

UNIVERSITY OF ICELAND

Ph.D. Dissertation
in Mathematics

Stability Analysis of Switched Linear Systems via Numeric Computations of Continuous Piecewise Affine and Continuous Piecewise Quadratic Lyapunov Functions

Stefania Andersen

November 2026

FACULTY OF PHYSICAL SCIENCES

Stability Analysis of Switched Linear Systems via Numeric Computations of Continuous Piecewise Affine and Continuous Piecewise Quadratic Lyapunov Functions

Stefania Andersen

Dissertation submitted in partial fulfillment of a
Philosophiae Doctor degree in Mathematics

Ph.D. Committee
Elias August
Sigurdur Hafstein
Anna Soffía Hauksdóttir
Peter Giesl

Opponents
Shenyu Liu
Florian Rupp

Faculty of Physical Sciences
School of Engineering and Natural Sciences
University of Iceland
Reykjavik, November 2026

Stability Analysis of Switched Linear Systems via Numeric Computations of Continuous Piecewise Affine and Continuous Piecewise Quadratic Lyapunov Functions
(CPA and CPQ Lyapunov Functions)

Dissertation submitted in partial fulfillment of a *Ph.D.* degree in Mathematics

Copyright © 2026 Stefania Andersen
All rights reserved

Faculty of Physical Sciences
School of Engineering and Natural Sciences
University of Iceland
Dunhagi 5
107 Reykjavik
Iceland

Telephone: 525-4000

Bibliographic information:
Stefania Andersen, 2026, *Stability Analysis of Switched Linear Systems via Numeric Computations of Continuous Piecewise Affine and Continuous Piecewise Quadratic Lyapunov Functions*, Ph.D.Dissertation, Faculty of Physical Sciences, University of Iceland, 85 pp.

Author ORCID: 0000-0001-6747-775X
ISBN: 978-9935-585-12-7

Printing: Háskólaprent
Reykjavik, Iceland, November 2026

Abstract

In this dissertation we present numerical methods for computing continuous piecewise affine (CPA) and continuous piecewise quadratic (CPQ) Lyapunov functions via linear programming (LP) for switched linear systems. We discuss the application of these methods to both arbitrary switching signals and signals subject to average dwell-time (ADT) constraints. Further, we discuss the challenges of computing the minimum average dwell-time (MADT) and present a search algorithm for its computation. Finally, we discuss our approach to computing CPQ functions via LP, rather than semidefinite programming (SDP).

Ágrip

Farið er yfir tölulegar aðferðir fyrir útreikninga Lyapunov falla, sem eru samfelld og línuleg- eða ferningsmargliður á köflum, með línulegri bestun fyrir línuleg skiptikerfi. Fjallað verður um notkun þessara aðferða við stöðugleikagreiningar þegar stýrimerkið er annarsvegar ótiltekið og þegar settar eru meðaldvalartímaskorður hinsvegar. Einnig verður fjallað um þær áskoranir sem eru fólgnar í útreikningum lámarksmeðaldvalartíma fyrir stöðugleika og kynnt leitarreiknirit til að reikna lámarksmeðaldvalartíma. Loks verður fjallað um hvernig við nálgumst tölulega útreikninga á Lyapunov föllum sem eru samfelld og ferningsmargliður á köflum með línulegri bestun í stað hálfákvarðari bestun.

やった!

Table of Contents

List of Figures **xi**
List of Publications **xiii**
Abbreviations **xv**
1 Introduction **1**
2 Stability of Switched Linear Systems **3**
2.1 Arbitrary switching 4
2.2 Minimum average dwell-time 5
3 Numerical methods **9**
3.1 CPA functions 12
3.2 CPQ functions 13
4 Conclusions **19**
References **20**
Paper I **24**
Paper II **33**
Paper III **58**
Paper IV **65**
Paper V **79**

List of Figures

2.1 Trajectories of (a) two unstable subsystems starting at $\mathbf{x} = [0,1]^T$ and (b) a stable switched system starting at $\mathbf{x} = [0,10]^T$ [1, Figure 2.1]. 3

2.2 Trajectories of (a) two stable subsystems starting at $\mathbf{x} = [0,1]^T$ and (b) an unstable switched system starting at $\mathbf{x} = [-1,0]^T$ [1, Figure 2.2]. 4

2.3 An example of the area D where (6) has a feasible solution given α and μ. Note that the area extends indefinitely as $\mu \to \infty$. Adapted from [2]. 7

3.1 Level sets of $V : \mathbb{R}^2 \to \mathbb{R}$ for the three different function classes under consideration: (a) quadratic, (b) continuous piecewise affine (CPA), and (c) continuous piecewise quadratic (CPQ) functions. 10

3.2 Example of the triangular fan used in (a) two dimensions and (b) three dimensions. The shaded area in (b) corresponds to one face of the cube constructed as an intermediate step in the triangulation definition. Note that the origin is a vertex in both cases. Adapted from [1]. 11

3.3 Illustration of the global matrix Φ (left) and a triangulation with simplices and vertices marked (right). Using the indexing function g_v, the highlighted submatrices Ψ^7 (green) and Ψ^0 (red) are extracted from Φ, containing the decision variables corresponding to the vertex indices of simplices $\mathfrak{S}_7$ and $\mathfrak{S}_0$, respectively. 12

List of Publications

This thesis includes of the following papers. A short description of the papers can be found before each paper.

Paper I: **S. Andersen**, S. Hafstein, J. Roman and S.J.A.M van den Eijnden, 2024, Efficient Implementation of Piecewise Quadratic Lyapunov Function Computations for Switched Linear Systems. *Proceedings of the 21st International Conference on Informatics in Control, Automation and Robotics (ICINCO)*, pp. 277–284.

Paper II: **S. Andersen**, E. August, S. Hafstein and J. Piccini, 2024, Numerical Computations of Lyapunov Functions for Switched Linear Systems. *Simulation and Modeling Methodologies, Technologies and Applications*, pp. 192–213.

Paper III: **S. Andersen**, S. Hafstein, 2025, Investigating Average Dwell-Time Constraints on Multiple Lyapunov Functions for Switched Linear Systems. *The 13th IFAC Symposium on Nonlinear Control Systems*, pp. 150–155.

Paper IV: J.J. Palacios Roman, S.F. Hafstein, P.A. Giesl, S.J.A.M. van den Eijnden, **S. Andersen**, W.P.M.H. Heemels, 2026, Constructing Piecewise Quadratic Lyapunov Functions with Linear Programming for Continuous-Time Switched and Conewise Linear Systems. *Automatica*, Vol. 186, Article no. 112838.

Paper V: **S. Andersen**, S.F. Hafstein, 2026, Minimum Average Dwell-Time Computations via Continuous Piecewise Quadratic Lyapunov Functions, *2026 30th International Conference on Methods and Models in Automation and Robotics (MMAR)*, pp. 351–356.

