CERTIFYING UNSTABILITY OF SWITCHED SYSTEMS USING SUM OF SQUARES PROGRAMMING*

BENOÎT LEGAT†, PABLO A. PARRILO‡, AND RAPHAËL M. JUNGERS§

Abstract. The joint spectral radius (JSR) of a set of matrices characterizes the maximal asymptotic growth rate of an infinite product of matrices of the set. This quantity appears in a number of applications including the stability of switched and hybrid systems. A popular method used for the stability analysis of these systems searches for a Lyapunov function with convex optimization tools. We investigate dual formulations for this approach and leverage these dual programs for developing new analysis tools for the JSR.

We show that the dual of this convex problem searches for the *occupations measures* of trajectories with high asymptotic growth rate. We both show how to generate a sequence of guaranteed high asymptotic growth rate and how to detect cases where we can provide lower bounds on the JSR.

All results of this paper are presented for the general case of constrained switched systems, that is, systems for which the switching signal is constrained by an automaton.

Key words. Joint spectral radius, Sum of squares programming, Switched Systems, Path-complete Lyapunov functions

AMS subject classifications. 93D05, 93D20, 93D30

1. Introduction.

In recent years, the study of the stability of hybrid systems has been the subject of extensive research using methods based on classical ideas from Lyapunov theory and modern mathematical optimization techniques. Even for switched linear systems, arguably the simplest class of hybrid systems, determining stability is undecidable and approximating the maximal asymptotic growth rate that a trajectory can have is NP-hard [9]. Despite these negative results, the vast range of applications has motivated a wealth of algorithms to approximate this maximal asymptotic growth rate.

A switched linear system is characterized by a finite set of matrices $\mathcal{A} \triangleq \{A_1, A_2, \ldots, A_m\} \subset \mathbb{R}^{n\times n}$ and the iteration

$$x_k = A_{\sigma_k} x_{k-1}, \quad \sigma_k \in [m] \tag{1}$$

where $[m]$ denotes the set $\{1, \ldots, m\}$. The maximal asymptotic growth rate of this iteration is given by the *joint spectral radius* (JSR). The JSR $\rho(\mathcal{A})$ of a finite set of matrices $\mathcal{A}$ is defined as

$$\rho(\mathcal{A}) = \lim_{k\to\infty} \max_{\sigma\in[m]^k} \|A_{\sigma_k} \cdots A_{\sigma_2} A_{\sigma_1}\|^{1/k}.$$

This definition is independent of the norm used.

The JSR was introduced by Rota and Strang [41] and has many other applications such as wavelets, the capacity of some particular codes, zero-order stability of ordinary

*Submitted to the editors September 28, 2017. A preliminary version of this work appeared in Proceedings of Hybrid Systems: Computation and Control, 2016 [30].

†B. Legat is a FNRS Research Fellow. Address: ICTEAM institute, UCLouvain, Louvain-la-Neuve, Belgium (benoit.legat@uclouvain.be).

‡Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge MA 02139, USA (parrilo@mit.edu).

§R. M. Jungers is a FNRS honorary Research Associate. This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme under grant agreement No 864017 - L2C. His is also supported by the Walloon Region and the Innoviris Foundation. Address: ICTEAM institute, UCLouvain, Louvain-la-Neuve, Belgium (raphael.jungers@uclouvain.be).

differential equations, congestion control in computer networks, curve design and networked and delayed control systems; see [21] for a survey on the JSR and its applications. Many algorithms exist for estimating the JSR but not much is known on how to generate an infinite sequence of matrices with an asymptotic growth rate close to the JSR. However generating such sequence can be of particular interest, depending on the application, such as exhibiting unstable trajectories for switched linear systems to prevent them from occuring [15]. The currently known algorithms generate a sequence of matrices with high spectral radius using methods detailed in Section 3.6 and repeat this sequence infinitely [16, 18, 17, 22].

Approximating the JSR usually consists in certifying upper bounds $\overline{\gamma}$ to the JSR by exhibiting a Lyapunov function or an invariant set for the matrices $A_i/\overline{\gamma}$. The search for such Lyapunov functions can naturally be written as a convex optimization program; see Program 3. Certifying lower bounds $\underline{\gamma}$ is currently either achieved using the guarantees we have on the accuracy of the upper bound on the JSR or by exhibiting trajectories of asymptotic growth rate $\underline{\gamma}$. In this paper, we introduce a new way to certify lower bounds by exhibiting nonnegative measures satisfying some invariance condition parametrized by the matrices $A_i/\underline{\gamma}$; see (9). This invariance condition is linear on the measure hence the search of measures on the convex cone of nonnegative measures is a *convex* program; see Program 4. It turns out that this program is the dual of Program 3.

We revisit the sum-of-squares program proposed by Parrilo and Jadbabaie [36] and show that its dual formulation is the moment relaxation of the search of the measures satisfying the invariance condition.

Thanks to this duality, solving this pair of programs with a given candidate value γ for the JSR either returns Lyapunov functions certifying that $\rho(\mathcal{A}) \leq \gamma$ or returns moments that form a solution of the moment relaxation; see Section 3.2. These moments are not necessarily the moments of measures satisfying the invariance conditions. However, we give a rounding procedure to extract a (infinite) switching sequence from these moments and provide a guarantee on the asymptotic growth rate of this sequence. As a by-product of the rounding procedures, the spectral radius of a finite part of this infinite sequence can be used to give lower bounds on the JSR. In addition, we give a way to sometimes detect when the moments belongs to measures that satisfy the invariance conditions. This happens when the measures are the convex combination of the occupation measures of several periodic trajectories. Since the trajectories are periodic, the measures are atomic and we can recover them from moments of sufficiently high degree. We show on numerical examples that these techniques work well in practice.

In some applications the values that σ_k can take in (1) may depend on $\sigma_{k-1}, \sigma_{k-2}, \ldots$. These constraints are often conveniently represented using a *finite automaton* and the JSR under such constraints is called *constrained joint spectral radius* (CJSR) [12]; an example of constrained switched system is given by Example 1 and its automaton is illustrated by Figure 1.

Example 1 (Running example). We borrow the example of [38, Section 4]. The set of matrices $\mathcal{A}$ is composed of the following four matrices

$$
\begin{aligned}
A_1 &= A + B\begin{pmatrix} k_1 & k_2 \end{pmatrix}, & A_2 &= A + B\begin{pmatrix} 0 & k_2 \end{pmatrix},\\
A_3 &= A + B\begin{pmatrix} k_1 & 0 \end{pmatrix}, & A_4 &= A.
\end{aligned}
$$

where $k_1 = -0.49$, $k_2 = 0.27$,

$$A = \begin{pmatrix} 0.94 & 0.56 \\ 0.14 & 0.46 \end{pmatrix} \text{ and } B = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

The corresponding automaton is represented by Figure 1.

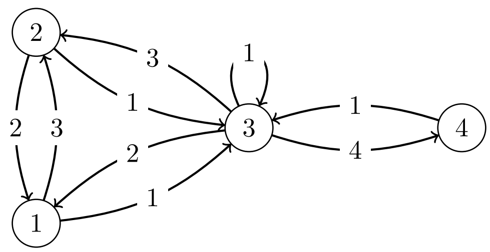

Fig. 1: Automaton for the running example. The numbers on the edges are their respective labels.

The automaton representing the constraints can be represented by a strongly connected labelled directed graph $G(V,E)$ of nodes V and edges E, possibly with parallel edges. The labels are elements of the set $[m]$ and E is a subset of $V \times V \times [m]$. We say that $(u, v, \sigma) \in E$ if there is an edge between node u and node v with label σ. The iteration 1 is rewritten as follows to take the automaton into account:

(2) $\quad x_k = A_{\sigma_k} x_{k-1}, \qquad (\sigma_1, \dots, \sigma_k)$ are the respective labels of a path in G.

Reproducibility. The code used to obtain the results is published on codeocean [33]. The algorithms are part of the SwitchOnSafety package [29] which computes invariant sets for hybrid sytems represented with the HybridSystems package [28]. The implementation relies on the SumOfSquares [27] and SetProg [31] extensions of JuMP [13]. The solver used is Mosek v8.1.0.82 [3]. Both the new methods presented in this paper and the alternative approaches we compare our algorithm with are implemented in Julia [6] in order to ensure a fair performance comparison.

Notations. We define the automaton $G^\top(V, E^\top)$ where $E^\top = \{\, (v, u, \sigma) : (u, v, \sigma) \in E \,\}$. We denote as E_k the subset of E^k (the kth cartesian power of E) that represents paths of length k in G. The k-tuple $(\sigma_1, \sigma_2, \dots, \sigma_k)$ is said to be *G-admissible* if $\sigma_1, \dots, \sigma_k$ are the respective labels of a path of length k in G. We denote the set $\{1, \dots, m\}$ as $[m]$ and the set of all k-tuples of $[m]^k$ that are G-admissible as G_k. The sequence $\sigma_1, \sigma_2, \dots$ is G-admissible (resp. $G^\top$-admissible) if $(\sigma_1, \dots, \sigma_k)$ (resp. $(\sigma_k, \dots, \sigma_1)$) is G-admissible for any $k \geq 1$. We denote $A_{\sigma_k} \cdots A_{\sigma_1}$ as A_s where $s = (\sigma_1, \dots, \sigma_k)$ or s is a path with these respective labels.

To shorten the notation we denote the ith node of a path s as $s(i)$ and the ith edge as $s[i]$. Also, for a given k-tuple s, we denote $(s(i), \dots, s(k))$ by $s(i:)$. We define

$$\begin{aligned} E_k^-(v) &= \{\, s \in E_k \mid s(k+1) = v \,\} & E_k^-[e] &= \{\, s \in E_k \mid s[k] = e \,\} \\ E_k^+(v) &= \{\, s \in E_k \mid s(1) = v \,\} & E_k^+[e] &= \{\, s \in E_k \mid s[1] = e \,\} \\ E_k(u,v) &= E_k^+(u) \cap E_k^-(v). \end{aligned}$$

We denote the indegree (resp. outdegree) of a node $v \in V$ as $d^-(v)$ (resp. $d^+(v)$) and the maximum indegree (resp. outdegree) of G as $\Delta^-(G) = \max_{v \in V} d^-(v)$ (resp.

$\Delta^+(G) = \max_{v \in V} d^+(v)$). We also denote the number of paths of length k ending (resp. starting) at a node $v \in V$ as $d_k^-(v) \triangleq |E_k^-(v)|$ (resp. $d_k^+(v) \triangleq |E_k^+(v)|$) and define $\Delta_k^-(G) = \max_{v \in V} d_k^-(v)$ and $\Delta_k^+(G) = \max_{v \in V} d_k^+(v)$. Note that $\Delta_1^-(G) = \Delta^-(G)$, $\Delta_1^+(G) = \Delta^+(G)$ and for any k, $\Delta_k^+(G^\top) = \Delta_k^-(G)$.

