Common Lyapunov Functions and Gradient Algorithms

Daniel Liberzon and Roberto Tempo

***Abstract*—This note is concerned with the problem of finding a quadratic common Lyapunov function for a large family of stable linear systems. We present gradient iteration algorithms which give deterministic convergence for finite system families and probabilistic convergence for infinite families.**

***Index Terms*—Common Lyapunov functions, gradient algorithms, randomized algorithms, switched linear systems.**

I. Introduction

The work reported in this note is motivated primarily by the problem of stability for switched systems. A switched system is a dynamical system described by a family of continuous-time subsystems and a rule that governs the switching between them. Such systems arise as models of processes regulated by switching control mechanisms and/or affected by abrupt changes in the dynamics. It is well known and easy to demonstrate that switching between stable subsystems may lead to instability. This fact makes stability analysis of switched systems an important and challenging problem, which has received considerable attention in the recent literature. We refer the reader to the recent book [11] for an overview of available results.

It is very desirable for individual subsystems to share a common Lyapunov function. If a switched system enjoys this property, then stability is preserved for arbitrary switching sequences.[1] When the subsystems being switched are obtained as feedback interconnections of a given process with different stabilizing controllers, this means that one does not need to worry about stability and can concentrate on other issues such as performance. Moreover, rather than simply knowing that a common Lyapunov function exists, it is important to compute it, as this yields precise information about the behavior of the switched system.

A particular case of interest is when the subsystems are linear time-invariant and a quadratic common Lyapunov function is sought. Although a number of conditions for the existence of such a Lyapunov function have been obtained, general results are lacking (particularly for system families with no special structure); see [11, Ch. 2]. On the other hand, the problem of finding a quadratic common Lyapunov function amounts to solving a system of linear matrix inequalities (LMIs), and efficient methods for solving such inequalities are available [2]. However, these algorithms offer limited theoretical insight and become ineffective as the number of subsystems being switched increases. Moreover, this approach is in general not useful for an infinite family of subsystems (with the notable exception of the convex hull of a finite family). Gridding techniques which are generally used for infinite families have the obvious drawbacks that the number of grid points grows exponentially with dimension and no guaranteed solution is obtained between grid points (see, e.g., [20]). Thus there is a need for developing simple and computationally tractable algorithms which can be used to find quadratic common Lyapunov functions for large (and unstructured) families of stable linear systems.

The approach proposed here is based on the idea of handling matrix inequality constraints sequentially rather than simultaneously, by means of an iterative gradient descent algorithm. For finite families of systems, this provides a useful alternative to solving systems of LMIs, which is of independent theoretical interest. (For comparison, we mention the iterative algorithm described in [14] for a special case of the above problem, namely, finding a common Lyapunov function for a finite family of commuting linear systems.) In the case of an infinite family of systems, for which general deterministic algorithms are not available, this method can be combined with randomization to yield probabilistic convergence. The main results of this note state that: i) in the deterministic setting, convergence to a quadratic common Lyapunov function (if one exists) in a finite number of steps is guaranteed for a finite family of linear systems; and ii) in the probabilistic setting, randomization allows us to obtain convergence with probability one for an infinite family. Probabilistic methods can also be applied, of course, to finite families; in this case, they can provide faster convergence than deterministic ones, at the expense of introducing a risk. (See [20] for more information on randomized algorithms.)

The use of gradient algorithms for solving *matrix* inequalities was proposed in [19] in the context of probabilistic control design for uncertain linear systems; see also [3], [5], [8], [16], and [21] for related recent developments. The main ideas go back to the early work on solving algebraic inequalities reported in [1], [13], [18], and [22]. Among the aforementioned references, [19] is especially relevant to this note, although it addresses a different problem and there are significant distinctions between the two papers. Specifically, we consider both deterministic and probabilistic settings, whereas [19] concentrates on the latter; the general framework described here allows a number of design choices, some of which were used in previous work while others were not; in addition, since we are concerned with systems of Lyapunov inequalities, while [19] deals with systems of Riccati inequalities, important technical differences arise as we explain below.