Abbreviations

CLF	common Lyapunov function
LMI	linear matrix inequality
LP	linear programming
SDP	semidefinite programming
CPA	continuous piecewise affine
CPQ	continuous piecewise quadratic
MADT	minimum average dwell-time
ADT	average dwell-time
p.d.	positive definite
n.d.	negative definite
p.s.d.	positive semidefinite
n.s.d.	negative semidefinite
GAS	globally asymptotically stable
GES	globally exponentially stable

1 Introduction

At the core of control theory are differential equations defining system behaviour. In general we can describe the behaviour with

$$\dot{\mathbf{x}} = f(\mathbf{x}) \tag{1}$$

where $\mathbf{x} \in \mathbb{R}^n$ are the state variables of the n-dimensional system and $f\colon \mathbb{R}^n \to \mathbb{R}^n$ defines the system dynamics. The solution curve or trajectory is denoted with $\phi(t,\xi)$, where $\xi \in \mathbb{R}^n$ is the initial condition, i.e. $\phi(0,\xi) = \xi$.

It is often beneficial for design, control, and stability analysis, to describe the dynamics with a set of subsystems. Such systems are called switched systems and can be described with

$$\dot{\mathbf{x}} = f_{\sigma(t)}(\mathbf{x}),$$

where $\sigma : [0,\infty) \to [\![1,N]\!]$ is called the switching signal, indicating which of the N subsystems is currently describing the dynamics. Here $[\![a,b]\!] := [a,b] \cap \mathbb{Z}$ is a discrete index set.

To facilitate the discussion of the topic of this dissertation let's first consider the following definitions:
Definition 1. *Consider the dynamic system* (1)*. A point* $\tilde{\mathbf{x}} \in \mathbb{R}^n$ *is called an* equilibrium point *of the system if and only if* $f(\tilde{\mathbf{x}}) = \mathbf{0}$.
Definition 2. *An equilibrium point* $\tilde{\mathbf{x}}$ *is said to be*

1. Lyapunov stable*, if, for every* $\varepsilon > 0$ *there exists a* $\delta > 0$ *such that if* $\|\xi - \tilde{\mathbf{x}}\| < \delta$ *then for every* $t \geq 0$ *we have* $\|\phi(t,\xi) - \tilde{\mathbf{x}}\| < \varepsilon$,
2. asymptotically stable*, if, in addition to Lyapunov stability, there exists a* $\delta > 0$ *such that if* $\|\xi - \tilde{\mathbf{x}}\| < \delta$ *then we have* $\lim_{t\to\infty} \|\phi(t,\xi) - \tilde{\mathbf{x}}\| = 0$,
3. exponentially stable*, if, in addition to asymptotic stability, there exists a* $\delta > 0$, $\alpha > 0$ *and* $\beta > 0$ *such that if* $\|\xi - \tilde{\mathbf{x}}\| < \delta$ *then for every* $t \geq 0$ *we have* $\|\phi(t,\xi) - \tilde{\mathbf{x}}\| \leq \alpha \|\xi - \tilde{\mathbf{x}}\| \exp(-\beta t)$.

Note Definition 2 defines a locally stable equilibrium point. If the conditions are satisfied for all $\xi \in \mathbb{R}^n$ the system (or equilibrium point) is said to be globally asymptotically stable (GAS) or globally exponentially stable (GES). We can, without loss of generality, assume $\tilde{\mathbf{x}} = \mathbf{0}$ by considering a change of coordinates. From this point on we will consider the origin as the equilibrium point of interest.

In 1892, Aleksandr Mikhailovich Lyapunov, described a new method now referred to as *Lyapunov's second method* [15] (eng. trans. [16]). The method can be described by the following definition and theorem.

Definition 3. *Let $V : \mathbb{R}^n \to \mathbb{R}$ be a radially unbounded function that has continuous first derivatives, and consider the system in* (1) *with an equilibrium point at the origin. If V satisfies*

$$\begin{aligned} V(\mathbf{x}) &> 0 && \forall \mathbf{x} \in \mathbb{R}^n \setminus \{\mathbf{0}\}, \\ V(\mathbf{0}) &= 0, \\ \langle \nabla V(\mathbf{x}), f(\mathbf{x}) \rangle &< 0 && \forall \mathbf{x} \in \mathbb{R}^n \setminus \{\mathbf{0}\}, \end{aligned}$$

then V is called a Lyapunov function.

Theorem 1. *Consider the dynamic system* (1) *with an equilibrium point at the origin. If there exists a Lyapunov function for the system then the origin is GAS*[1] .

Here $\langle \cdot, \cdot \rangle$ is the inner product between two vectors.

In this dissertation we will present numerical methods to compute Lyapunov functions for different function classes.

Finally note that a matrix $Q \in \mathbb{R}^{n \times n}$ is said to be symmetric positive definite (p.d.) on the set D, denoted with $Q \succ_D 0$, if $\mathbf{x}^T Q \mathbf{x} > 0$ for all $\mathbf{x} \in D \setminus \{\mathbf{0}\}$ and positive semidefinite (p.s.d.) on the set D, denoted with $Q \succeq_D 0$, if $\mathbf{x}^T Q \mathbf{x} \geq 0$ for all $\mathbf{x} \in D \setminus \{\mathbf{0}\}$. The matrix is similarly said to be negative definite (n.d.) and negative semidefinite (n.s.d.) on the set D when $\mathbf{x}^T Q \mathbf{x} < 0$ and $\mathbf{x}^T Q \mathbf{x} \leq 0$ for all $\mathbf{x} \in D \setminus \{\mathbf{0}\}$ denoted with $Q \prec_D 0$ and $Q \preceq_D 0$ respectively. When $D = \mathbb{R}^n$ we write $Q \succ 0$ and say the matrix is symmetric p.d. and equivalently for the other relations. When a matrix is symmetric p.d. on the set $\mathbb{R}^n_+$ the matrix is said to be co-positive[2].

[1] Note that when the system is linear GAS is equivalent to GES

[2] $\mathbb{R}_+ = [0, \infty)$

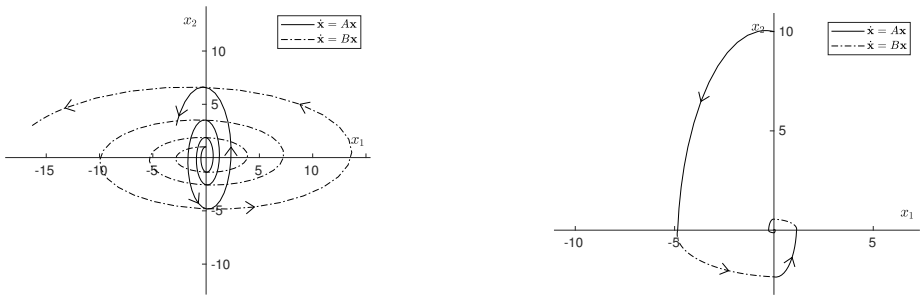
2 Stability of Switched Linear Systems

The stability analysis of switched systems can be tricky. One can not rely solely on the stability analysis of the subsystems since switching can stabilize unstable systems and vice versa, see Figures 2.1 and 2.2 respectively.