2. Instability certificate using measures. The definition of the JSR is generalized as follows for constrained systems.

Definition 2 ([12]). *The* constrained joint spectral radius *(CJSR) of a finite set of matrices $\mathcal{A}$ constrained by an automaton G, denoted as $\rho(G, \mathcal{A})$, is*

$$\limsup_{k \to \infty} \rho_k(G, \mathcal{A}) = \rho(G, \mathcal{A}) = \lim_{k \to \infty} \hat{\rho}_k(G, \mathcal{A}, \|\cdot\|) \tag{3}$$

where

$$\rho_k(G, \mathcal{A}) = \max \left\{ \rho(c) : c \in G_k, c \text{ is a cycle} \right\}, \quad \rho(c) = [\rho(A_c)]^{1/k}, \tag{4}$$

and

$$\hat{\rho}_k(G, \mathcal{A}, \|\cdot\|) = \max \left\{ \|A_s\|^{1/k} : s \in G_k \right\}. \tag{5}$$

We can readily see that $\rho_k(G, \mathcal{A}) \le \hat{\rho}_k(G, \mathcal{A}, \|\cdot\|)$ for any k and *submultiplicative*[1] norm $\|\cdot\|$. Equality (3) is called the *Joint Spectral Radius Theorem* and was proved in 1992 by Berger and Wang [5] in the unconstrained case. Elsner [14] provided a somewhat simpler self contained proof in 1995. Both proofs use rather involved results on the joint spectral radius. In the general constrained case, the equality (3) was first proved in [12] with the help of heavy-weighted machinery of ergodic theory, and later simpler proofs appeared.

A popular method for proving stability of a dynamical system is to find a Lyapunov function. In this section, we introduce measures playing a role dual to Lyapunov function for switched system. These measures provide a certificate for instability. Finding Lyapunov functions and finding these measures are in fact two dual programs, they are respectively provided by Program 3 and Program 4. We will be succinct in our definition of measure-theoretic concepts but the interested reader can find an good introduction to writing programs using measures and functions as decision variables in [25].

Consider the dual pair $(\mathcal{B}, \mathcal{M})$ where $\mathcal{B}$ is the space of bounded measurable functions on $\mathbb{S}^{n-1} = \{ x \in \mathbb{R}^n : \|x\|_2 = 1 \}$ where $\|\cdot\|_2$ is the *Euclidean* norm, $\mathcal{M}$ is the space of *finite*[2] *signed*[3] Borel measures on $\mathbb{S}^{n-1}$ and the scalar product between a function $f \in \mathcal{B}$ and a measure $\mu \in \mathcal{M}$ is $\langle f, \mu \rangle = \int f \,\mathrm{d}\mu$. Given a function $f(x) \in \mathcal{B}$, we can define the homogeneous[4] function $h(f) \triangleq x \mapsto \|x\|_2 f(x/\|x\|_2)$ on $\mathbb{R}^n$. We define $\mathcal{F} = \{ h(f) \mid f \in \mathcal{B} \}$ with the scalar product $\langle h(f), \mu \rangle = \langle f, \mu \rangle$ for $f \in \mathcal{B}, \mu \in \mathcal{M}$.

Given an application A and a measure $\mu \in \mathcal{M}$, the *pushforward measure* $A\#\mu$ is often defined to be the measure given by $(A\#\mu)(B) = \mu(A^{-1}(B))$ for $B \in \mathbb{S}^{n-1}$. However, since $\mathbb{S}^{n-1}$ may not be invariant under application of the matrices of $\mathcal{A}$, we will use an alternative definition. Given an application A and a measure μ, the pushforward measure $A\#\mu$ is defined to be the measure such that $\langle f, A\#\mu \rangle = \langle f \circ$

[1] A matrix norm $\|\cdot\|$ is *submultiplicative* if $\|AB\| \le \|A\| \cdot \|B\|$ for all matrices A and B.

[2] The measure μ is *finite* if $\mu(\mathbb{S}^{n-1})$ is finite.

[3] A *signed* measure is a difference between two measures, i.e. $\mu - \nu$ where μ and ν are measures is a signed measure.

[4] A function f is homogeneous if $f(\alpha x) = \alpha f(x)$ for any scalar value α.

$A, \mu\rangle$ for any $f \in \mathcal{F}$. Moreover, given $B \subseteq \mathbb{S}^{n-1}$, we define $\mu(B) = \langle h(\mathbf{1}_B), \mu\rangle$ so that $(A\#\mu)(B)$ is well defined. Using these definitions, one can verify that for any application A and measure $\mu \in \mathcal{M}$,

$$(A\#\mu)(\mathbb{S}^{n-1}) \leq \mu(\mathbb{S}^{n-1}) \max_{x \in \mathrm{supp}(\mu)} \|Ax\|_2 \tag{6}$$

where $\mathrm{supp}(\mu)$ is the support of μ.

Let $\mathcal{F}_+$ (resp. $\mathcal{B}_+$) be the set of nonnegative functions of $\mathcal{F}$ (resp. $\mathcal{B}$), $\mathcal{M}_+$ be the set of (nonnegative) measures of $\mathcal{M}$ and $\mathcal{F}_{++}$ be the set of positive functions of $\mathcal{F}$. Given two functions $f, g \in \mathcal{F}$, $f \geq 0$ denotes $f \in \mathcal{F}_+$ and $f \geq g$ denotes $f - g \in \mathcal{F}_+$. Similarly, given two measures $\mu, \nu \in \mathcal{M}$, $\mu \geq 0$ denotes $\mu \in \mathcal{M}$ and $\mu \geq \nu$ denotes $\mu - \nu \in \mathcal{M}_+$.

Program 3 (Primal).
Input: A finite set of matrices $\mathcal{A}$ and an automaton G.
Output: Functions f_v and a number $\overline{\gamma}$.

$$\inf_{f_v \in \mathcal{F}, \overline{\gamma} \in \mathbb{R}} \overline{\gamma}$$

$$\text{subject to } f_v(A_\sigma x) \leq \overline{\gamma} f_u(x), \quad \forall x \in \mathbb{R}^n, \forall (u, v, \sigma) \in E, \tag{7}$$

$$f_v(x) \in \mathcal{F}_{++}, \quad \forall v \in V,$$

$$\sum_{v \in V} \int_{\mathbb{S}^{n-1}} f_v(x)\,\mathrm{d}x = 1. \tag{8}$$

Program 4 (Dual of Program 3).
Input: A finite set of matrices $\mathcal{A}$ and an automaton G.
Output: Measures $\mu_{uv\sigma}$ and a number $\underline{\gamma}$.

$$\sup_{\mu_{uv\sigma} \in \mathcal{M}, \underline{\gamma} \in \mathbb{R}} \underline{\gamma}$$

$$\text{subject to } \sum_{(u,v,\sigma) \in E} A_\sigma \# \mu_{uv\sigma} \geq \underline{\gamma} \sum_{(v,w,\sigma) \in E} \mu_{vw\sigma}, \quad \forall v \in V, \tag{9}$$

$$\mu_{uv\sigma} \in \mathcal{M}_+, \quad \forall (u, v, \sigma) \in E,$$

$$\sum_{(u,v,\sigma) \in E} \mu_{uv\sigma}(\mathbb{S}^{n-1}) = 1. \tag{10}$$

The constraint (7) is the Lyapunov constraint. The constraint (9) is similar to the *measure invariance constraint* $A\#\mu = \mu$ of a linear dynamical system $x_{k+1} = Ax_k$ and to the *mass balance constraint* of a *circulation problem* [2]. Without constraint (8) (resp. (10)), the feasible set of Program 3 (resp. Program 4) is a cone. These constraints have no effect on the optimal objective value but they make the feasible set bounded.

The main result of this section is summarized in the following theorem.

Theorem 5. *Consider a finite set of matrices $\mathcal{A}$ constrained by an automaton G. Let $\overline{\gamma}^\star$ (resp. $\underline{\gamma}^\star$) be the optimal value of Program 3 (resp. Program 4). The following identity holds:*

$$\underline{\gamma}^\star = \rho(G, \mathcal{A}) = \overline{\gamma}^\star.$$

As a consequence of Theorem 5, we have a new criterion for lower bounds on the CJSR using measures.

COROLLARY 6. *Consider a finite set of matrices $\mathcal{A}$ constrained by an automaton $G(V,E)$. If there exist non-trivial*[5] *measures $\mu_{uv\sigma}$ for each $(u,v,\sigma) \in E$ such that*

$$\sum_{(u,v,\sigma)\in E} A_\sigma \# \mu_{uv\sigma} \geq \underline{\gamma} \sum_{(v,w,\sigma)\in E} \mu_{vw\sigma}, \quad \forall v \in V$$

then $\underline{\gamma} \leq \rho(G,\mathcal{A})$.

The following lemma shows a recursive way to build an optimal solution of Program 3.

LEMMA 7. *Consider a finite set of matrices $\mathcal{A}$ constrained by an automaton $G(V,E)$. For any natural number k and norm $\|\cdot\|$, we have*

$$\overline{\gamma}^\star \leq \hat{\rho}_k(G,\mathcal{A},\|\cdot\|)$$

where $\hat{\rho}_k(G,\mathcal{A},\|\cdot\|)$ is defined in (5).

Proof. Let $A'_\sigma = A_\sigma/\hat{\rho}_k(G,\mathcal{A},\|\cdot\|)$, $f_v(x) = \max_{s\in\cup_{i=0}^{k-1} E_i^+(v)} \|A'_s x\|$. For any edge $(u,v,\sigma) \in E$,

$$\begin{aligned} f_v(A'_\sigma x) &= \max\left(\max_{s\in E_{k-1}^+(v)} \|A'_s A'_\sigma x\|, \max_{s\in\cup_{i=0}^{k-2} E_i^+(v)} \|A'_s A'_\sigma x\|\right) \\ &\leq \max\left(\|x\|, \max_{s\in\cup_{i=1}^{k-1} E_i^+(u)} \|A'_s x\|\right) = f_u(x) \end{aligned}$$

so the Lyapunov functions f_v are solution for $\overline{\gamma} = \hat{\rho}_k(G,\mathcal{A},\|\cdot\|)$. ∎

Proof of Theorem 5. By Lemma 34, we have $\underline{\gamma}^\star = \overline{\gamma}^\star$, by Theorem 33, we have $\rho(G,\mathcal{A}) \leq \overline{\gamma}^\star$ and by Lemma 7 and (3), we have $\overline{\gamma}^\star \leq \rho(G,\mathcal{A})$. ∎

The following lemma illustrates the relation between atomic solutions of Program 4 and periodic trajectories. Lemma 7 and Lemma 8 somehow suggest that Program 3 is related to the definition (5) of the CJSR with norms while Program 4 is related to the definition (4) of the CJSR with the spectral radius.