The rest of the note is structured as follows. Section II presents the problem formulation and preliminaries for the case of a finite family, Section III introduces the gradient iteration algorithms, and Section IV states and proves the deterministic convergence result. Section V presents the corresponding developments for infinite families, establishing a probabilistic counterpart of the previous result. In Section VI, we discuss several specific design choices. Section VII contains concluding remarks regarding computer simulations and questions for future work.

II. Problem Formulation and Notation

Suppose that we are given a family of real Hurwitz $n \times n$ matrices $A_1, \ldots, A_N$, where n and N are positive integers. We write $P \geq 0$ (or $P > 0$) to indicate that a matrix P is symmetric nonnegative definite (respectively, positive definite), and $P \leq 0$ (or $P < 0$) to indicate that P is symmetric nonpositive definite (respectively, negative definite). Suppose that there exists a matrix $P > 0$ which satisfies

$$PA_i + A_i^T P < 0, \qquad i = 1, \ldots, N.$$

Manuscript received September 10, 2002; revised April 14, 2003 and January 20, 2004. Recommended by Associate Editor A. Bemporad. The work of D. Liberzon was supported by the National Science Foundation under Grant ECS-0134115 CAR, ECS-0114725 and by DARPA/AFOSR under Grant MURI F49620-02-1-0325.

D. Liberzon is with the Coordinated Science Laboratory, the University of Illinois at Urbana-Champaign, Urbana, IL 61801 USA (e-mail: liberzon@uiuc.edu).

R. Tempo is with the IEIIT-CNR, Politecnico di Torino, 10129 Torino, Italy (e-mail: tempo@polito.it).

Digital Object Identifier 10.1109/TAC.2004.829632

[1]In the case when the given family of subsystems is infinite, this statement readily extends to a general time-varying system generated by this family.

This means that the quadratic function $V(x) := x^T Px$ is a common Lyapunov function for the family of asymptotically stable linear systems

$$\dot{x} = A_i x, \qquad i = 1, \ldots, N. \tag{1}$$

Fix an arbitrary matrix $Q > 0$. Multiplying P by a sufficiently large positive number, we see that the system of inequalities

$$PA_i + A_i^T P + Q \leq 0, \qquad i = 1, \ldots, N \tag{2}$$

has a solution $P > 0$. Moreover, if a symmetric matrix P satisfies (at least one of) the inequalities (2), then it is well known that we automatically have $P > 0$; see, e.g., [4, p. 132]. Thus, the problem of finding a quadratic common Lyapunov function for the family (1) is equivalent to that of finding a symmetric matrix P which satisfies (2). In what follows, we will be concerned with the latter problem. We denote the set of symmetric solutions of the inequalities (2) by $\mathcal{L}$.

The space of real symmetric $n \times n$ matrices is a Hilbert space with the inner product $\langle R, S \rangle := \mathrm{tr} RS$ and the Frobenius norm $\|R\| := (\sum_{i,j=1}^n R_{ij}^2)^{1/2}$. It is important to note that when the set $\mathcal{L}$ is nonempty, it must have a nonempty interior. Indeed, if $P \in \mathcal{L}$, then a standard perturbation argument such as the one given in [9, Ex. 5.1] can be used to show that $\mathcal{L}$ contains a neighborhood of γP for every $\gamma > 1$. In fact, we have $\gamma P + \Delta P \in \mathcal{L}$ for every symmetric matrix ΔP satisfying the bound

$$\lambda_{\max}(\Delta P) \leq \frac{(\gamma - 1)\lambda_{\min}(Q)}{2 \max\limits_{i=1,\ldots,N} \sigma_{\max}(A_i)}$$

where $\lambda_{\max}$ and $\lambda_{\min}$ denote the largest and the smallest eigenvalue, respectively, and $\sigma_{\max}$ denotes the largest singular value. It is well known that $\lambda_{\max}(\Delta P) \leq \|\Delta P\|$; see, e.g., [6, pp. 296–297]. Since γ can be arbitrarily large, we see that for every $r > 0$ there exists a ball of radius r which is contained in $\mathcal{L}$. This is in contrast with the case of Riccati inequalities treated in [19], where a sufficiently small ball inside the solution set $\mathcal{L}$ is assumed to exist but its radius r is not explicitly known.