In this dissertation we will consider switched linear systems of the form

$$\dot{\mathbf{x}} = A_{\sigma(t)}\mathbf{x}, \tag{3}$$

where $A_i \in \mathbb{R}^{n\times n}$, $i \in [\![1,N]\!]$, and two different constraints on the switching signal; arbitrary switching and minimum average dwell-time (MADT) switching. We always assume σ is right-continuous and has a finite number of discontinuity-points over a finite time interval. When considering MADT we are additionally interested in limiting the average number of discontinuity-points over a fixed time interval.



(a) *(b)*

Figure 2.1. Trajectories of (a) two unstable subsystems starting at $\mathbf{x} = [0,1]^T$ *and (b) a stable switched system starting at* $\mathbf{x} = [0,10]^T$ *[1, Figure 2.1].*

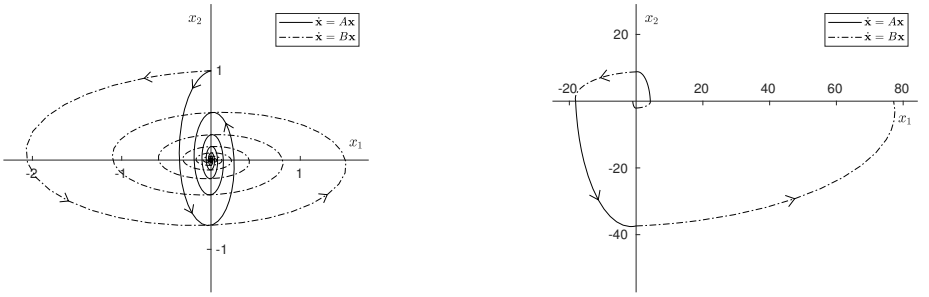


Figure 2.2. Trajectories of (a) two stable subsystems starting at $\mathbf{x} = [0,1]^T$ *and (b) an unstable switched system starting at* $\mathbf{x} = [-1,0]^T$ *[1, Figure 2.2].*

2.1 Arbitrary switching

As arbitrary switching implies the switching signal could be a constant, therefore the subsystems need to be GES in their own right if we require the switched system to be GES. In 1960, Rudolf Emil Kalman and John Edward Bertram reformalized the results presented in Section 20 of [15] that became the most common way to prove the stability of linear dynamic system, $\dot{\mathbf{x}} = A\mathbf{x}$ [13, Chapter 6]. Namely to find a solution to a special case of the Sylvester equation, now called the Lyapunov equation.

Theorem 2. *Consider the system* $\dot{\mathbf{x}} = A\mathbf{x}$ *and let* Q *be symmetric p.d. matrix, where* $A, Q \in \mathbb{R}^{n\times n}$*. The existence of a symmetric p.d. matrix* $P \in \mathbb{R}^{n\times n}$ *such that*

$$A^T P + PA = -Q$$

is equivalent to A being Hurwitz, namely the real part of all its eigenvalues being strictly negative and the system is GES at the origin. We can aditionally define a Lyapunov function for the system as $V(\mathbf{x}) = \mathbf{x}^T P\mathbf{x}$*.*

Consequently a necessary condition is that each A_i is Hurwitz, where $i \in [\![1,N]\!]$. As discussed this is of course only a necessary and not sufficient condition for GES under arbitrary switching. We additionally need to ensure that switching between subsystems does not destabilize the switched system. In this case we search for a common Lyapunov function (CLF). A CLF is one Lyapunov function that satisfies (2) for all

subsystems, i.e. a Lyapunov function $V : \mathbb{R}^n \to \mathbb{R}$ that satisfies

$$V(\mathbf{x}) > 0 \qquad \forall \mathbf{x} \in \mathbb{R}^n \setminus \{\mathbf{0}\}, \tag{4a}$$

$$V(\mathbf{0}) = 0, \tag{4b}$$

$$\langle \nabla V(\mathbf{x}), A_i \mathbf{x} \rangle < 0 \qquad \forall \mathbf{x} \in \mathbb{R}^n \setminus \{\mathbf{0}\},\ i \in [\![1, N]\!]. \tag{4c}$$

The natural progression from these two conditions is to solve the linear matrix inequality (LMI)[3]

$$P \succ 0 \tag{5a}$$

$$A_i^T P + P A_i \prec 0 \tag{5b}$$

using semidefinite programming (SDP). For notes on SDP and the limitations of this method see Paper II [5].

2.2 Minimum average dwell-time

The concept of average dwell-time (ADT), rather than constant dwell-time, is relatively new. It was first defined and considered in a paper by João P. Hespanha and A. Stephen Morse in 1999 [11], where they defined the ADT of a switching signal in the following way:

Definition 4. *Let $N_\sigma(t,s)$ denote the number of discontinuity-points of switching signal σ on the open time interval (s,t). We say that σ has an* average dwell-time (ADT) *$\tau > 0$, if there exists an $N_0 > 0$ such that for all $t \geq s \geq 0$ we have*

$$N_\sigma(t,s) \leq N_0 + \frac{t-s}{\tau}$$

In layman's terms, the average time interval between consecutive switchings is τ. In the case of proving GES under ADT constraints we want to find multiple Lyapunov functions (one for each subsystem) with additional conditions on each function and in relation to the other Lyapunov functions [14, Chapter 3]. That is, we find Lyapunov functions V_i such

[3] Note that $\langle \nabla (\mathbf{x}^T P \mathbf{x}), A_i \mathbf{x} \rangle = \mathbf{x}^T (A_i^T P + P A_i) \mathbf{x}$

that

$$V_i(\mathbf{x}) > 0 \qquad \mathbf{x} \in \mathbb{R}^n \setminus \{\mathbf{0}\},\ \forall i \in [\![1,N]\!], \quad (6a)$$
$$V_i(\mathbf{0}) = 0 \qquad \forall i \in [\![1,N]\!] \quad (6b)$$
$$\langle \nabla V_i(\mathbf{x}), A_i\mathbf{x}\rangle \leq -\alpha V_i(\mathbf{x}) \qquad \forall \mathbf{x} \in \mathbb{R}^n \setminus \{\mathbf{0}\},\ \forall i \in [\![1,N]\!] \quad (6c)$$
$$V_i(\mathbf{x}) \leq \mu V_j(\mathbf{x}) \qquad \forall \mathbf{x} \in \mathbb{R}^n,\ \forall i,j \in [\![1,N]\!] \quad (6d)$$

then (3) is GES if the switching signal has an ADT

$$\tau > \tau^\star = \frac{\ln(\mu)}{\alpha}.$$

We generally call $\tau^\star$ a MADT rather than ADT given that this holds true for all ADTs τ greater than $\tau^\star$. Consequently, the main point of interest is finding a feasible (α,μ) pair such that (6) has a solution V_i for all $i \in [\![1,N]\!]$ with the smallest possible τ, i.e. the "absolute" MADT. This is the main subject of Paper III [2].