LEMMA 8. *Consider a finite set of matrices $\mathcal{A}$ constrained by an automaton G and a cycle $c = (\sigma_1,\ldots,\sigma_k)$ of length k with intermediary nodes $v_0,\ldots,v_{k-1},v_k = v_0 \in V$ such that $(v_{i-1},v_i,\sigma_i) \in E$ for $i = 1,\ldots,k$. Let $x_0 \in \mathbb{R}^n$ and $\lambda > 0$ be such that $A_c x_0 = \lambda x_0$ with $\|x_0\|_2 = 1$, consider the following iteration*

$$x_i = A_{\sigma_i} x_{i-1} \qquad \hat{x}_i = x_i/\|x_i\|_2 \qquad \alpha_i = \|x_i\|_2/\lambda^{i/k}$$

The following solution

$$\left(\mu_{uv\sigma} = \sum_{i=1,v_i=v}^{k} \alpha_i \delta_{\hat{x}_i}\right)_{(u,v,\sigma)\in E}$$

is feasible for Program 4 with any $\underline{\gamma} \geq \lambda^{1/k}$ and it satisfies the constraints (9) as equality for $\underline{\gamma} = \lambda^{1/k}$.

Proof. By construction, $\alpha_k = 1$ so $\alpha_k \delta_{\hat{x}_k} = \delta_{x_0}$ and for each $i = 0,\ldots,k-1$, we have

$$A_{\sigma_i} \# (\alpha_i \delta_{\hat{x}_i}) = \alpha_i \frac{\|x_{i+1}\|_2}{\|x_i\|_2} \delta_{\hat{x}_{i+1}} = \lambda^{1/k} \alpha_{i+1} \delta_{\hat{x}_{i+1}} \leq \underline{\gamma} \alpha_{i+1} \delta_{\hat{x}_{i+1}}$$

which equality if $\lambda^{1/k} = \underline{\gamma}$. ∎

[5] At least one $\mu_{uv\sigma}$ must be nonzero.

In some sense, Lemma 8 is encoding a trajectory in the measures $\mu_{uv\sigma}$. We say that the resulting measures are the *occupation measures* of the trajectory $x_0, x_1, \dots, x_k$ defined in Lemma 8.

Example 9. Consider the unconstrained system [1, Example 2.1] with $m = 2$:

$$\mathcal{A} = \{A_1 = e_1 e_2^\top, A_2 = e_2 e_1^\top\}$$

where e_i denotes the ith canonical basis vector.

A solution to Program 3 is given by $(f(x), \overline{\gamma}) = (\|x\|_2, 1)$. This means that $\|x\|_2$ is a Lyapunov function for the system so as it is well known this certifies that $\rho(\mathcal{A}) \leq 1$.

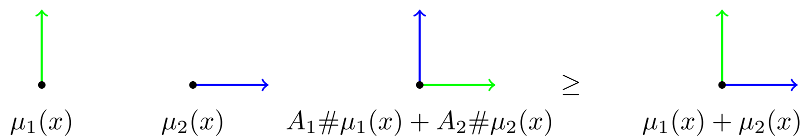

Fig. 2: A representation of the optimal dual solution of Example 9 with the constraint (9).

A dual solution μ_1 (resp. μ_2)[6] for the first (resp. second) matrix has the measure $\mu_1 = \delta_{(0,1)}/2$ (resp. $\mu_2 = \delta_{(1,0)}/2$). This is the solution obtained by applying Lemma 8 to the cycle $(1, 2)$. This is shown in Figure 2.

Remark 10. Occupation measures for continuous switched systems are studied in [11]. These measures are supported on the cartesian product of the state space and a finite interval of time $t \in [0, T]$ while in this paper, the measures are only supported on the subset $\mathbb{S}^{n-1}$ of the state space. Indeed, since the system (1) is homogeneous and time-invariant, we can encode trajectories in a measure on $\mathbb{S}^{n-1}$ (Lemma 8) and still be able to recover it (Corollary 6).

The measures studied in [19] are supported on the paths in G. They are related to the measures studied in this paper since given a cycle c, we can compute the occupation measures of the trajectory using this switching cycle and starting with a leading eigenvector of A_c as x_0 with Lemma 8.

One may wonder whether Lemma 8 also works in the reverse direction to give a *constructive* proof for Corollary 6 when the measures $\mu_{uv\sigma}$ are atomic. Namely, can we extract a periodic trajectory of period c with $\rho(c) \geq \underline{\gamma}$ from any atomic feasible solution of Program 4 with $\underline{\gamma}$. As such solution may be the convex hull of solutions obtained by the construction of Lemma 8, we may recover several periodic trajectory, from which there might be only one that satisfies $\rho(c) \geq \underline{\gamma}$. The following Lemma provides a constructive way to recover a periodic trajectory of period c satisfying $\rho(c) \geq \underline{\gamma}$ in the scalar case[7], i.e. $n = 1$

Lemma 11. *Consider a finite set of matrices $\mathcal{A} \subseteq \mathbb{R}^{1\times 1}$ constrained by an automaton G. If there exists a feasible solution μ of Program 4 with $\underline{\gamma}$, then there exists a cycle c with $\rho(c) \geq \underline{\gamma}$.*

Proof. Let (μ, γ) be the solution. By (10) and (9), we can find a cycle c for which each edge e has a nonzero measure μ_e.

[6] In the arbitrary switching case, we write μ_σ instead of $\mu_{uv\sigma}$ for short

[7] Note that in this case, any measure is atomic since $\mathbb{S}^{n-1}$ is zero-dimensional

If $\rho(c) \geq \gamma$, we are done. Otherwise, if $\rho(c) < \gamma$, using Lemma 8, we can build a feasible solution ν such that (9) is satisfied with equality for $\underline{\gamma} = \rho(c)$. This means that $\mu - \lambda\nu$ is feasible with γ for any $\lambda \geq 0$ such that $\mu - \lambda\nu \geq 0$. Let $\lambda^\star$ be the maximum value of λ such that $\mu - \lambda\nu \geq 0$. Since $n = 1$, $\mathbb{S}^{n-1}$ is zero dimensional so for at least one edge e of the cycle c, $\mu_e - \lambda^\star \nu_e$ is zero. Moreover, since μ_e is nonzero for all edge e of the cycle, $\lambda > 0$. Therefore, the number of edges with nonzero measure has decreased and at least one of the constraints (9) is now satisfied with strict inequality.

This process can only be repeated finitely many times until μ becomes the trivial solution since the number of edges with nonzero measure decrease each time. Moreover we will have $\rho(c) \geq \gamma$ at least once since the constraints (9) cannot be satisfied with strict inequality for the trivial solution. ∎

Given a feasible solution of Program 4 and a common partition of the support of the measures, we show in Proposition 12 how to transform the solution into a solution of a scalar switched system. Using this transformation, we can always recover a cycle c for which $\rho(c) = \gamma$ from a solution of Program 4 with $\underline{\gamma} = \gamma$ for which the measures are atomic.

PROPOSITION 12. *Consider a finite set of matrices $\mathcal{A}$ constrained by an automaton $G(V, E)$. Suppose that there exists a feasible solution μ of Program 4 with $\underline{\gamma} = \gamma$ and a finite family $\mathcal{S}$ of disjoint subsets of $\mathbb{S}^{n-1}$ such that the support of each measure is included in the union of the sets of the family $\mathcal{S}$. Then there exists sets $B_1, \ldots, B_k \in \mathcal{S}$ and a cycle $\sigma_1, \ldots, \sigma_k$ of G such that*

$$\prod_{i=1}^{k} \max_{x \in B_i} \|A_{\sigma_i} x\|_2 \geq \gamma^k$$

and $A_{\sigma_i} B_i \cap B_{i+1} \neq \emptyset$ for $i = 1, \ldots, k$ where $B_{k+1} = B_1$.

Proof. Given a set $B \in \mathcal{S}$ and an edge $e \in E$, let μ_e^B denote the measure defined as $\mu_e^B(C) = \mu_e(C \cap B)$. We consider a new constrained switched system with matrices $\mathcal{A}' \subseteq \mathbb{R}^{1\times 1}$ and automaton $G'(V', E')$ where $V' = \{\,(v, B) \mid v \in V, B \in \mathcal{S}\,\}$, $e'((u, v, \sigma), B, C) = ((u, B), (v, C), (\sigma, B))$, $E' = \{e'(e, B, C) \mid e \in E, B, C \in \mathcal{S}, A_e B \cap C \neq \emptyset\}$, and $A'_{(\sigma,B)} = \max_{x \in B} \|A_\sigma x\|_2$. From any solution μ of the original system feasible for $\underline{\gamma}$, the following solution of the system with matrices $\mathcal{A}'$ and automaton G'

$$\mu'_{e'(e,B,C)} = \frac{(A_e \# \mu_e^B)(C)}{(A_e \# \mu_e^B)(\mathbb{S}^{n-1})} \mu_e(B)$$

is also feasible with $\underline{\gamma}$. Indeed, by construction, for any $v \in V, C \in \mathcal{S}$, we have

$$\sum_{e \in E_1^-(v), B \in \mathcal{S}} (A_e \# \mu_e^B)(C) = \sum_{e \in E_1^-(v), B \in \mathcal{S}} \mu'_{e'(e,B,C)} \frac{(A_e \# \mu_e^B)(\mathbb{S}^{n-1})}{\mu_e(B)} \stackrel{(6)}{\leq} \sum_{e \in {E'}_1^-(v,C)} A'_e \# \mu'_e \tag{11}$$

$$\sum_{e \in E_1^+(v)} \mu_e(C) = \sum_{e \in E_1^+(v), D \in \mathcal{S}} \frac{(A_e \# \mu_e^C)(D)}{(A_e \# \mu_e^C)(\mathbb{S}^{n-1})} \mu_e(C) = \sum_{e \in {E'}_1^+(v,C)} \mu'_e. \tag{12}$$

By (9) on μ, the left-hand side of (12) is smaller than the left-hand side of (11). Therefore, the right-hand side of (12) is smaller than the right-hand side of (11) hence μ' satisfies (9) on the new switched system.

Therefore, by Lemma 11, there is a cycle $(\sigma_1, B_1), \dots, (\sigma_k, B_k)$ of G' such that the modes σ_i and sets B_i are as required. ∎

Example 13. Consider the dual solution obtained in Example 9.

The supports of μ_1 and μ_2 are respectively $B_1 = \{(0,1)\}$ and $B_2 = \{(1,0)\}$. The automaton $G'(V', E')$ obtained by the transformation of Proposition 12 is defined by $V' = \{(1, B_1), (1, B_2)\}$ and $E' = \{((1, B_1), (1, B_2), (1, B_1)), ((1, B_2), (1, B_1), (2, B_2))\}$. The new 1×1 matrices are $A'_{(1,B_1)} = 1$ and $A'_{(2,B_2)} = 1$.

The computation of the CJSR of this scalar system is a *maximum cycle mean* problem as outlined in [1]. The cycle of maximum geometric mean is $((1, B_1), (2, B_2))$ which geometric mean $\sqrt{1 \cdot 1} = 1$. We recover the cycle $(1, 2)$ found in Example 9.