In the sequel, we need the notion of projection of a symmetric matrix R onto the convex cone of nonnegative–definite matrices. This projection is defined as

$$R^+ := \arg\min_{S \geq 0} \|R - S\|.$$

The matrix R^+ can be computed explicitly as follows (see [18]): If $R = U\Lambda U^T$, where U is orthogonal and Λ is diagonal with entries $\lambda_1, \ldots, \lambda_n$, then $R^+ = U\Lambda^+ U^T$, where Λ^+ is diagonal with entries $\max\{0, \lambda_1\}, \ldots, \max\{0, \lambda_n\}$. We denote the matrix $R - R^+$ by R^-.

III. Gradient Algorithms

Let f be a convex functional on the space of symmetric matrices, which assigns to a matrix R a real number $f(R)$, with the property that $f(R) \leq 0$ if and only if $R \leq 0$. We suppose that this functional is differentiable (this condition can be relaxed somewhat, as discussed later). Specific examples of such functionals will be given in Section VI; here we keep the discussion general. We denote the gradient of f by $\partial_R f$, and similar notation will be used for other gradients appearing later. Recall that by definition

$$f(R + \Delta R) \approx f(R) + \langle \partial_R f, \Delta R \rangle$$

where ΔR denotes a small (symmetric) perturbation in R and $\approx$ stands for equality up to first-order terms in ΔR.

Given a symmetric matrix P and another matrix A, we let

$$v(P, A) := f(PA + A^T P + Q)$$

where f is the functional introduced previously and $Q > 0$ is the matrix from (2). Since f is convex, v is convex in P. Well-known results imply that for each integer i between 1 and N, solutions of the gradient system $\dot{P} = -\partial_P v(P, A_i)$ converge to the set $\{P : v(P, A_i) \leq 0\} = \{P : PA_i + A_i^T P + Q \leq 0\}$. The same is true for the associated discrete iterations $P_{k+1} = P_k - \mu_k \partial_P v(P_k, A_i)$, $k = 0, 1, \ldots$ if the step-sizes μ_k are chosen appropriately. Moreover, by switching between the above iterations for different values of i, we can make P_k converge to the intersection of the corresponding sets, which is precisely the set $\mathcal{L}$ of solutions of (2). This happens because the distance from P_k to $\mathcal{L}$ with respect to the Frobenius norm is a decreasing function of k.

We now make the aforementioned discussion precise by describing how the gradient iterations are to be carried out to ensure convergence to $\mathcal{L}$ in a finite number of steps. We need to pick a "scheduling function" h from nonnegative integers to the set $\{1, \ldots, N\}$ which has the following *revisitation property*: For every integer i between 1 and N and for every integer $l \geq 0$ there exists an integer $k \geq l$ such that $h(k) = i$. One obvious choice is

$$h(k) := (k \bmod N) + 1 = k - N \left\lfloor \frac{k}{N} \right\rfloor + 1. \tag{3}$$

An alternative approach based on randomization is also possible and will be pursued in Section V.

For each $k = 0, 1, \ldots$, let us define the step-size by the formula

$$\mu_k := \frac{\alpha v(P_k, A_{h(k)}) + r\|\partial_P v(P_k, A_{h(k)})\|}{\|\partial_P v(P_k, A_{h(k)})\|^2} \tag{4}$$

where $0 \leq \alpha \leq 2$ and $r > 0$ are arbitrary. Consider the iterations

$$P_{k+1} = \begin{cases} P_k - \mu_k \partial_P v(P_k, A_{h(k)}), & \text{if } v(P_k, A_{h(k)}) > 0 \\ P_k, & \text{otherwise.} \end{cases} \tag{5}$$

We take the initial condition P_0 to be symmetric. [For example, a solution of one of the inequalities (2) provides a convenient choice for P_0.] Then, P_k is symmetric for each k, since $\partial_P v$ is symmetric in light of the following lemma and the fact that $\partial_R f$ is symmetric.