The primary bottleneck here is that one cannot simply minimize the function $\ln(\mu)/\alpha$ without taking into account that (6) must also have a feasible solution. To that end we considered an area $D \subset \mathbb{R}\times\mathbb{R}$ defined as

$$D \coloneqq \{(\alpha,\mu) \in (0,\infty)\times[1,\infty) : (6) \text{ is feasible}\},$$

and are interested in minimizing τ with (α,μ) on D. This poses a question; Can we determine the area analyticity or numerically?

Alternatively we can describe D with

$$D = \left\{(\alpha,\mu) \in \mathbb{R}\times\mathbb{R} : 0 < \alpha < \overline{\alpha} \quad \text{and} \quad \mu > \underline{\mu}(\alpha)\right\},$$

and aim to determine $\overline{\alpha}$ and the function $\underline{\mu}(\alpha)$, see Figure 2.3 for a schematic view of this interpretation. The upper bound on $\overline{\alpha}$ can be found via analytical means dependent on the Lyapunov function classes.

As a demonstrative example, let V_i be quadratic: $V_i = \mathbf{x}^T P_i \mathbf{x}$. Condition (6c) yields

$$\langle \nabla V_i, A_i\mathbf{x}\rangle = \mathbf{x}^T(A_i^T P_i + P_i A_i)\mathbf{x} \leq -\alpha \mathbf{x}^T P_i \mathbf{x},$$

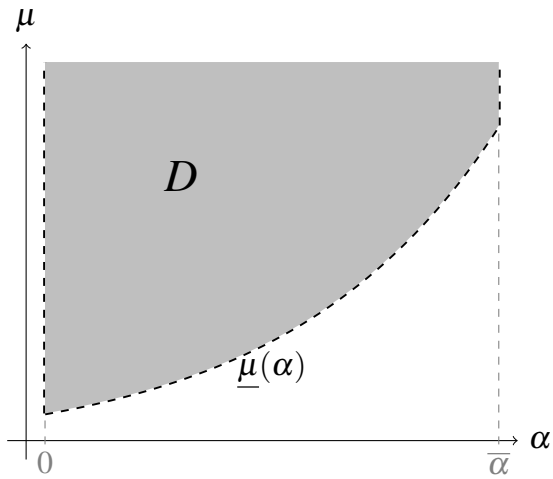


Figure 2.3. An example of the area D where (6) *has a feasible solution given* α *and* μ*. Note that the area extends indefinitely as* $\mu \to \infty$*. Adapted from [2].*

that is,

$$A_i^T P_i + P_i A_i + \alpha P_i = \left(A_i + \frac{\alpha}{2} I\right)^T P_i + P_i \left(A_i + \frac{\alpha}{2} I\right) \preceq 0.$$

Thus if $P_i \succ 0$ we need the real part of all the eigenvalues of A_i to be strictly smaller than $-\alpha/2$ and consequently

$$\overline{\alpha} = \sup \left\{ \alpha > 0 : A_i + \frac{\alpha}{2} I \text{ is Hurwitz for all } i \in [\![1, N]\!] \right\}.$$

A similar manipulation of (6c) can yield an $\overline{\alpha}$ for different function classes, see Paper III [2].

Determining the function $\underline{\mu}(\alpha)$ is more difficult. Throughout our numerical experiments we found it to be convex and the minimum of $\tau(\alpha, \mu)$ is achieved along the curve of $\underline{\mu}(\alpha)$. We therefore aimed at devising a method that searches for a feasible (α, μ) set close to $\underline{\mu}(\alpha)$ and then follows that path with increasing α until the minimum is found. Note it is worth checking for a CLF, with $(\alpha, \mu) = (0, 1)$, which corresponds to arbitrary switching before looking for a MADT. This numerical search routine, presented in Paper III [2], is formalized with Algorithm 1

Algorithm 1. *We fix $K, L \in \mathbb{N}$, $K \geq 2$, and $a, b > 0$ and distribute the values $a = \tau_K < \tau_{K-1} < \ldots < \tau_2 < \tau_1 = b$ and $0 < \alpha_1 < \ldots < \alpha_L < \overline{\alpha}$ uniformly, i.e.*

$$\tau_j = (a-b)\frac{j-1}{K-1} + b \quad \text{and} \quad \alpha_i = \overline{\alpha}\frac{i}{L+1}$$

for $j \in [\![1, K]\!]$ and $i \in [\![1, L]\!]$.

Set $i = 1$ and $j = 1$. Then execute:

1. *Solve the feasibility problem* (6) *with $\alpha = \alpha_i$ and $\mu = \exp(\tau_j \alpha_i)$.*
2. - *If there is a feasible solution to the problem, then increase j by one, i.e. $j \leftarrow j + 1$, and go back to step (1). That is, we decrease τ from τ_j to τ_{j+1} and check if this problem, with a lower average dwell-time and a smaller $\mu = \exp(\tau_{j+1}\alpha_i)$, also has a feasible solution.*
   - *If there does not exist a feasible solution to the problem, then increase i by one, i.e. $i \leftarrow i + 1$, and go back to step (1). That is, we increase α from α_i to α_{i+1} and set $\mu = \exp(\tau_j \alpha_{i+1})$ and check if this problem, with the same average dwell-time $\tau = \tau_j$, but with larger parameters α and μ, has a feasible solution.*

The efficacy of this method is of course dependent on the numerical method chosen for the feasibility problem based on (6) but we repeatedly found a lower MADT than in previous literature. For a more detailed discussion on this subject see Paper III [2].

3 Numerical methods

Figure 3.1 shows examples of the three function classes primarily considered for Lyapunov functions; quadratic, continuous piecewise affine (CPA), and continuous piecewise quadratic (CPQ). Different programming methods are applicable when considering these classes. Paper II [5] discusses the two methods we use, linear programming (LP) and SDP.