3. Sum of Squares implementation and algorithmic aspects.

In this section, we show how to approximate the pair of Program 3 and Program 4 using sum of squares programming. While the dual formulation is a relaxation of Program 4, we show in Section 3.4 how Proposition 12 can still be used in some cases and we give a rounding procedure in Section 3.5 that generates an infinite sequence of guaranteed growth rate from a feasible dual solution. We show how these methods can be used to find lower bounds to the CJSR in Section 3.6.

3.1. Sum of squares programming.

Deciding whether a multivariate polynomial of degree $2d \geq 4$ is nonnegative is known to be NP-hard. However a sufficient condition for a polynomial to be nonnegative is easy to check. We say that a polynomial is a *sum of squares* (SOS) if there exist polynomials $q_1, \dots, q_M$ such that

$$p(x) = \sum_{k=1}^{M} q_k^2(x).$$

If a polynomial is SOS, then it is obviously nonnegative.

It is well known that if $p(x)$ is an homogeneous polynomial of degree $2d$ then each $q_k(x)$ must be an homogeneous polynomial of degree d; this can be shown easily using the Newton polytope of $p(x)$ and [40, Theorem 1]. Let $x^{[d]}$ represent a basis of the homogeneous polynomials of degree d. We can check whether a polynomial is SOS using semidefinite programming thanks to the following theorem.

THEOREM 14 ([10, 34, 35, 37, 42]). *A homogeneous multivariate polynomial $p(x)$ of degree $2d$ is a sum of squares if and only if*

$$p(x) = (x^{[d]})^\top Q x^{[d]}$$

where Q is a symmetric positive semidefinite matrix.

We denote the set of homogeneous polynomials of degree $2d$ as $\mathbb{R}_{\mathbf{x}}$, the cone of homogeneous SOS polynomials of degree $2d$ as $\Sigma_{\mathbf{2d}}$ and the dual of $\Sigma_{\mathbf{2d}}$ as $\Sigma_{\mathbf{2d}}^*$.

3.2. Moments.

A common interpretation of the dual space $\mathbb{R}_{\mathbf{2d}}^*$ of linear functionals on homogeneous polynomials of degree $2d$ is the space of moments of monomials of degree $2d$; see [7, Section 3.5] and [25]. If $p(x) = a^\top x^{[d]}$ and m is the vector of moments of $x^{[d]}$ of a measure μ then

$$\langle m, a \rangle = \int p(x) \, \mathrm{d}\mu = \langle \mu, p \rangle.$$

As a sum of squares polynomial is nonnegative, this integral is nonnegative for any measure μ. Therefore, given a moment vector m, a necessary condition for a measure to exist with these moments is that $\langle m, a \rangle \geq 0$ for any vector of coefficients a of a sum of squares polynomial. That is, $\Sigma_{\mathbf{2d}}^*$ is a superset of the set of moments of measures. The members of $\Sigma_{\mathbf{2d}}^*$ are often called *pseudo-measures* and denoted $\widetilde{\mu}$; see [4].

Given a program on measures such as Program 4, the *moment relaxation* consists in truncating the infinite moment series to the finite set of moments of the monomials in the matrix $M = (x^{[d]})(x^{[d]})^\top$. Let Q be the matrix such that $Q_{i,j}$ is the moment of the monomial $M_{i,j}$. The constraint that a measure exists with these moments is relaxed to a semidefinite constraint on Q, which is in fact equivalent to requiring that the measure belongs to the cone $\Sigma_{\mathbf{2d}}^*$ introduced above.

3.3. CJSR Approximation via SOS. In this section, we survey and summarize recent methods that approximate the CJSR using SOS programming.

The $2d$th root of homogeneous polynomials of degree $2d$ can be used as Lyapunov function.

THEOREM 15 ([36, 38]). *Consider a finite set of matrices $\mathcal{A}$ constrained by an automaton $G(V, E)$. Suppose that there exist $|V|$ strictly positive homogeneous polynomials $p_v(x)$ of degree $2d$ such that*

$$p_v(A_\sigma x) \leq \overline{\gamma}^{2d} p_u(x)$$

holds for all edge $(u, v, \sigma) \in E$. Then $\rho(G, \mathcal{A}) \leq \overline{\gamma}$.

Proof. Define $f_v(x) = [p_v(x)]^{\frac{1}{2d}}$ and use Theorem 33. ∎

We relax the positivity condition of Theorem 15 by the more tractable sum of squares (SOS) condition and define $\rho_{\text{SOS-}2d}(G, \mathcal{A})$ as the solution of the following SOS restriction of Program 3.

Program 16 (Primal).
Input: A finite set of matrices $\mathcal{A}$ and an automaton G.
Output: Polynomials $p_v(x)$ and a number $\overline{\gamma}$.

$$\begin{aligned}
&\inf_{p_v(x) \in \mathbb{R}_{\mathbf{x}}, \overline{\gamma} \in \mathbb{R}} \overline{\gamma} \\
&\text{subject to } \overline{\gamma}^{2d} p_u(x) - p_v(A_\sigma x) \text{ is SOS}, \quad \forall (u, v, \sigma) \in E, \\
&\qquad p_v(x) \text{ is SOS}, \quad \forall v \in V, \qquad (13)\\
&\qquad p_v(x) \text{ is strictly positive}, \quad \forall v \in V, \qquad (14)\\
&\sum_{v \in V} \int_{\mathbb{S}^{n-1}} p_v(x) \, \mathrm{d}x = 1.
\end{aligned}$$

Remark 17. In practice we can replace (13) and (14) by "$p_v(x) - \epsilon \|x\|_2^{2d}$ is SOS" for any $\epsilon > 0$. This constrains $p_v(x)$ to be in the interior of the SOS cone, which is sufficient for $p_v(x)$ to be strictly positive.

By Theorem 15, a feasible solution of Program 16 gives an upper bound for $\rho(G, \mathcal{A})$, and thus, for any positive degree $2d$,

$$\rho(G, \mathcal{A}) \leq \rho_{\text{SOS-}2d}(G, \mathcal{A}). \qquad (15)$$

Example 18. Consider the unconstrained system [1, Example 2.1] with $m = 3$:

$$\mathcal{A} = \{A_1 = e_1 e_2^\top, A_2 = e_2 e_3^\top, A_3 = e_3 e_1^\top\}$$

where e_i denotes the ith canonical basis vector.

For any d, a solution to Program 16 is given by

$$(p(x), \gamma) = (x_1^{2d} + x_2^{2d} + x_3^{2d}, 1).$$

Example 19. Let us reconsider our running example; see Example 1. The optimal solution of Program 16 is represented by Figure 3 for $2d = 2$ and 12.

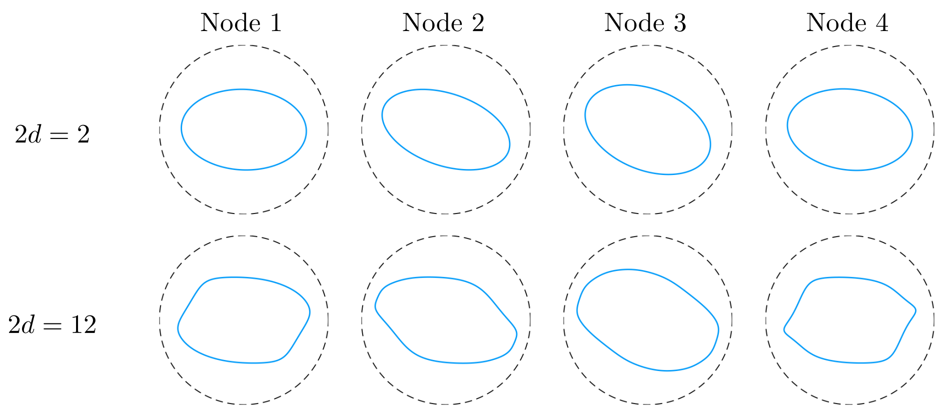

Fig. 3: Representation of the solutions to Program 16 with different values of d for the running example. The blue curve represents the boundary of the 1-sublevel set of the optimal solution p_v at each node $v \in V$. The dashed curve is the boundary of the unit circle. Observe that some sets are not convex.

3.4. Dual SOS program. In Section 3.3, we introduced the SOS restriction of Program 3 with Program 16. In Section 3.4, we introduce Program 20, the moment relaxation of Program 4. It turns out that Program 16 and Program 20 are dual to each other. Indeed, the proof of Lemma 34 can be translated verbatim in order to prove that Program 20 is the dual of Program 16.

Program 20 (Dual of Program 16).
Input: A finite set of matrices $\mathcal{A}$ and an automaton G.
Output: Pseudo-measures $\widetilde{\mu}_{uv\sigma}$ and a number $\underline{\gamma}$.

$$\sup_{\widetilde{\mu}_{uv\sigma}\in\mathbb{R}^*_{\mathbf{2d}},\,\underline{\gamma}\in\mathbb{R}} \underline{\gamma}$$

$$\text{subject to} \quad \sum_{(u,v,\sigma)\in E} A_\sigma \# \widetilde{\mu}_{uv\sigma} - \underline{\gamma}^{2d} \sum_{(v,w,\sigma)\in E} \widetilde{\mu}_{vw\sigma} \in \Sigma^*_{\mathbf{2d}}, \quad \forall v \in V, \tag{16}$$

$$\widetilde{\mu}_{uv\sigma} \in \Sigma^*_{\mathbf{2d}}, \quad \forall (u,v,\sigma) \in E, \tag{17}$$

$$\sum_{(u,v,\sigma)\in E} \widetilde{\mu}_{uv\sigma}(\mathbb{S}^{n-1}) = 1. \tag{18}$$

It is important to note that a solution of Program 20 is not necessarily a solution of Program 4. First $\widetilde{\mu}_{uv\sigma}$ may not be a measure even if it belongs to $\Sigma^*_{\mathbf{2d}}$ as discussed in Section 3.1. Second, the left-hand side of (16) may also not be a measure. For this second concern, it helps to be more explicit. Suppose for instance that we are in the quadratic case, i.e. $d = 1$. In that case, if $\widetilde{\mu} \in \Sigma_{\mathbf{2}}^*$, there always exists a measure μ that has the moments of the pseudo-measure $\widetilde{\mu}$. We can take for instance a Gaussian distribution with these second order moments. Hence we can find Gaussian distributions $\mu_{uv\sigma}$ that have the second order moments $\widetilde{\mu}_{uv\sigma}$ and Gaussian distributions ν_v that have the second order moments given by the left-hand side of (16). However, we may have

$$\sum_{(u,v,\sigma)\in E} A_\sigma \# \mu_{uv\sigma} - \underline{\gamma}^{2d} \sum_{(v,w,\sigma)\in E} \mu_{vw\sigma} \neq \nu_v$$

as we only know that the left-hand side and right-hand side of the above equation have the same second order moments; see Example 23.

