sets: $N_i^1 = \{k \in N_i; \text{condition (a) is satisfied}\}$, and $N_i^2 = \{k \in N_i; \text{condition (b) is satisfied}\}$ satisfying $N_i^1 \cup N_i^2 = N_i$ and $N_i^1 \cap N_i^2 = \emptyset$. Therefore we have:

$$M_{ii} = 1 - \sum_{k \in N_i^1} M_{ik} = 1 - \sum_{k \in N_i^1} \frac{v_k}{v_i}K_{ik} - \sum_{k \in N_i^2} K_{ki}.\quad (20)$$

As a result, for $i = 1, \ldots, m$

$$\sum_k M_{ik}x_k = M_{ii}x_i + \sum_{k \in N_i^1} M_{ik}x_k + \sum_{k \in N_i^2} M_{ik}x_k = (1 - \sum_{k \in N_i^1} \frac{v_k}{v_i}K_{ik} - \sum_{k \in N_i^2} K_{ki})x_i + \sum_{k \in N_i^1} K_{ik}x_k + \sum_{k \in N_i^2} \frac{v_i}{v_k}K_{ki}x_k \leq d_i, \quad \text{for } i = 1, \ldots, m.\quad (21)$$

Note that, the first inequality in (19) is due to $K$ satisfying (18), the last inequality follows directly from the M-H algorithm (i.e., because for any $k \in N_i^2$, $C \geq 1$). It remains to show that $u_i \leq \gamma$,

$$u_i = \sum_{k \in N_i^1} (K_{ki} - \frac{v_k}{v_i}K_{ik}) = \frac{1}{v_i} \sum_{k \in N_i^1} (v_iK_{ik} - v_kK_{ik}) = \frac{1}{v_i} \sum_{k=1}^m \max\{0, v_iK_{ik} - v_kK_{ik}\} \leq \frac{1}{v_i} \gamma v_i \leq \gamma, \quad \text{for } i = 1, \ldots, m.$$ 

where (21) follows from the definition of $N_i^1$ and the inequality (22) follows from the definition of $\mathcal{V}_\gamma$. Combining (20) and (23) shows that $M$ is safe (i.e., $MX \leq d$ for all $x \leq d$), which concludes the proof.

REFERENCES