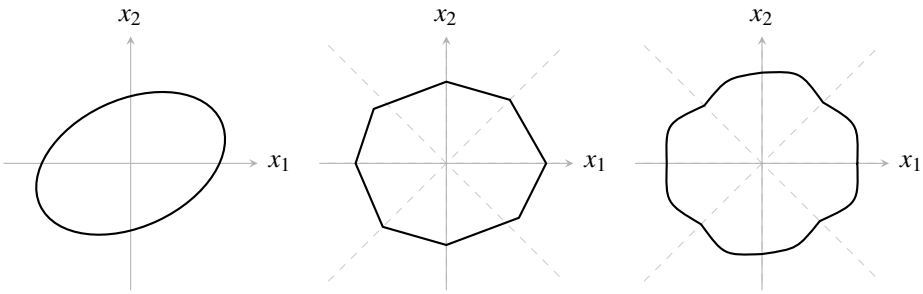
To facilitate the discussion of CPA and CPQ functions let us first review triangulations. A triangulation $\mathcal{T}$ with vertices $\{\mathbf{v}_0, \mathbf{v}_1, \ldots, \mathbf{v}_p\} \subset \mathbb{R}^n$ is a subdivision of a subset of $\mathbb{R}^n$ into simplices $\mathfrak{S}$. We define a simplex $\mathfrak{S}_\nu$ as

$$\begin{aligned}\mathfrak{S}_\nu &\coloneqq \operatorname{co}\{\mathbf{x}_0^\nu, \mathbf{x}_1^\nu, \mathbf{x}_2^\nu, \ldots, \mathbf{x}_n^\nu\} \\ &= \left\{ \mathbf{x} \in \mathbb{R}^n : \mathbf{x} = \sum_{r=0}^{n} \lambda_r \mathbf{x}_r^\nu,\ \lambda_r \geq 0,\ \sum_{r=0}^{n} \lambda_r = 1 \right\}\end{aligned}$$

where $\mathbf{x}_r^\nu = \mathbf{v}_{g_\nu(r)}$ for $r \in [\![0,n]\!]$ and $g_\nu : [\![0,n]\!] \to [\![1,p]\!]$ is an index-function. For our purposes we require $\mathcal{T}$ to be shape-regular, i.e. every two different simplices either intersect at a common lower dimensional face or not at all. Additionally, we consider a triangulation around the origin such that the origin is a vertex of every simplex. The set theoretic union of all $\mathfrak{S}_\nu \in \mathcal{T}$ must be a neighbourhood of the origin denoted $\mathcal{D}_\mathcal{T}$. An efficient implementation of a triangulation that satisfies these conditions, shown in Figure 3.2, is the triangular fan discussed in [9], for a more details on the technical implementation see [8].

The neighbourhood $\mathcal{D}_\mathcal{T}$ can be radially expanded to $\mathbb{R}^n$ by defining cones $\mathfrak{C}_\nu$ corresponding to each simplex $\mathfrak{S}_\nu$ such that

$$\begin{aligned}\mathfrak{C}_\nu &\coloneqq \operatorname{cone}\{\mathbf{0}, \mathbf{x}_1^\nu, \mathbf{x}_2^\nu, \ldots, \mathbf{x}_n^\nu\} \\ &= \left\{ \mathbf{x} \in \mathbb{R}^n : \mathbf{x} = \sum_{r=1}^{n} \lambda_r \mathbf{x}_r^\nu,\ \lambda_r \geq 0 \right\}\end{aligned}$$



(a) *(b)* *(c)*

Figure 3.1. Level sets of $V : \mathbb{R}^2 \to \mathbb{R}$ for the three different function classes under consideration: (a) quadratic, (b) continuous piecewise affine (CPA), and (c) continuous piecewise quadratic (CPQ) functions.

where $\{\mathbf{0}, \mathbf{x}_1^\nu, \mathbf{x}_2^\nu, \ldots, \mathbf{x}_n^\nu\}$ are the vertices that define the simplex $\mathfrak{S}_\nu$. We call the λ_r the cone-coordinates of $\mathbf{x}$ and can define the vertex matrix

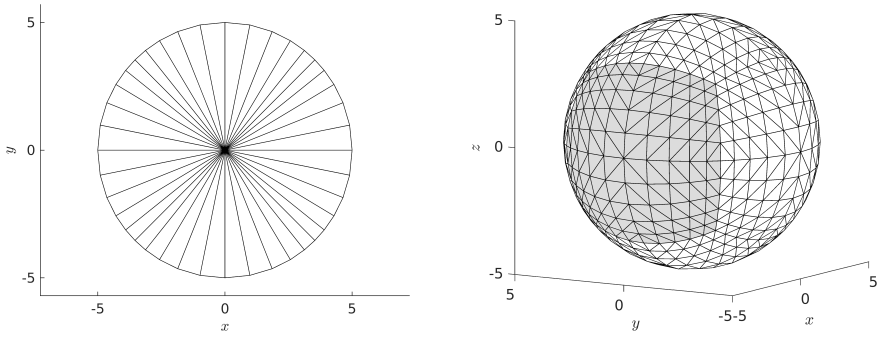
$$X_\nu = \begin{bmatrix} | & | & & | \\ \mathbf{x}_1^\nu & \mathbf{x}_2^\nu & \ldots & \mathbf{x}_n^\nu \\ | & | & & | \end{bmatrix} \in \mathbb{R}^{n \times n},$$

that is the non-zero vertices of the simplex make up the column vectors of X_ν, and write $\mathbf{x} = X_\nu \lambda$ for all $\mathbf{x} \in \mathfrak{C}_\nu$. Note that this matrix X_ν depends on the order of the vertices, so this order must be fixed.

Remark. *In general we use ν to denote the index of a simplex or corresponding cone. To denote the indices of subsystems we use i and j, and for vertices we use r, k and ℓ. To denote the system dimensions we use n and the number of subsystems is N. The number of non-zero vertices in the triangulation is denoted with p. As a demonstrative example of the notation consider* (9) *rewritten here:*

$$\sum_{\ell=1}^{n} r_{k,\ell}^{\nu,i,j} < 0$$

The variable $r_{k,\ell}^{\nu,i,j}$ is the (k,ℓ)-th element of the matrix $R_{i,j}^\nu$ which corresponds to ν-th simplex, $\mathfrak{S}_\nu$, and i-th and j-th subsystems, A_i and A_j respectively.



(a) *(b)*

Figure 3.2. Example of the triangular fan used in (a) two dimensions and (b) three dimensions. The shaded area in (b) corresponds to one face of the cube constructed as an intermediate step in the triangulation definition. Note that the origin is a vertex in both cases. Adapted from [1].

Quadratic functions

We define quadratic functions with a matrix $P \in \mathbb{R}^{n\times n}$ such that

$$V(\mathbf{x}) = \mathbf{x}^T P \mathbf{x}.$$

Continuous piecewise affine functions

We define CPA Lyapunov functions with a vector $\Phi \in \mathbb{R}^{p+1}$ such that $V(\mathbf{v}_r) := \varphi_r$, and for each simplex $\mathfrak{S}_\nu$ we define the vector $\Psi^\nu := \left(\varphi_{g_\nu(1)}, \varphi_{g_\nu(2)}, \ldots, \varphi_{g_\nu(n)}\right)^T$ where g_ν is an indexing function. Then for every $\mathbf{x} \in \mathfrak{C}_\nu$ we have

$$V(\mathbf{x}) = (\Psi^\nu)^T X_\nu^{-1} \mathbf{x} = (\Psi^\nu)^T \boldsymbol{\lambda}$$

where $\boldsymbol{\lambda} \in \mathbb{R}^n_+$ are the cone coordinates of $\mathbf{x}$. This parametrization of CPA functions was covered in [4] where a LP based numerical approach for common Lyapunov functions was presented, for further reading see also Paper II and [1].