However, in some cases, we can recover a feasible solution of Program 4 from a feasible solution of Program 20. In these cases, by Corollary 6, this provides a lower bound on the CJSR. Moreover, there exist efficient techniques allowing to detect situations where the solution is moments of an atomic measure; see [20, 26]. Then, using the transformation of Proposition 12, we can transform these atomic measures into a feasible solution of a constrained scalar switched systems. For such system, we could use the algorithm described in Lemma 11 but as pointed out in [1], computing the CJSR of a scalar system can easily be done by solving a maximum cycle mean problem for which efficient algorithm exists [23].

If we recover a feasible solution of Program 4 from a feasible solution of Program 20 with $\underline{\gamma} = \rho_{\text{SOS-}2d}(G, \mathcal{A})$, we can directly conclude that $\rho_{\text{SOS-}2d}(G, \mathcal{A}) = \rho(G, \mathcal{A})$. This is somewhat similar to the minimization of a multivariate polynomial using SOS where we can detect that we have reached the optimum when the measure is atomic and recover the minimizers of the polynomial from the atoms of the measure.

However, we may also check for atomic feasible solutions of Program 4 with $\underline{\gamma} < \rho_{\text{SOS-}2d}(G, \mathcal{A})$ to provide lower bounds. Moreover, in practice, $\rho_{\text{SOS-}2d}(G, \mathcal{A})$ is computed by binary search on $\underline{\gamma}$ so we often have several such solutions.

Example 21. Consider Example 18. For $i = 1, 2, 3$, let $\widetilde{\mu}_i$ be the solution of Program 20 corresponding to the matrix A_i. For any d, we can see that the dual solution for $\gamma = 1$ is such that the only monomial x^α such that $\langle \widetilde{\mu}_1, x^\alpha \rangle$ (resp. $\langle \widetilde{\mu}_2, x^\alpha \rangle$, $\langle \widetilde{\mu}_3, x^\alpha \rangle$) is non-zero is x_1^{2d} (resp. x_2^{2d}, x_3^{2d}) and $\langle \widetilde{\mu}_1, x_1^{2d} \rangle = \langle \widetilde{\mu}_2, x_2^{2d} \rangle = \langle \widetilde{\mu}_3, x_3^{2d} \rangle = 1/3$. Note that it means that $\widetilde{\mu}_1 = \delta_{(1,0,0)}/3$, $\widetilde{\mu}_2 = \delta_{(0,1,0)}/3$ and $\widetilde{\mu}_3 = \delta_{(0,0,1)}/3$ where δ_x is the Dirac measure centered on x. Since these measures are solution to Program 4 with $\underline{\gamma} = 1$, by Corollary 6, this means that $\rho(\mathcal{A}) \geq 1$.

Example 22. We consider [36, Example 2.8]:

$$A_1 = \begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix} \qquad A_2 = \begin{pmatrix} 0 & 1 \\ 0 & -1 \end{pmatrix}.$$

to illustrate the fact that this atom extraction procedure can be used to determine when the upper bound found by Program 16 is equal to the CJSR. In this unconstrained example, the JSR is one but the upper bound found by Program 16 for $d = 1$ is $\sqrt{2}$. However, for $d = 2$, the upper bound found is 1 and the solutions of Program 20 for $\gamma = 1$ is

$$\mu_1 = 0.59698\delta_{(1,1)} + 0.59513\delta_{(1,-1)} \qquad \mu_2 = 0.59513\delta_{(1,1)} + 0.59322\delta_{(1,-1)}.$$

Since $A_1 \# \delta_{(1,1)} = \delta_{(1,1)}$, the cycle extraction method immediately find the cycle $c = 1$ for which $\rho(A_c) = 1$.

Example 23. We continue the running example; see Example 1 and Example 19.

For all d, $\widetilde{\mu}_{212} = \widetilde{\mu}_{323} = \widetilde{\mu}_{344} = \widetilde{\mu}_{431} = 0$ hence the node 4 is "unused" by the dual. For $2d = 2, 4$, $\widetilde{\mu}_{123} = \widetilde{\mu}_{231} = 0$ so the node 2 is "unused" for low degree.

At first, one could think that the dual variables can be used to reduce the systems, e.g. remove nodes or edges. However, it would be a mistake to remove the node 2 since the periodic trajectory with highest growth rate uses this node.

It is also interesting to notice that the matrices corresponding to the dual variables have low rank. For example, for $2d = 2$, $\widetilde{\mu}_{131}$ (resp. $\widetilde{\mu}_{312}$, $\widetilde{\mu}_{331}$) is the Dirac measure $5.873 \cdot \delta_{(0.917,0.399)}$ (resp. $3.966 \cdot \delta_{(0.875,0.485)}$, $6.704 \cdot \delta_{(0.757,-0.653)}$). However, this is not a feasible solution of Program 4. Indeed, while (9) is satisfied for node 1 since $A_2\#\delta_{(0.875,0.485)}$ gives $\delta_{(0.917,0.399)}$, (9) is not satisfied for node 3 as $A_1\#\delta_{(0.917,0.399)}$ gives $\delta_{(0.999,-0.0271)}$ and $A_1\#\delta_{(0.757,-0.653)}$ gives $\delta_{(0.422,-0.906)}$.

3.5. Constructing high growth sequence. In this section we give an algorithm that generates an infinite sequence of matrices such that the asymptotic growth rate of the product of the matrices is arbitrarily close to the CJSR. Note that by Definition 2, this asymptotic growth rate must be smaller than the CJSR.

Given an edge $e \in E$, let $\widetilde{\mathbb{E}}_e[p(x)] = \langle \widetilde{\mu}_e, p(x) \rangle$. Given a polynomial $p_0(x) \in \operatorname{int}(\Sigma_{\mathbf{2d}})$ and an initial edge (v_0, v_{-1}, σ_0), the algorithm builds a $G^\top$-admissible sequence $(v_1, v_0, \sigma_1), (v_2, v_1, \sigma_2), \ldots$ such that

$$\theta_k \triangleq \widetilde{\mathbb{E}}_{v_k v_{k-1} \sigma_k}[p_0(A_{\sigma_1} \cdots A_{\sigma_k} x)] \tag{19}$$

remains "large" for increasing k. As we will see, using Lemma 25, this implies that $A_{\sigma_1} \cdots A_{\sigma_k}$ has a "large" norm.

Lemma 24 ([30, Lemma 6]). *For any polynomial $p(x) \in \operatorname{int}(\Sigma_{\mathbf{2d}})$, there exists a constant $\beta > 0$ such that for any matrix A,*

$$\beta \|A\|_2^{2d} p(x) - p(Ax) \quad \text{is SOS}$$

where $\|A\|_2 = \rho(A^\top A)^{1/2}$ is the Euclidean norm.

Lemma 25. *Let us consider a solution $(\widetilde{\mu}_e : e \in E)$ of Program 20. For any polynomial $p(x) \in \operatorname{int}(\Sigma_{\mathbf{2d}})$, there exists a positive constant τ such that for any matrix $A \in \mathbb{R}^{n \times n}$ and edge $e \in E$,*

$$\widetilde{\mathbb{E}}_e[p(Ax)] \leq \tau \|A\|_2^{2d}$$

Proof. If all pseudo-expectations are zero, the result is trivially true. Therefore we can suppose that at least one is nonzero. By Lemma 24, there exists a constant $\beta > 0$ such that

$$\beta \|A\|_2^{2d} p(x) - p(Ax) \text{ is SOS.}$$

Hence for any edge $e \in E$,

$$\widetilde{\mathbb{E}}_e[p(Ax)] \leq \beta \|A\|_2^{2d} \widetilde{\mathbb{E}}_e[p(x)].$$

We obtain the result with the constant $\tau = \beta \max_{e \in E} \widetilde{\mathbb{E}}_e[p(x)]$. Since at least one pseudo-expectation is nonzero and $p(x)$ is in the interior of the SOS cone, $\tau > 0$. ☐

Lemma 27 provides a guarantee on the growth rate of θ_k, defined in (19), using the dual constraint (16).

Algorithm 1 Generates a sequence of large asymptotic growth.

Data: Length of subpaths: $l \in \mathbb{N}$; degree: $d \in \mathbb{N}$; lower bound to $\rho_{\text{SOS-}2d}(G, \mathcal{A})$: $0 < \gamma < \rho_{\text{SOS-}2d}(G, \mathcal{A})$; and feasible solution $(\,\widetilde{\mu}_e : e \in E\,)$ of Program 20 with $\underline{\gamma} = \gamma$ and degree d.
Result: Sequence of arbitrary length $s = (\ldots, v_k, \sigma_k, \ldots, v_0, \sigma_0, v_{-1})$.
Pick an arbitrary polynomial $p_0(x) \in \text{int}(\Sigma_{\mathbf{2d}})$
Pick an edge $(v_0, v_{-1}, \sigma_0) \in E$ such that $\widetilde{\mu}_{v_0 v_{-1} \sigma_0}$ is nonzero
for $k = 0, l, 2l, \ldots$ **do**
  Pick $s \in \arg\max_{s \in E_l^-(v_k)} \widetilde{\mathbb{E}}_{s[1]}[p_k(A_s x)]$
  Set $(v_{k+l}, \sigma_{k+l}, \ldots, \sigma_{k+1}, v_k) \leftarrow s$
  Set $p_{k+l} \leftarrow p_k(A_s x)$
end for

Lemma 26. *Given a finite set of matrices $\mathcal{A}$ constrained by an automaton G, if $\widetilde{\mu}$ is a feasible solution of Program 20 then, for any edge $(\bar{u}, \bar{v}, \bar{\sigma}) \in E$, the following holds:*

$$\sum_{s \in E_k^-(\bar{u})} A_s \# \widetilde{\mu}_{s[1]} \succeq \gamma^{2dk} \widetilde{\mu}_{\bar{u}\bar{v}\bar{\sigma}} \tag{20}$$

where $\widetilde{\mu}_1 \succeq \widetilde{\mu}_2$ denotes $\widetilde{\mu}_1 - \widetilde{\mu}_2 \in \Sigma^*_{\mathbf{2d}}$.