Lemma 1: The gradient of v is given by

$$\partial_P v(P, A) = A\partial_R f(PA + A^T P + Q) + \partial_R f(PA + A^T P + Q)A^T. \tag{6}$$

Proof: Denoting by ΔP a small perturbation in P and by $\approx$ equality up to first-order terms in ΔP, we write

$$\begin{aligned} &v(P + \Delta P, A) \\ &= f(PA + A^T P + Q + \Delta P A + A^T \Delta P) \\ &\approx f(PA + A^T P + Q) + \langle \partial_R f(PA + A^T P + Q), \Delta P A + A^T \Delta P \rangle \\ &= v(P, A) \\ &\quad + \mathrm{tr}\left(\left(A\partial_R f(PA + A^T P + Q) + \partial_R f(PA + A^T P + Q)A^T \right) \Delta P \right) \end{aligned}$$

from which (6) follows. □

On the other hand, P_{k+1} is not guaranteed to be positive definite or at least nonnegative definite, even if $P_k > 0$. To make sure that a nonnegative definite matrix is generated at every step, we could instead consider

$$P_{k+1} = \begin{cases} [P_k - \mu_k \partial_P v(P_k, A_{h(k)})]^+, & \text{if } v(P_k, A_{h(k)}) > 0 \\ P_k, & \text{otherwise} \end{cases} \tag{7}$$

using the projection operation defined in Section II. Since all matrices in the set $\mathcal{L}$ are positive definite, this modification also has the potential of improving convergence; see the proof of Theorem 1 in Section IV.

Remark 1: The algorithm (7) exactly parallels that proposed in [19] for finding nonnegative–definite solutions of systems of Riccati inequalities. In contrast with that setting, the Lyapunov inequalities (2) have the property that if a symmetric matrix P_k satisfies at least one of them, then necessarily $P_k \geq 0$. This means that the projection is not really needed, and the convergence result presented in the next section implies that the algorithm (5) generates only nonnegative–definite matrices after sufficiently many steps.

IV. Deterministic Convergence for Finite Families

We now demonstrate that the gradient algorithms presented in the previous section provide convergence to the desired set $\mathcal{L}$ in a finite number of steps, unless $\mathcal{L}$ is empty. When for a given k we have $v(P_k, A_{h(k)}) > 0$ in (5) or (7), we say that a *correction step* is executed.

Theorem 1: When the algorithm (5) or the algorithm (7) is applied with the step-size given by (4) and the set $\mathcal{L}$ is nonempty, there exists an integer k^* such that $P_{k^*} \in \mathcal{L}$.

Proof: The proof is carried out along the lines of [19]. Consider a k for which $v(P_k, A_{h(k)}) > 0$ and so a correction step is executed. Suppose that the set $\mathcal{L}$ is nonempty. As shown in Section II, it then contains a ball of radius r, centered at some matrix P^*. We will prove that

$$\|P_{k+1} - P^*\|^2 \leq \|P_k - P^*\|^2 - r^2. \tag{8}$$

Since the revisitation property of h guarantees that a correction step occurs at least once in every N steps until $P_k \in \mathcal{L}$, we can conclude that no more than $N\lfloor \|P_0 - P^*\|^2/r^2 \rfloor$ steps are needed, and the proof will be complete.

Consider (7). We have

$$\begin{aligned}\|P_{k+1} - P^*\|^2 &= \|[P_k - \mu_k \partial_P v(P_k, A_{h(k)})]^+ - P^*\|^2 \\ &\leq \|P_k - \mu_k \partial_P v(P_k, A_{h(k)}) - P^*\|^2\end{aligned}$$

where the last inequality follows from the definition of the projection and the supporting hyperplane theorem. Thus, we see that it is enough to show (8) for (5).

To this end, define

$$\bar{P} := P^* + \frac{r}{\|\partial_P v(P_k, A_{h(k)})\|}\partial_P v(P_k, A_{h(k)}) \in \mathcal{L}.$$

Then, we use (5) to write

$$\begin{aligned}\|P_{k+1} - P^*\|^2 &= \|P_k - P^* - \mu_k \partial_P v(P_k, A_{h(k)})\|^2 \\ &= \|P_k - P^*\|^2 + \mu_k^2 \|\partial_P v(P_k, A_{h(k)})\|^2 \\ &\quad - 2\mu_k \langle \partial_P v(P_k, A_{h(k)}), P_k - \bar{P}\rangle \\ &\quad - 2\mu_k \langle \partial_P v(P_k, A_{h(k)}), \bar{P} - P^*\rangle.\end{aligned}$$