Continuous piecewise quadratic functions

We define CPQ Lyapunov functions with matrices $\Phi \in \mathbb{R}^{p+1\times p+1}$ such that $V(\mathbf{x}) = \mathbf{x}^T P^\nu \mathbf{x}$ where $P^\nu = (X_\nu^{-1})^T \Psi^\nu X_\nu^{-1}$. The (k,ℓ)-th element of

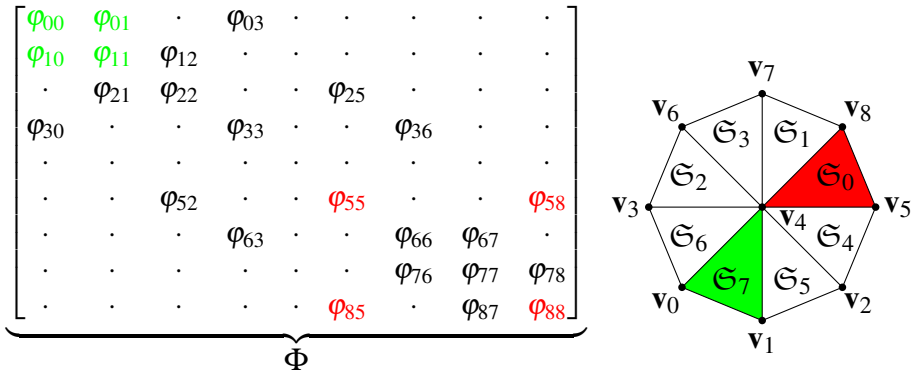


Figure 3.3. Illustration of the global matrix Φ (left) and a triangulation with simplices and vertices marked (right). Using the indexing function g_ν, the highlighted submatrices Ψ^7 (green) and Ψ^0 (red) are extracted from Φ, containing the decision variables corresponding to the vertex indices of simplices $\mathfrak{S}_7$ and $\mathfrak{S}_0$, respectively.

$\Psi^\nu \in \mathbb{R}^{n\times n}$ is equal to the $(g_\nu(k), g_\nu(\ell))$-th element of Φ, again g_ν is an indexing function. Moreover see Ψ^ν is a principal submatrix of Φ. See Figure 3.3 for a more intuitive view of the relationship between Φ and Ψ. Then $V(\mathbf{x}) = \lambda^T \Psi^\nu \lambda$ where $\lambda \in \mathbb{R}^n_+$ are the cone-coordinates of $\mathbf{x} \in \mathfrak{C}_\nu$. This parametrization of CPQ functions was suggested in [19] and its use as common Lyapunov functions is the subject of Paper IV [18].

3.1 CPA functions

From previous literature [1, 4] we have an LP problem for computing CPA CLFs.

LP Problem 1. *For a triangulation $\mathcal{T}$ with vertices $\{\mathbf{v}_0, \mathbf{v}_1, \ldots, \mathbf{v}_p\}$ and simplices $\mathfrak{S}_\nu := \text{co}\{\mathbf{0}, \mathbf{x}_1^\nu, \mathbf{x}_2^\nu, \ldots, \mathbf{x}_n^\nu\}$, we have $X_\nu = [\mathbf{x}_1^\nu, \mathbf{x}_2^\nu, \ldots, \mathbf{x}_n^\nu]$ and index-functions g_ν such that $\mathbf{x}_r^\nu = \mathbf{v}_{g_\nu(r)}$. On every simplex $\mathfrak{S}_\nu$ and for each subsystem A_i*

1. $\varphi_{g_\nu(r)} > 0$, *if* $\mathbf{v}_{g_\nu(r)} \neq 0$
2. $(\Psi_i^\nu)^T X_\nu^{-1} A_i \mathbf{x}_r^\nu < 0$

The problem variables are the elements of the vector Φ.

In Paper III [2] we apply this idea to MADT switching:
LP Problem 2. *For a triangulation* $\mathcal{T}$ *with vertices* $\{\mathbf{v}_0, \mathbf{v}_1, \ldots, \mathbf{v}_p\}$ *and simplices* $\mathfrak{S}_\nu := \mathrm{co}\{\mathbf{0}, \mathbf{x}_1^\nu, \mathbf{x}_2^\nu, \ldots, \mathbf{x}_n^\nu\}$, *we have* $X_\nu = [\mathbf{x}_1^\nu, \mathbf{x}_2^\nu, \ldots, \mathbf{x}_n^\nu]$ *and index-functions* g_ν *such that* $\mathbf{x}_r^\nu = \mathbf{v}_{g_\nu(r)}$. *On every simplex* $\mathfrak{S}_\nu$ *and for each subsystem* A_i, *with a fixed* $\mu \geq 1$ *and* $\alpha > 0$ *we have*

1. $\varphi^i_{g_\nu(r)} > 0$, *if* $\mathbf{v}_{g_\nu(r)} \neq 0$
2. $(\Psi_i^\nu)^T X_\nu^{-1} A_i \mathbf{x}_r^\nu + \alpha (\Psi_i^\nu)^T X_\nu^{-1} \mathbf{x}_r^\nu \leq 0$ [4]
3. $\varphi^i_{g_\nu(r)} - \mu \varphi^j_{g_\nu(r)} \leq 0$

The problem variables are the elements of the vectors Φ_i.

3.2 CPQ functions

In Paper IV [18], along with Paper I [6], we presented a method for the construction of CPQ CLFs with LP. Simply applying the conditions of (5) globally is inherently conservative, since a matrix can be p.d. on a subset of $\mathbb{R}^n$ and not p.d. on another disjoint subset (analogous to a saddle surface rather than a bowl for $\mathbf{x}^T P \mathbf{x}$ with $\mathbf{x} \in \mathbb{R}^2$). We therefore only consider the positive definiteness of each individual piece of the CPQ function, that is for every simplex $\mathfrak{S}_\nu$ we have

$$P^\nu \succ_{\mathfrak{C}_\nu} 0, \tag{7a}$$

$$A_i^T P^\nu + P^\nu A_i \prec_{\mathfrak{C}_\nu} 0 \qquad \forall i \in [\![1, N]\!]. \tag{7b}$$

Before we present the LP problems, let us first consider the following theorem, paraphrased from Paper IV.
Theorem 3. *Let* $B \in \mathbb{R}^{n \times n}$ *be a symmetric matrix and define* $C \in \mathbb{R}^{n \times n}$ *such that*

$$\begin{aligned} c_{kk} &\coloneqq b_{kk}, & k &\in [\![1, n]\!], \\ c_{k\ell} &\coloneqq \max(b_{k\ell}, 0), & k, \ell &\in [\![1, n]\!], k \neq \ell. \end{aligned}$$