Proof. We prove (20) by induction, the case of $k = 0$ being trivial. Suppose that

$$\sum_{s' \in E_{k-1}^-(\bar{u})} A_{s'} \# \widetilde{\mu}_{s'[1]} \succeq \gamma^{2d(k-1)} \widetilde{\mu}_{\bar{u}\bar{v}\bar{\sigma}}. \tag{21}$$

We can rewrite the left-hand side of (20) as

$$\sum_{s \in E_k^-(\bar{u})} A_s \# \widetilde{\mu}_{s[1]} = \sum_{s' \in E_{k-1}^-(\bar{u})} A_{s'} \# \sum_{(u, s'(1), \sigma) \in E} A_\sigma \# \widetilde{\mu}_{us'(1)\sigma}. \tag{22}$$

By (16),

$$\sum_{(u, s'(1), \sigma) \in E} A_\sigma \# \widetilde{\mu}_{us'(1)\sigma} \succeq \gamma^{2d} \sum_{(s'(1), w, \sigma') \in E} \widetilde{\mu}_{s'(1)w\sigma'}.$$

Since the dual variables $\widetilde{\mu}_{s'(1)w\sigma'}$ of the right-hand side are in the dual of the SOS cone, and one of them is $\widetilde{\mu}_{s'[1]}$, we have

$$\sum_{(u, s'(1), \sigma) \in E} A_\sigma \# \widetilde{\mu}_{us'(1)\sigma} \succeq \gamma^{2d} \widetilde{\mu}_{s'[1]}.$$

Applying $A_{s'}\#$ on both sides and using (22) gives

$$\sum_{s \in E_k^-(\bar{u})} A_s \# \widetilde{\mu}_{s[1]} \succeq \gamma^{2d} \sum_{s' \in E_{k-1}^-(\bar{u})} A_{s'} \# \widetilde{\mu}_{s'[1]} \overset{(21)}{\succeq} \gamma^{2dk} \widetilde{\mu}_{\bar{u}\bar{v}\bar{\sigma}}.$$

∎

Lemma 27. *Consider a finite set of matrices $\mathcal{A}$ constrained by an automaton $G(V, E)$. For any positive integers d and l, using Program 20 with any $\gamma < \rho_{SOS\text{-}2d}$*

$(G,\mathcal{A})$, Algorithm 1 with paths of length l produces a $G^\top$-admissible sequence $(v_1, v_0, \sigma_0), (v_2, v_1, \sigma_1), \dots$ for which the sequence of θ_k defined in (19) satisfies the following inequality for all $k > 0$ multiple of l:

$$\theta_k \geq \frac{\gamma^{2dl}}{d_l^-(v_{k-l+1})}\theta_{k-l}$$

Proof. By Lemma 26,

$$\sum_{s \in E_l^-(v_{k-l+1})} \widetilde{\mathbb{E}}_{s[1]}[p_{k-l}(A_s x)] \geq \gamma^{2dl}\theta_{k-l}.$$

Since the value of s chosen by Algorithm 1 maximises $\widetilde{\mathbb{E}}_{s[1]}[p_{k-l}(A_s x)]$, the left-hand side of the above inequality is smaller or equal to $d_l^-(v_{k-l+1})\theta_k$. ∎

Theorem 28 translates the guarantee on θ_k to a guarantee on $A_{\sigma_1} \cdots A_{\sigma_k}$ using Lemma 25.

THEOREM 28. *Consider a finite set of matrices $\mathcal{A}$ constrained by an automaton $G(V,E)$. For any positive integers d, l and a lower bound $\gamma < \rho_{SOS\text{-}2d}(G,\mathcal{A})$, Algorithm 1 with input l, d and γ produces a $G^\top$-admissible sequence $(v_1, v_0, \sigma_0), (v_2, v_1, \sigma_1), \dots$ that satisfies the following inequality:*

$$\lim_{k\to\infty} \|A_{s_k}\|_2^{\frac{1}{k}} \geq \frac{\gamma}{(\Delta_l^-(G))^{\frac{1}{2dl}}}$$

where $s_k = (\sigma_k, \dots, \sigma_1)$.

Proof. By Lemma 27, for any k multiple of l,

$$\widetilde{\mathbb{E}}_{s_k[1]}[p_0(A_{s_k}x)] \geq \frac{\gamma^{2dk}}{(\Delta_l^-(G))^{\frac{k}{l}}}\widetilde{\mathbb{E}}_{v_0 v_{-1}\sigma_0}[p_0(x)]$$

By Lemma 25, there exists a constant $\tau > 0$ such that

$$\widetilde{\mathbb{E}}_{s_k[1]}[p_0(A_{s_k}x)] \leq \tau\|A_{s_k}\|^{2d}.$$

Combining these two inequalities, we obtain

$$\tau\|A_{s_k}\|^{2d} \geq \frac{\gamma^{2dk}}{(\Delta_l^-(G))^{\frac{k}{l}}}\widetilde{\mathbb{E}}_{v_0 v_{-1}\sigma_0}[p_0(x)].$$

Since $\widetilde{\mathbb{E}}_{v_0 v_{-1}\sigma_0}$ is nonzero, $\widetilde{\mathbb{E}}_{v_0 v_{-1}\sigma_0}[p_0(x)] > 0$. Therefore taking the $(2dk)$th root and the limit $k \to \infty$ we obtain the result. ∎

Example 29. Suppose that we apply Algorithm 1 with $l = 1$ to Example 21 and let us denote by c_α the coefficient of the monomial x^α in the polynomial $p_0(x)$ chosen arbitrarily by the algorihtm. The start of the sequence produced depends on the order between the coefficients $c_{(2d,0,0)}, c_{(0,2d,0)}, c_{(0,0,2d)}$. If $c_{(2d,0,0)}$ is the largest then the G-admissible left-infinite sequence found is $\dots, 1, 2, 3, 1, 2, 3, 1, 2, 3$.

The product $A_{\sigma_1}A_{\sigma_2}A_{\sigma_3}\cdots = A_3A_2A_1A_3A_2A_1\cdots$ is periodic and has an asymptotic growth rate $\rho(A_{\sigma_1}A_{\sigma_2}A_{\sigma_3})^{1/3} = 1$. Hence $1 \leq \rho(G,\mathcal{A})$.

3.6. Deducing a lower bound certificate. By definition of the CJSR, the asymptotic growth rate of the norm of the product of any G-admissible (or $G^\top$-admissible) sequence of matrices gives a lower bound on the CJSR. In particular the sequence produced by Algorithm 1 provides a lower bound on the CJSR.

If there are two integers $\bar{k}, k$ such that the sequence after $\bar{k}$ is periodic of period k, the asymptotic growth rate of the norm is equal to the kth root of the spectral radius of the product of the matrices of one period. This is due to the Gelfand's formula $\rho(A) = \lim_{k\to\infty} \|A^k\|^{1/k}$. From the same identity, we see that the spectral radius of the product of the matrices of one G-admissible cycle gives a lower bound on the CJSR.

To find lower bounds for the CJSR, one could generate all the cycles of length smaller than some maximum length and compute the spectral radius for all of them. This brute force approach is not scalable because the number of paths considered grows exponentially with the maximum length.[8]

Gripenberg [16] proposes a branch-and-bound algorithm that prunes the search using an a priori fixed absolute error. Other alternative methods exist such as the balanced complex polytope algorithm [18, 17] and the invariant conitope algorithm [22]. The methods attempts to generate an invariant polytope from the eigenvector of a cycle of high growth rate. A candidate of cycle of higher growth rate can be found while constructing this polytope, the construction is then restarted with its eigenvector as a new stating point. While computing this polytope, convexity arguments allows to prune paths which attenuate the exponential growth of the number of paths. Specialized methods exist for some particular matrix structures such as the "spectral simplex method" [39] in the case of nonnegative matrices with a "product structure".

These algorithms can also be used to produce a G-admissible sequence of matrices of high asymptotic growth rate by reproducing the cycles of high spectral radius infinitely. The advantage of Algorithm 1 is that it provides a guarantee of accuracy given in Theorem 28. Algorithm 1 provides at the same time a high growth infinite trajectory and lower bounds of guaranteed accuracy.

Algorithm 1 requires to solve a semidefinite program with semidefinite matrices of size $\binom{n+d-1}{d}$. Then, in order to add l new edges to the sequence, it needs to go through $\Delta_l^-(G)$ paths and compare them by computing the scalar product between a polynomial and moments with $\binom{n+2d-1}{2d}$ monomials. The semidefinite program can be solved in a time polynomial in $\binom{n+d-1}{d}$ and $|E|$ [43], and adding l edges to the sequence can be done in a time proportional to $\Delta_l^-(G)\binom{n+2d-1}{2d}$. While polytopes are used in [18, 17, 22] to prune paths, Algorithm 1 uses a solution of Program 20 to guide the search which enables the discovery of sequences of guaranteed high growth rate even with a small value of l. Moreover, Example 30 and [15] give examples where Algorithm 1 uncovers rather long cycles of high asymptotic growth rate. This shows the complementarity of Algorithm 1 with existing approaches which performs better when the cycles of high growth rate have a small length as they iterate over possible cycles of increasing length (although some are pruned). Moreover, [15] shows that the algorithms can handle constrained switched systems with automaton of large size, as it stabilizes a system with 64 nodes and 512 edges [15, Table I].

Example 30. We consider the switched system introduced in [8] as a counterexample to the finiteness conjecture [24]. We use the value $\alpha = 0.7493265463303675$

[8] The exponential growth of the brute force approach is the reason why one should choose a small l for Algorithm 1.

length	cycle	growth rate
13	2112112121121	1.4092472220583443
21	211211212112112121121	1.4092472220583487

Table 1: Two cycles of high growth rate for the switched system of Example 30.

which is the IEEE double-precision number that is closest to the value given in [19] for which the system does not satisfy the finiteness property. Table 1 gives two cycles of high growth rate; as the reader can check, their growth rates are rather close. We verified that there exists no cycle of length up to 32 that provides a larger lower bound.

Using Algorithm 1 with $2d = 2$, $l = 4$ and p_0 equal to the solution of Program 16, the algorithm generates a sequence starting with the cycle of length 21 in Table 1.

We consider now the Balanced Polytope algorithm exploiting the nonnegativity of the matrices [17, Section 4]. A point p is considered to belong to the interior of a balanced polytope P if Mosek [3] with its simplex algorithm certifies that the maximal t such that $tp \in P$ is larger than $1 + 1 \times 10^{-13}$. The algorithm first finds the cycle of length 13 in Table 1 and is then able to find the cycle of length 21 with the tolerance 1×10^{-13}. However, if the 1×10^{-13} tolerance is replaced by -1×10^{-12} or lower, then the algorithm does not find this second cycle as shown in [33]. This behavior is not surprising given how close the growth rates are as shown in Table 1.

A cycle with growth rate equal to the CJSR is often called *spectral maximizing product* (s.m.p.). The algorithm is able to conclude that the cycle of length 21 is an s.m.p. with the tolerance 1×10^{-13}. If we replace the 1×10^{-13} by 1.1×10^{-13}, the algorithm does not conclude that the cycle of length 21 is an s.m.p., even up to depth 1000 as shown in [33]. We will consider this cycle to be an s.m.p. for the purpose of the benchmark even though it cannot be said for certain.

We can compute "non-constructive" lower bounds (it is not constructive as it does not exhibit a cycle certifying the lower bound) using the guarantee (given in [30, Corollary 1]) on the upper bound (15) provided by Program 16, but in practice the trajectories found by Algorithm 1 are periodic after some time $\bar{k}$ so we are able to compute much better lower bounds than the pessimistic bound provided by the guarantee. This is shown by Example 31.

Example 31. We tried the atom extraction procedure introduced in Section 3.4 and Algorithm 1 for $l = 1$ and $l = 3$ on our running example; see Example 1, Example 19 and Example 23. The result is shown in Figure 4. We showed in [30] that the CJSR of the system is equal to 0.97482. We can see that this lower bound is found for $d = 4$ for $l = 1$ and for $d = 1$ for $l = 3$. The atom extraction finds the lower bound 0.939255.