We now consider the last two terms. Due to convexity of v in P, we have

$$\langle \partial_P v(P_k, A_{h(k)}), P_k - \bar{P}\rangle \geq v(P_k, A_{h(k)})$$

while the definition of $\bar{P}$ gives

$$\langle \partial_P v(P_k, A_{h(k)}), \bar{P} - P^*\rangle = r\|\partial_P v(P_k, A_{h(k)})\|.$$

Therefore

$$\begin{aligned}\|P_{k+1} - P^*\|^2 \leq \|P_k - P^*\|^2 + \mu_k^2 \|\partial_P v(P_k, A_{h(k)})\|^2 \\ -2\mu_k \left(v(P_k, A_{h(k)}) + r\|\partial_P v(P_k, A_{h(k)})\| \right).\end{aligned}$$

Substituting the value of μ_k defined in (4), we obtain

$$\begin{aligned}\|P_{k+1} - P^*\|^2 \leq \|P_k - P^*\|^2 - \frac{\alpha(2-\alpha)\left(v(P_k, A_{h(k)})\right)^2}{\|\partial_P v(P_k, A_{h(k)})\|^2} \\ - \frac{2r\, v(P_k, A_{h(k)})}{\|\partial_P v(P_k, A_{h(k)})\|} - r^2 \leq \|P_k - P^*\|^2 - r^2\end{aligned}$$

and so (8) holds as claimed. □

Since the matrix P^* used in the previous proof is not known, the number k^* may be difficult to estimate in practice. One can, of course, let the algorithm run for some number of steps k and then check whether or not the matrix P_k satisfies the inequalities (2); performing such a check is an easy task compared with that of solving the inequalities directly. We also have a lot of freedom in the choice of the step-size; more information on how to choose the step-size efficiently in gradient algorithms of this kind can be found, e.g., in [10], [18], and [22].

V. Probabilistic Convergence for Infinite Families

Let us now consider the more general situation where we are given a compact set of real Hurwitz matrices $\mathcal{A} := \{A_p : p \in \mathcal{P}\}$, parameterized by some index set $\mathcal{P}$ which is in general infinite. As we explained in Section I, this problem cannot be solved with available deterministic algorithms without resorting to approximation techniques such as gridding. Because of compactness, it is still true that the problem of finding a matrix $P > 0$ which satisfies the inequalities

$$PA_p + A_p^T P < 0 \qquad \forall p \in \mathcal{P}$$

is equivalent to the problem of finding a symmetric matrix P which satisfies the inequalities

$$PA_p + A_p^T P + Q \leq 0 \qquad \forall p \in \mathcal{P}$$

where $Q > 0$ is arbitrary. As before, we denote the set of such matrices P by $\mathcal{L}$. We can show as in Section II that when $\mathcal{L}$ is nonempty, it contains balls of arbitrarily large radii.

Since the set of matrices is infinite, choosing a matrix at each step using a function h with the revisitation property as in Section III is no longer possible. Instead, we use randomization. Namely, for each k we randomly pick a matrix in $\mathcal{A}$ according to some probability distribution on $\mathcal{A}$ with the following property: Every subset of $\mathcal{A}$ which is open relative to $\mathcal{A}$ has a nonzero probability measure. We let $h(k)$ be the index of the matrix chosen in this way (this index may not be unique if the map $p \mapsto A_p$ is not injective). Note that $\mathcal{A}$ may be composed of several disjoint subsets such as intervals or isolated points, and we must assign a positive probability measure to each of them. If $\mathcal{A}$ consists of a finite number of isolated points, then randomization provides a valid alternative to the approach described in Section III. As is well known, for infinite families described by convex polyhedra the problem reduces to the corresponding problem for finite families generated by the vertices; see, e.g., [12] and the references therein.