If each row sum of C *is negative, then* $-B$ *is copositive.*

[4] Alternatively $(\Psi_i^\nu)^T X_\nu^{-1} A_i \mathbf{x}_r^\nu + \alpha \varphi^i_{g_\nu(r)} < 0$

Proof. [5] First note that $c_{k\ell} \geq 0$ and for the k-th row of C we have

$$0 > \sum_{\ell=1}^{n} c_{k\ell} = c_{kk} + \sum_{\substack{\ell=1 \\ \ell \neq k}}^{n} c_{k\ell} \implies |c_{kk}| > \sum_{\substack{\ell=1 \\ \ell \neq k}}^{n} |c_{k\ell}| .$$

Namely, C is a symmetric, strictly diagonally dominant matrix with negative elements on the diagonal and therefore negative definite. In particular, we have $\lambda^T C\lambda < 0$ with $\lambda \in \mathbb{R}^n_+ \setminus \{\mathbf{0}\}$. Now note that $c_{k\ell} \geq b_{k\ell}$ and

$$\begin{aligned} \lambda^T C\lambda - \lambda^T B\lambda &= \lambda^T (C-B)\lambda \\ &= \sum_{k=1}^{n} \underbrace{\lambda_k^2 (c_{kk} - b_{kk})}_{=0} + \sum_{k=1}^{n} \sum_{\substack{\ell=1 \\ \ell \neq k}}^{n} \underbrace{\lambda_k \lambda_\ell (c_{k\ell} - b_{k\ell})}_{\geq 0} \geq 0 \end{aligned}$$

That is $\lambda^T B\lambda \leq \lambda^T C\lambda < 0$, or more precisely $\lambda^T(-B)\lambda > 0$, for all $\lambda \in \mathbb{R}^n_+ \setminus \{\mathbf{0}\}$. □

Note that Theorem 3 is only a sufficient condition resulting in a constraint more restrictive than the original (7).

Remark. *Let* $\mathbf{x} \in \mathfrak{C}_\nu := \text{cone}\{\mathbf{0}, \mathbf{x}_1^\nu, \mathbf{x}_2^\nu, \ldots, \mathbf{x}_n^\nu\}$. *We can write* $\mathbf{x} = X_\nu \lambda$, *with* $\lambda \in \mathbb{R}^n_+ \setminus \{\mathbf{0}\}$, *and*

$$\mathbf{x}^T M\mathbf{x} = \lambda^T X_\nu^T M X_\nu \lambda$$

That is, a matrix $M \in \mathbb{R}^{n\times n}$ *is p.d. on* $\mathfrak{C}_\nu$ *if and only if* $X_\nu^T M X_\nu$ *is copositive.*

To summarize, we solve for a matrix $\Phi \in \mathbb{R}^{p\times p}$, with principal submatrices Ψ^ν as described above, such that on each simplex $\mathfrak{S}_\nu$ we set

$$\begin{aligned} P^\nu &:= \left(X_\nu^{-1}\right)^T \Psi^\nu X_\nu^{-1}, \\ Q_i^\nu &:= A_i^T P^\nu + P^\nu A_i, \end{aligned}$$

for each $i \in [\![1, N]\!]$. Define $B_i^\nu := X_\nu^T Q_i^\nu X_\nu = \left(\Psi^\nu \hat{A}_i^\nu\right)^T + \Psi^\nu \hat{A}_i^\nu$ where $\hat{A}_i^\nu := X_\nu^{-1} A_i X_\nu$, and C_i^ν is constructed as in Theorem 3. To avoid nonlinear functions, the condition $c_{k\ell} = \max(b_{k\ell}, 0)$ is replaced by the linear constraints $c_{k\ell} \geq 0$ and $c_{k\ell} \geq b_{k\ell}$ for $k \neq \ell$.

[5] Note this is an alternative proof to the one in Paper IV.

While constraints like (5) and (7) are often solved with SDP implemented in interpreted environments like Matlab [17], that approach does not benefit from the efficiency associated with compiled languages. In Paper I [6], we present a C++ parser for the C Gurobi [7] syntax and compare the computation time of CPQ CLFs between Matlab and C++.

Paper I [6] we presented an LP problem in the cases of arbitrary switching:

LP Problem 3. *For a triangulation $\mathcal{T}$ with vertices $\{\mathbf{v}_0, \mathbf{v}_1, \ldots, \mathbf{v}_p\}$ and simplices $\mathfrak{S}_\nu := \text{co}\{\mathbf{0}, \mathbf{x}_1^\nu, \mathbf{x}_2^\nu, \ldots, \mathbf{x}_n^\nu\}$, we have $X_\nu = [\mathbf{x}_1^\nu, \mathbf{x}_2^\nu, \ldots, \mathbf{x}_n^\nu]$ and index-functions g_ν such that $\mathbf{x}_r^\nu = \mathbf{v}_{g_\nu(r)}$. On every simplex $\mathfrak{S}_\nu$ and for each subsystem A_i we have*

1. *$V(\mathbf{x}_r^\nu) > 0$ for all $r \in [\![1, n]\!]$.*
2. *For each subsystem define $\hat{A}_i^\nu := X_\nu^{-1} A_i X_\nu$ and C_i^ν such that*

$$c_{k,k}^{\nu,i} = 2 \sum_{r=1}^{n} \varphi_{g_\nu(r), g_\nu(k)} \hat{a}_{r,k}^{\nu,i},$$

for all $k \in [\![1, n]\!]$ and

$$c_{k,\ell}^{\nu,i} \geq \sum_{r=1}^{n} \left(\varphi_{g_\nu(\ell), g_\nu(r)} \hat{a}_{r,k}^{\nu,i} + \varphi_{g_\nu(k), g_\nu(r)} \hat{a}_{r,\ell}^{\nu,i} \right),$$
$$c_{k,\ell}^{\nu,i} \geq 0$$

for all $k, \ell \in [\![1, n]\!]$ and $k \neq \ell$. Then require that for all $k \in [\![1, n]\!]$

$$\sum_{\ell=1}^{n} c_{k,\ell}^{\nu,i} < 0 \quad \forall k \in \{1, 2, \ldots, n\}.$$

The problem variables are the elements of the matrices Φ and C_i^ν.

Remark. *Note that $V(\mathbf{v}_i) > 0$ for all nonzero vertices $\mathbf{v}_i$ of $\mathcal{T}$ does not ensure that $V(\mathbf{x}) > 0$ for all $\mathbf{x} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$. This can however be checked a posteriori to solving the LP problem, e.g. by checking whether there exists a solution to the LMI*

$$P^\nu - (X_\nu^{-1})^T U_\nu X_\nu^{-1} \succ 0,$$

where $U_\nu \in \mathbb{R}_+^{n \times n}$ is the problem variable. This a posteriori check was proposed in [12, Proposition 4.4]. For further reading on checking positivity of CPQ functions see [10].

A greater focus in our research was investigating MADT constraints. Paper V [3], compares CPQ and CPA Lyapunov functions and the number of simplices and variables needed to achieve the same results with respects to MADT.