Example 32. Consider the unconstrained switched system with the following two matrices[9]:

$$A_1 = \begin{bmatrix} -1 & 1 & -1 \\ -1 & -1 & 1 \\ 0 & 1 & 1 \end{bmatrix} \qquad A_2 = \begin{bmatrix} -1 & 1 & -1 \\ -1 & -1 & 0 \\ 0 & 1 & 1 \end{bmatrix}.$$

[9]This example was found in a previous collaboration with N. Guglielmi and A. Cicone (unpublished).

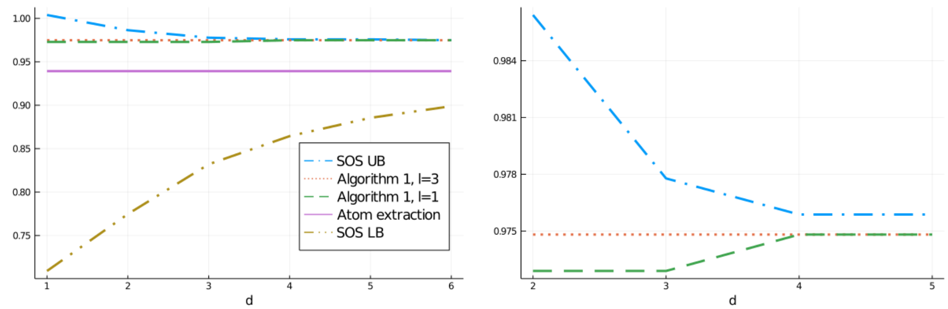

Fig. 4: Result of Example 31. The SOS UB is the upper bound found by Program 16 and the SOS LB is obtained from its guarantee; see [30, Corollary 1]. The value d of horizontal axis corresponds to using polynomials of degree $2d$. The right figure is a zoom of the left figure.

Example	length	GRIP [s]	BP [s]	d	l	SOS [s]	SEQ [s]
30	21	0.072	0.02	1	4	0.06	0.0013
				15	2	0.70	0.013
31	8	0.067	0.10	1	3	0.08	0.0012
				4	1	0.20	0.0012
32	41	0.035	1.81	1	9	0.06	0.113
				4	2	0.62	0.071

Table 2: Comparison of the performance of different algorithms to find an s.m.p. The second column provides the length of the smallest s.m.p. GRIP is the time taken by the Gripenberg algorithm [16] to find an s.m.p. BP is the time taken by the Balanced Polytope algorithm [17] to find an s.m.p. The nonnegativity of the matrices is exploited for Example 30 as suggested in Section 4 of [17]. A point p is considered to belong to the interior of a balanced polytope P if Mosek [3] certifies that the maximal t such that $tp \in P$ is larger than $1 + 1 \times 10^{-13}$. SOS is the time taken by Mosek [3] to solve the pair of primal-dual programs Program 16/Program 20 with degree d using a bisection on γ until $\log(\overline{\gamma}) - \log(\underline{\gamma}) < 1 \times 10^{-2}$ where $\overline{\gamma}$ is the smallest γ such that Program 16 is feasible and $\underline{\gamma}$ is the largest γ such that Program 20 is feasible. SEQ is the time taken by Algorithm 1 with input l, d and $\underline{\gamma}$. The timings are taken from `Benchmark.html` of [33].

The s.m.p. has length 41 and growth rate 1.684185:

$$11122112211221122112211221122112211221112.$$

We summarize in Table 2 and Table 3 the time taken by the different methods on the examples. As we can see in Table 2, the time taken by Algorithm 1 to find the s.m.p. is competitive compared to alternative approach once the Sum-of-Squares pair of primal-dual programs Program 16/Program 20 has been solved. Moreover, as we can see in Table 3, finding upper bounds by solving this pair of programs is competitive with alternative approaches.

4. Conclusions. We have analysed the dual of the SOS Lyapunov program for switched systems and shown how to leverage it to study the system stability. We also

Example	δ	GRIP [s]	BP [s]	d	SOS [s]
30	6×10^{-4}	1.37	2.086	7	0.62
31	25×10^{-8}	6.50	0.046	7	1.28
32	1×10^{-3}	0.38	2.484	6	8.25

Table 3: Comparison of the performance of different algorithms to find an upper bound to the CJSR. GRIP is the time taken by the Gripenberg algorithm [16] to prove the upper bound $\rho(G, \mathcal{A}) + \delta$. The timing BP differs from the timing BP in Table 2 in the fact that we wait for the algorithm to prove that it is an s.m.p. The timing SOS differs from the timing SOS in Table 2 only in the bisection stopping criterion which is $\overline{\gamma} - \rho(G, \mathcal{A}) < \delta$ for this table. The timings are taken from `Benchmark.html` of [33].

generalized the whole approach to *constrained switched systems*, a class of systems that has attracted increasing attention recently.

Our analysis shows (and thrives on) the close relationship between the optimization approach for computing the JSR and the notion of *constrained* switching systems: First, our Theorem 28, which leverages the dual of the classical JSR algorithm, actually naturally applies to the constrained case; Even more, Proposition 12 transforms an unconstrained system into a scalar constrained one for the purpose of computing a lower bound. Let us also mention our work [32, Theorem 5.1], where unconstrained systems with low rank matrices naturally lead to the definition of an auxiliary *constrained* system.

We have introduced two techniques to generate lower bounds from the solution of the SOS dual program. In practice, these techniques provide periodic trajectories of high asymptotic growth rate. Since the SOS program can be solved efficiently, does this give an efficient algorithm to generate lower bounds on the CJSR with *guaranteed accuracy*? This is not clear, because our algorithm provides firm guarantees only when the computed measures are atomic, which is not always the case.

More generally, the techniques developed in this work, based on generating "bad" trajectories for a dynamical system via dual solutions, naturally extend to many other problems in systems theory. We are currently exploring such possibilities.

REFERENCES

[1] A. A. Ahmadi and P. A. Parrilo, *Joint spectral radius of rank one matrices and the maximum cycle mean problem.*, in CDC, 2012, pp. 731–733.

[2] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows: Theory, Algorithms, and Applications*, Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1993.

[3] M. ApS, *Mosek optimization suite release 8.1.0.82.* URL: http://docs.mosek.com/8.1/intro.pdf, 2019.

[4] B. Barak, F. G. Brandao, A. W. Harrow, J. Kelner, D. Steurer, and Y. Zhou, *Hypercontractivity, sum-of-squares proofs, and their applications*, in Proceedings of the forty-fourth annual ACM Symposium on Theory of Computing, ACM, 2012, pp. 307–326.

[5] M. A. Berger and Y. Wang, *Bounded semigroups of matrices*, Linear Algebra and its Applications, 166 (1992), pp. 21–27.

[6] J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah, *Julia: A fresh approach to numerical computing*, SIAM review, 59 (2017), pp. 65–98.

[7] G. Blekherman, P. A. Parrilo, and R. R. Thomas, *Semidefinite Optimization and Convex Algebraic Geometry*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 2012, https://arxiv.org/abs/http://epubs.siam.org/doi/pdf/10.1137/1.9781611972290.

[8] V. D. Blondel, J. Theys, and A. A. Vladimirov, *An elementary counterexample to the finiteness conjecture*, SIAM Journal on Matrix Analysis and Applications, 24 (2003), pp. 963–

970.

[9] V. D. Blondel and J. N. Tsitsiklis, *The boundedness of all products of a pair of matrices is undecidable*, Systems & Control Letters, 41 (2000), pp. 135–140.

[10] M.-D. Choi, T. Y. Lam, and B. Reznick, *Sums of squares of real polynomials*, in Proceedings of Symposia in Pure mathematics, vol. 58, American Mathematical Society, 1995, pp. 103–126.

[11] M. Claeys, J. Daafouz, and D. Henrion, *Modal occupation measures and lmi relaxations for nonlinear switched systems control*, Automatica, 64 (2016), pp. 143–154.

[12] X. Dai, *A Gel'fand-type spectral radius formula and stability of linear constrained switching systems*, Linear Algebra and its Applications, 436 (2012), pp. 1099–1113.

[13] I. Dunning, J. Huchette, and M. Lubin, *JuMP: A modeling language for mathematical optimization*, SIAM Review, 59 (2017), pp. 295–320.

[14] L. Elsner, *The generalized spectral-radius theorem: an analytic-geometric proof*, Linear Algebra and its Applications, 220 (1995), pp. 151–159.

[15] C. Gomes, R. M. Jungers, B. Legat, and H. Vangheluwe, *Minimally constrained stable switched systems and application to co-simulation*, in 57th IEEE Conference on Decision and Control, IEEE, 2018.

[16] G. Gripenberg, *Computing the joint spectral radius*, Linear Algebra and its Applications, 234 (1996), pp. 43–60.

[17] N. Guglielmi and V. Protasov, *Exact computation of joint spectral characteristics of linear operators*, Foundations of Computational Mathematics, 13 (2013), pp. 37–97.

[18] N. Guglielmi and M. Zennaro, *An algorithm for finding extremal polytope norms of matrix families*, Linear Algebra and its Applications, 428 (2008), pp. 2265–2282.

[19] K. G. Hare, I. D. Morris, N. Sidorov, and J. Theys, *An explicit counterexample to the Lagarias-Wang finiteness conjecture*, Advances in Mathematics, 226 (2011), pp. 4667 – 4701, https://doi.org/http://dx.doi.org/10.1016/j.aim.2010.12.012, http://www.sciencedirect.com/science/article/pii/S0001870810004457.

[20] D. Henrion and J.-B. Lasserre, *Detecting global optimality and extracting solutions in gloptipoly*, in Positive polynomials in control, Springer, 2005, pp. 293–310.

[21] R. Jungers, *The joint spectral radius: theory and applications*, vol. 385, Springer Science & Business Media, 2009.

[22] R. M. Jungers, A. Cicone, and N. Guglielmi, *Lifted polytope methods for computing the joint spectral radius*, SIAM Journal on Matrix Analysis and Applications, 35 (2014), pp. 391–410.

[23] R. M. Karp, *A characterization of the minimum cycle mean in a digraph*, Discrete Mathematics, 23 (1978), pp. 309 – 311, https://doi.org/http://dx.doi.org/10.1016/0012-365X(78)90011-0, http://www.sciencedirect.com/science/article/pii/0012365X78900110.

[24] J. C. Lagarias and Y. Wang, *The finiteness conjecture for the generalized spectral radius of a set of matrices*, Linear Algebra and its Applications, 214 (1995), pp. 17 – 42, https://doi.org/http://dx.doi.org/10.1016/0024-3795(93)00052-2, http://www.sciencedirect.com/science/article/pii/0024379593000522.

[25] J. B. Lasserre, *Moments, positive polynomials and their applications*, World Scientific, 2009.