With $h(k)$ generated in this way for each k, *randomized versions* of the algorithms from Section III can now be defined by the same formulas (5) and (7). Remark 1 also applies here. If $P_k \notin \mathcal{L}$, then

$v(P_k, \bar{A}) > 0$ for some $\bar{A} \in \mathcal{A}$, and by continuity the same is true for all A in a sufficiently small neighborhood of $\bar{A}$ in $\mathcal{A}$. Since the probability measure of this neighborhood is positive, a correction step will be executed with probability one after a finite number of steps. Proceeding exactly as in the proof of Theorem 1, we arrive at the following probabilistic counterpart of that theorem.

Theorem 2: When the randomized version of the algorithm (5) or the algorithm (7) is applied with the step-size given by (4) and the set $\mathcal{L}$ is nonempty, with probability one there exists an integer k^* such that $P_{k^*} \in \mathcal{L}$.

This result parallels [19, Th. 1], although the context here is different. In [19], it is also shown how one can compute a lower bound on the probability of finding a solution in a given number of steps, provided that additional a priori information is available. Some related recent references are briefly discussed in Section VII.

VI. Possible Choices for f

We suggest two admissible choices of the functional f. One is given by

$$f(R) := \|R^+\|^2. \tag{9}$$

This functional has all the properties required in Section III, as shown in [18]. For completeness, we sketch the argument.

Lemma 2: The functional (9) is convex and differentiable, with gradient given by

$$\partial_R f(R) = 2R^+. \tag{10}$$

Proof: The calculations that follow rely on some standard properties of the projection and the cone of symmetric nonnegative–definite matrices; see [18] for details. We write

$$\begin{aligned} f(R+\Delta R) &= \|R+\Delta R-(R+\Delta R)^-\|^2 = \inf_{S\le 0}\|R+\Delta R-S\|^2 \\ &\le \|R+\Delta R-R^-\|^2 = \|R^+ + \Delta R\|^2 \\ &= \|R^+\|^2+\langle 2R^+, \Delta R\rangle+\|\Delta R\|^2 \approx f(R)+\langle 2R^+, \Delta R\rangle. \end{aligned}$$

On the other hand

$$\begin{aligned} &f(R+\Delta R) \\ &= \|R+\Delta R-(R+\Delta R)^-\|^2 = \|R^+ + R^- + \Delta R-(R+\Delta R)^-\|^2 \\ &= \|R^+\|^2+\langle 2R^+, \Delta R\rangle + 2\langle R^+, R^-\rangle + 2\langle R^+, -(R+\Delta R)^-\rangle \\ &\quad + \|R^- + \Delta R - (R+\Delta R)^-\|^2 \ge f(R) + \langle 2R^+, \Delta R\rangle. \end{aligned}$$

This proves that f is differentiable with gradient given by (10), and the inequality $f(R+\Delta R) \ge f(R) + \langle \partial_R f(R), \Delta R\rangle$ implies that f is convex. □

Combining (6) and (10), we see that with this choice of f the gradient of v is given by

$$\partial_P v(P, A) = 2A(PA+A^TP+Q)^+ + 2(PA+A^TP+Q)^+A^T.$$

This is the quantity that needs to be calculated at every step when implementing the gradient algorithms described earlier. As explained in Section II, this is a routine calculation based on eigenvalue decomposition. An analogous construction was used in [19], where the authors work with the functional $f(R) := \|R^+\|$ for which the gradient expressions are similar (but convergence properties may be different).

Another option is to let $f(R)$ be the largest eigenvalue of R

$$f(R) := \lambda_{\max}(R). \tag{11}$$

The following result is standard; see, e.g, [6, p. 372].

Lemma 3: The functional (11) is convex and, when $\lambda_{\max}(R)$ is a simple eigenvalue, differentiable with gradient given by

$$\partial_R f(R) = xx^T$$

where x is a unit eigenvector of R with eigenvalue $\lambda_{\max}(R)$.

With this choice of f, the gradient of v is given by

$$\partial_P v(P, A) = Axx^T + xx^TA^T \tag{12}$$

and the projection is no longer required for implementation of the algorithm (5). A disadvantage of this f is that it is not differentiable when $\lambda_{\max}(R)$ is not a simple eigenvalue. However, since a generic matrix has distinct eigenvalues, this problem can always be fixed by slightly perturbing the matrix P_k if necessary. We could also work with subgradients rather than gradients, as done in [19].

More generally, we can take $f(R)$ to be the sum of m largest eigenvalues of a symmetric matrix R, where $1 \le m \le n$. This gives a convex functional, as shown in [17]. When the eigenvalues are simple, the gradient of v is the sum of the quantities (12) for the corresponding eigenvectors.

VII. Concluding Remarks

We presented gradient iteration algorithms for finding a quadratic common Lyapunov function for a family of asymptotically stable linear systems. We derived a deterministic convergence result for finite families, and then used randomization to obtain a probabilistic counterpart for infinite families. The algorithms were described on a general level with some freedom left in the design choices, several possibilities for which were also discussed.

We used MATLAB to implement the algorithms (5) and (7) with f given by (9) or (11) for interval families[2] of 4×4 and 5×5 upper-triangular Hurwitz matrices, for which quadratic common Lyapunov functions are known to exist (see, e.g., [11, Prop. 2.9]). We first applied the method of Section III, noting that it suffices to work with the vertices and using deterministic iterations based on (3). Programming each algorithm was a straightforward task, and their performance was comparable. In the 4×4 case, we observed consistent convergence for randomly generated initial matrices P_0 in less than 10 000 iterations (which took just a few seconds). In the 5×5 case, we observed consistent convergence in less than 10 000 000 iterations (this took several hours); a matrix satisfying more than 90% of the inequalities, on the other hand, was usually found after about 100 000 iterations (i.e., in a matter of minutes). In both cases, the number of actual correction steps was two orders of magnitude lower than the total number of iterations. We then repeated the experiments with the vertices generated randomly at each step, and observed that the number of correction steps needed for complete convergence was typically reached after a noticeably smaller number of iterations.

Tradeoffs among different choices of the specific gradient algorithm and the step-size remain to be understood. It is important to note that no analytical characterizations of the existence of a common Lyapunov function are available for generic system matrices of dimension higher than 2×2, and that interval families of 5×5 triangular matrices give rise to systems of 2^{15} LMIs, solving which simultaneously is a difficult

[2]In an *interval family* of matrices, the value of each element varies over an interval.

task (particularly without using relaxation techniques). For example, the program quadstab from MATLAB's LMI Toolbox was not able to produce an answer for the 5×5 case (its performance in the 4×4 case was comparable to that of the gradient algorithms). We attribute this difference in performance to the fact that our method handles the inequalities one by one rather than simultaneously and so, while a lot of iterations may be required, each iteration only needs a very modest amount of computation.

A comparative study of computational aspects of the algorithms presented in this note, existing methods for solving LMIs, and other techniques such as the gradient algorithms for solving systems of Lyapunov inequalities described in [7] is an interesting topic which is beyond the scope of this note. Of course, such a comparison would only apply to situations where the given family of matrices is finite or a convex hull of a finite family (as in the above example). As for randomized algorithms for infinite families of the kind proposed here, the issue of their computational complexity and numerical efficiency has received a lot of attention in the recent literature. See, e.g, [16] for some preliminary results on computational complexity of gradient-based algorithms for finding probabilistic solutions, as well as numerical experiments regarding the number of required iterations. Further related work is reported in [15]; in that paper, explicit upper bounds on the number of steps of the ellipsoidal method are given, but the same bound also holds for gradient-based methods.

The method described here can be used to find a quadratic common Lyapunov function when one exists. In other words, strictly feasible problems are studied. This note does not offer insight into the existence question, which needs further investigation. By combining the aforementioned algorithms with a suitable recursive partitioning procedure, one can compute common Lyapunov functions for subsets of the given family of systems when the overall common Lyapunov function may not exist or is not found in a prescribed number of steps. Stability of the switched system is then guaranteed under appropriate constraints on the switching rate between different subsets; cf. [11, Ch. 3]. Problems for future work also include choosing an optimal schedule for the iterations with respect to some performance criterion (for example, the speed of convergence), incorporating additional constraints on the Lyapunov function to be found (such as the average or worst-case decay rate), and treating optimization rather than feasibility problems.