[26] M. Laurent, *Sums of squares, moment matrices and optimization over polynomials*, in Emerging applications of algebraic geometry, Springer, 2009, pp. 157–270.

[27] B. Legat, C. Coey, R. Deits, J. Huchette, and A. Perry, *Sum-of-squares optimization in Julia*, in The First Annual JuMP-dev Workshop, 2017.

[28] B. Legat, M. Forets, and C. Schilling, *blegat/HybridSystems.jl: v0.3.0*, May 2019, https://doi.org/10.5281/zenodo.1246104.

[29] B. Legat and C. Gomes, *blegat/SwitchOnSafety.jl: v0.0.4*, Oct. 2019, https://doi.org/10.5281/zenodo.3234046.

[30] B. Legat, R. M. Jungers, and P. A. Parrilo, *Generating unstable trajectories for Switched Systems via Dual Sum-Of-Squares techniques*, in Proceedings of the 19th International Conference on Hybrid Systems: Computation and Control, HSCC '16, ACM, 2016, pp. 51–60, https://doi.org/10.1145/2883817.2883821, http://doi.acm.org/10.1145/2883817.2883821.

[31] B. Legat, R. M. Jungers, P. A. Parrilo, and P. Tabuada, *Set Programming with JuMP*, in The Third Annual JuMP-dev Workshop, 2019.

[32] B. Legat, P. A. Parrilo, and R. M. Jungers, *Certifying unstability of Switched Systems using Sum of Squares Programming*, ArXiv e-prints, (2017), https://arxiv.org/abs/1710.01814.

[33] B. Legat, P. A. Parrilo, and R. M. Jungers, *Certifying unstability of switched systems using sum of squares programming*. https://www.codeocean.com/, May 2020, https://doi.org/10.24433/CO.9148109.v2.

[34] Y. Nesterov, *Squared functional systems and optimization problems*, in High performance optimization, Springer, 2000, pp. 405–440.
[35] P. A. Parrilo, *Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization*, PhD thesis, Citeseer, 2000.
[36] P. A. Parrilo and A. Jadbabaie, *Approximation of the joint spectral radius using sum of squares*, Linear Algebra and its Applications, 428 (2008), pp. 2385–2402.
[37] P. A. Parrilo and S. Lall, *Semidefinite programming relaxations and algebraic optimization in control*, European Journal of Control, 9 (2003), pp. 307–321.
[38] M. Philippe, R. Essick, G. E. Dullerud, and R. M. Jungers, *Stability of discrete-time switching systems with constrained switching sequences*, Automatica, 72 (2016), pp. 242–250.
[39] V. Y. Protasov, *Spectral simplex method*, Mathematical Programming, 156 (2016), pp. 485–511.
[40] B. Reznick, *Extremal PSD forms with few terms*, Duke Math. J., 45 (1978), pp. 363–374, https://doi.org/10.1215/S0012-7094-78-04519-2, http://dx.doi.org/10.1215/S0012-7094-78-04519-2.
[41] G.-C. Rota and W. Strang, *A note on the joint spectral radius*, Proceedings of the Netherlands Academy, (1960). 22:379–381.
[42] N. Shor, *Class of global minimum bounds of polynomial functions*, Cybernetics and Systems Analysis, 23 (1987), pp. 731–734.
[43] H. Wolkowicz, R. Saigal, and L. Vandenberghe, *Handbook of semidefinite programming: theory, algorithms, and applications*, vol. 27, Springer Science & Business Media, 2012.

Appendix A. Stability certificates and duality.

Theorem 33. *Consider a finite set of matrices $\mathcal{A}$ constrained by an automaton $G(V,E)$. We have $\lim_{k\to\infty}\hat{\rho}_k(G,\mathcal{A},\|\cdot\|)\leq\overline{\gamma}^\star$.*

Proof. Consider a norm $\|\cdot\|$ of $\mathbb{R}^n$ and its corresponding induced matrix norm of $\mathbb{R}^{n\times n}$. For each $v\in V$, we know by compactness of the unit ball in $\mathbb{R}^n$, continuity and strict positivity of $f_v(x)$ that there exist $0<\alpha_v\leq\beta_v$ such that $\alpha_v\|x\|\leq f_v(x)\leq\beta_v\|x\|$ for all $x\in\mathbb{R}^n$. Let $\alpha=\min_{v\in V}\alpha_v$ and $\beta=\max_{v\in V}\beta_v$.

For a G-admissible k-uple $(\sigma_1,\sigma_2,\dots,\sigma_k)$, $\|A_{\sigma_k}\cdots A_{\sigma_1}\|=\sup_{x\neq 0}\frac{\|A_{\sigma_k}\cdots A_{\sigma_1}x\|}{\|x\|}$. Consider a path such that the ith edge has label σ_i for $i=1,\dots,k$ and denote the intermediary nodes of that path as $v_0,v_1,\dots,v_k$. For any $x\in\mathbb{R}^n$, we have

$$\|A_{\sigma_k}\cdots A_{\sigma_1}x\|\leq\alpha_{v_k}f_{v_k}(A_{\sigma_k}\cdots A_{\sigma_1}x)\leq\alpha_{v_k}\overline{\gamma}f_{v_{k-1}}(A_{\sigma_{k-1}}\cdots A_{\sigma_1}x)\leq\alpha_{v_k}\overline{\gamma}^k f_{v_0}(x)$$

and $\|x\|\geq\beta_{v_0}p_{v_0}(x)$ hence $\|A_{\sigma_k}\cdots A_{\sigma_1}\|\leq\frac{\beta_{v_0}}{\alpha_{v_k}}\overline{\gamma}^k\leq\frac{\beta}{\alpha}\overline{\gamma}^k$. Taking the kth root, the limit $k\to\infty$ and using Definition 2 we obtain the result. ∎

Lemma 34 (No duality gap). *For a fixed γ,*

Weak duality *If Program 3 (resp. Program 4) is feasible for $\overline{\gamma}=\gamma$ (resp. $\underline{\gamma}=\gamma$) then Program 4 (resp. Program 3) is infeasible for all $\underline{\gamma}<\gamma$ (resp. $\overline{\gamma}>\gamma$).*

Strong duality *If Program 3 (resp. dual) is infeasible for $\overline{\gamma}=\gamma$ (resp. $\underline{\gamma}=\gamma$) then Program 4 (resp. Program 3) is feasible for $\underline{\gamma}=\gamma$ (resp. $\overline{\gamma}=\gamma$).*

In other words, there exists a value γ^* such that for every $\gamma>\gamma^*$, there exists a feasible solution to Program 3 and for every $\gamma<\gamma^*$, there exists a feasible solution to Program 4. Moreover, either Program 3, Program 4 or both have a feasible solution with $\gamma=\gamma^*$.

Proof. Consider the hyperplane $C\triangleq\left\{(f_v:v\in V)\in\mathcal{F}^{|V|}\,\middle|\,\sum_{v\in V}\int_{\mathbb{S}^{n-1}}f_v(x)\,\mathrm{d}x=1\right\}$ and the map $\mathcal{D}_\gamma\colon\mathcal{F}^{|V|}\to\mathcal{F}^{|E|}:(f_v:v\in V)\mapsto(\gamma f_u(x)-f_v(A_\sigma x):(u,v,\sigma)\in E)$.

Given a fixed γ, Program 3 has no solution for $\overline{\gamma}=\gamma$ if and only if $\mathcal{D}_\gamma(\mathcal{F}_{++}^{|V|}\cap C)\cap\mathcal{F}_+^{|E|}=\emptyset$. Since $\mathcal{F}_{++}^{|V|}\cap C$ is compact, so is $\mathcal{D}_\gamma(\mathcal{F}_{++}^{|V|}\cap C))$. We know that a compact set and a closed set have no intersection if and only if there exist a strict separating hyperplane separating the two sets. That is, a measure $\mu \in \mathcal{M}$ such that $\langle \mu, f \rangle \geq 0$ for all $f \in \mathcal{F}_+^{|E|}$ and $\langle \mu, f \rangle < 0$ for all $f \in \mathcal{D}_\gamma(\mathcal{F}_{++}^{|V|} \cap C)$. The first condition is simply $\mu \in \mathcal{M}_+$. For the second condition, we remark that $\mathcal{D}_\gamma(\mathcal{F}_{++}^{|V|} \cap C) = \mathcal{D}_\gamma(\mathrm{int}(\mathcal{F}_+^{|V|}) \cap C) = \mathrm{ri}\, \mathcal{D}_\gamma(\mathcal{F}_+^{|V|} \cap C)$ where ri denotes the *relative interior* of a set. We have $\langle \mu, f \rangle < 0$ for all $f \in \mathrm{ri}\, \mathcal{D}_\gamma(\mathcal{F}_+^{|V|} \cap C)$ if and only if $\langle \mu, f \rangle \leq 0$ for all $f \in \mathcal{D}_\gamma(\mathcal{F}_+^{|V|} \cap C)$ and

$$\exists f \in \mathcal{D}_\gamma(\mathcal{F}_+^{|V|} \cap C) : \langle \mu, f \rangle \neq 0. \tag{23}$$

Therefore, if Program 3 has no solution for $\overline{\gamma} = \gamma$ then there exists a *nonzero* measure $\mu \in (\mathcal{M}_+)^{|E|}$ such that for all $f \in C$ and $(u, v, \sigma) \in E$,

$$\sum_{v \in V} \sum_{(v,u,\sigma) \in E} \bar{\gamma} \mathbb{E}_{vu\sigma}[f_v(x)] \leq \sum_{v \in V} \sum_{(u,v,\sigma) \in E} \mathbb{E}_{uv\sigma}[f_v(A_\sigma x))] \tag{24}$$

and (23) holds.

Note that if the inequality (24) is respected for some $f \in C$, it is also respected for λf for all $\lambda > 0$. So we can impose that the inequality should be respected for all $f \in \mathcal{F}_+^{|V|} \setminus \{0\}$.

The constraint (24) must be true for all $f \in \mathcal{F}_+^{|V|} \setminus \{0\}$ so in particular in the case where there is a node $v \in V$ such that $f_u(x) = 0$ for all $u \neq v$. Therefore we must have

$$\gamma \sum_{(v,u,\sigma) \in E} \mathbb{E}_{vu\sigma}[f_v(x)] \leq \sum_{(u,v,\sigma) \in E} \mathbb{E}_{uv\sigma}[f_v(A_\sigma x))], \quad \forall f_v \in \mathcal{F}_+$$

for all $v \in V$. This is (9) so the strong duality is proven.

To show the weak duality, we show that if there exists a dual solution μ for $\underline{\gamma} = \gamma$ then (9) and (23) are satisfied for all $\underline{\gamma} < \gamma$. We know that (9) is satisfied for γ so the constraint (9) is also satisfied for any $\underline{\gamma} < \gamma$. Using (24) and (10) with $f_v(x) = \|x\|$ for all $v \in V$, we have $\langle \mu, f \rangle < 0$ for all $\underline{\gamma} < \gamma$. ∎