ACKNOWLEDGMENT

The authors would like to thank B. R. Barmish, G. Dullerud, Y. Fujisaki, C. Hadjicostis, and B. Polyak for helpful discussions. This research was performed in part while R. Tempo was visiting the Coordinated Science Laboratory at the University of Illinois as a CSL Visiting Research Professor; the support of this program is gratefully acknowledged.

REFERENCES

[1] S. Agmon, "The relaxation method for linear inequalities," *Canad. J. Math.*, vol. 6, pp. 382–392, 1954.
[2] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*. Philadelphia, PA: SIAM, 1994, vol. 15, SIAM Studies in Applied Mathematics.
[3] G. Calafiore and B. T. Polyak, "Stochastic algorithms for exact and approximate feasibility of robust LMIs," *IEEE Trans. Automat. Contr.*, vol. 46, pp. 1755–1759, Sept. 2001.
[4] C.-T. Chen, *Linear System Theory and Design*, 3rd ed. New York: Oxford Univ. Press, 1999.
[5] Y. Fujisaki, F. Dabbene, and R. Tempo, "Probabilistic design of LPV control systems," *Automatica*, vol. 39, pp. 1323–1337, 2003.
[6] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 1985.
[7] V. A. Kamenetskiy and Y. S. Pyatnitskiy, "An iterative method of Lyapunov function construction for differential inclusions," *Syst. Control Lett.*, vol. 8, pp. 445–451, 1987.
[8] S. Kanev, B. De Schutter, and M. Verhaegen, "An ellipsoid algorithm for probabilistic robust controller design," *Syst. Control Lett.*, vol. 49, pp. 365–375, 2003.
[9] H. K. Khalil, *Nonlinear Systems*, 2nd ed. Upper Saddle River, NJ: Prentice-Hall, 1996.
[10] H. J. Kushner and G. G. Yin, *Stochastic Approximation Algorithms and Applications*. New York: Springer-Verlag, 1997.
[11] D. Liberzon, *Switching in Systems and Control*. Boston, MA: Birkhäuser, 2003.
[12] A. P. Molchanov and Y. S. Pyatnitskiy, "Criteria of absolute stability of differential and difference inclusions encountered in control theory," *Syst. Control Lett.*, vol. 13, pp. 59–64, 1989.
[13] T. S. Motzkin and I. J. Schoenberg, "The relaxation method for linear inequalities," *Canad. J. Math.*, vol. 6, pp. 393–404, 1954.
[14] K. S. Narendra and J. Balakrishnan, "A common Lyapunov function for stable LTI systems with commuting A-matrices," *IEEE Trans. Automat. Contr.*, vol. 39, pp. 2469–2471, Dec. 1994.
[15] Y. Oishi, "Probabilistic design of a robust state-feedback controller based on parameter-dependent Lyapunov functions," in *Proc. 42nd IEEE Conf. Decision Control*, 2003, pp. 1920–1925.
[16] Y. Oishi and H. Kimura, "Computational complexity of randomized algorithms for solving parameter-dependent linear matrix inequalities," *Automatica*, vol. 39, pp. 2149–2156, 2003.
[17] M. L. Overton and R. S. Womersley, "Optimality conditions and duality theory for minimizing sums of the largest eigenvalues of symmetric matrices," *Math. Program.*, vol. 62, pp. 321–357, 1993.
[18] B. T. Polyak, "Gradient methods for solving equations and inequalities," *USSR Comp. Math. Math. Phys.*, vol. 4, pp. 17–32, 1964.
[19] B. T. Polyak and R. Tempo, "Probabilistic robust design with linear quadratic regulators," *Syst. Control Lett.*, vol. 43, pp. 343–353, 2001.
[20] R. Tempo, G. Calafiore, and F. Dabbene, *Randomized Algorithms for Analysis and Control of Uncertain Systems*. London, U.K.: Springer-Verlag, 2004.
[21] V. A. Ugrinovskii, "Randomized algorithms for robust stability and control of stochastic hybrid systems with uncertain switching policies," in *Proc. 40th IEEE Conf. Decision Control*, 2001, pp. 5026–5031.
[22] V. A. Yakubovich, "Recurrent finitely convergent algorithms for solving systems of inequalities," *Soviet Math. Dokl.*, vol. 7, pp. 300–304, 1966.