Similarly as the conditions in (4) become (7), via (5), we have the conditions in (6) and write for every simplex $\mathfrak{S}_\nu$:

$$\begin{aligned} P_i^\nu &\succ_{\mathfrak{C}_\nu} 0 && \forall i \in [\![1,N]\!] \\ A_i^T P_i^\nu + P_i^\nu A_i + \alpha P_i^\nu &\preceq_{\mathfrak{C}_\nu} 0 && \forall i \in [\![1,N]\!] \\ P_i^\nu - \mu P_j^\nu &\preceq_{\mathfrak{C}_\nu} 0 && \forall i,j \in [\![1,N]\!] \end{aligned}$$

Using the same relaxation method as before we define the LP problem as follows

LP Problem 4. *For a triangulation $\mathcal{T}$ with vertices $\{\mathbf{v}_0,\mathbf{v}_1,\ldots,\mathbf{v}_p\}$ and simplices $\mathfrak{S}_\nu := \mathrm{co}\{\mathbf{0},\mathbf{x}_1^\nu,\mathbf{x}_2^\nu,\ldots,\mathbf{x}_n^\nu\}$, we have $X_\nu = [\mathbf{x}_1^\nu,\mathbf{x}_2^\nu,\ldots,\mathbf{x}_n^\nu]$ and index-functions g_ν such that $\mathbf{x}_r^\nu = \mathbf{v}_{g_\nu(r)}$. On every simplex $\mathfrak{S}_\nu$ and for each subsystem A_i, with a fixed $\mu \geq 1$ and $\alpha > 0$ we have*

1. *$V(\mathbf{x}_r^\nu) > 0$ for all $r \in [\![1,n]\!]$.*
2. *For each subsystem define $\hat{A}_i^\nu := X_\nu^{-1} A_i X_\nu$ and Q_i^ν such that*

$$q_{k,k}^{\nu,i} = \alpha \varphi^i_{g_\nu(k),g_\nu(k)} + 2\sum_{r=1}^{n} \hat{a}_{r,k}^{\nu,i} \varphi^i_{g_\nu(r),g_\nu(k)},$$

for all $k \in [\![1,n]\!]$ and

$$\begin{aligned} q_{k,\ell}^{\nu,i} &\geq \left(\sum_{r=1}^{n} \hat{a}_{r,k}^{\nu,i} \varphi^i_{g_\nu(r),g_\nu(\ell)} + \hat{a}_{r,\ell}^{\nu,i} \varphi^i_{g_\nu(k),g_\nu(r)} \right) + \alpha \varphi^i_{g_\nu(k),g_\nu(\ell)} \\ q_{k,\ell}^{\nu,i} &\geq 0, \end{aligned}$$

for all $k,\ell \in [\![1,n]\!]$ and $k \neq \ell$. Then require that for all $k \in [\![1,n]\!]$

$$\sum_{\ell=1}^{n} q_{k,\ell}^{\nu,i} < 0$$

3. *Define $R_{i,j}^\nu$ such that*

$$r_{k,k}^{\nu,i,j} = \varphi^i_{g_\nu(k),g_\nu(k)} - \mu \varphi^j_{g_\nu(k),g_\nu(k)},$$

for all $k \in [\![1,n]\!]$ *and*

$$r_{k,\ell}^{\nu,i,j} \geq \varphi^i_{g_\nu(k),g_\nu(\ell)} - \mu\,\varphi^j_{g_\nu(k),g_\nu(\ell)},$$
$$r_{k,\ell}^{\nu,i,j} \geq 0,$$

for all $k,\ell \in [\![1,n]\!]$ *and* $k \neq \ell$*. Then require that for all* $k \in [\![1,n]\!]$

$$\sum_{\ell=1}^{n} r_{k,\ell}^{\nu,i,j} \leq 0 \tag{9}$$

The problem variables are the elements of the matrices Φ_i*,* Q_i^ν *and* $R_{i,j}^\nu$*.*

4 Conclusions

This dissertation has researched numerical methods used for stability analysis of switched linear systems under both arbitrary switching and average dwell-time (ADT) constraints. It presents a linear programming (LP)-based method for computing continuous piecewise quadratic (CPQ) Lyapunov functions for both types of switching signals.

Beyond this parametrization of CPQ functions, this research addresses the complexity inherent in computing the minimum average dwell-time (MADT) that ensures the stability of a switched linear system. The proposed method of considering the feasibility region D defined by $\overline{\alpha}$ and the function $\underline{\mu}(\alpha)$ highlighted two main observations regarding the MADT: that it lies along the border of $\underline{\mu}(\alpha)$ and that this function is convex. Both statements have yet to be proven, and the question of whether $\underline{\mu}(\alpha)$ can be found via analytic means or a more efficient method than the proposed Algorithm 1 is an open question as well.

Earlier research into continuous piecewise affine (CPA) Lyapunov functions was expanded upon. An LP-based numerical method using CPA Lyapunov functions to compute the MADT was presented and, using Algorithm 1, was compared to quadratic functions found via semidefinite programming (SDP). In addition to offering greater geometric flexibility than quadratic functions, leading to less conservative stability conditions, the use of LP in conjunction with a C++ implementation yielded favourable results compared to SDP in an interpreted programming environment, achieving both faster execution speeds and lower MADT bounds with CPA functions.

Lastly, the use of CPQ Lyapunov functions was compared to CPA Lyapunov functions, with respect to numerical complexity, with favourable results. As suggested in earlier research, the efficacy of CPA functions, with respect to the number of simplices, is dependent on the triangulation and coordinate system used. However, tailoring triangulations is particu-

larly difficult, and subsystem preconditioning failed to yield any concrete results, rendering this approach unreliable. The use of CPQ functions successfully combats the orientation issue, resulting in fewer simplices and problem variables without using a coordinate change.

References

[1] S. Andersen. Common Lyapunov Functions for Switched Linear Systems: Computations and comparisons of numerical approaches. Master's thesis, University of Iceland, October 2023.

[2] S. Andersen and S. Hafstein. Investigating Average Dwell-Time Constraints on Multiple Lyapunov Functions for Switched Linear Systems. *IFAC-PapersOnLine*, 59(19):150–155, 2025. ISSN 2405-8963. 13th IFAC Symposium on Nonlinear Control Systems NOLCOS 2025.

[3] S. Andersen and S. Hafstein. Minimum average dwell-time computations via continuous piecewise quadratic lyapunov functions. In *2026 30th International Conference on Methods and Models in Automation and Robotics (MMAR)*, pages 351–356, 2026.

[4] S. Andersen, P. Giesl, and S. Hafstein. Common Lyapunov functions for switched linear systems: Linear programming-based approach. *IEEE Control Systems Letters*, 7:901–906, 2023.

[5] S. Andersen, E. August, S. Hafstein, and J. Piccini. Numerical Computations of Lyapunov Functions for Switched Linear Systems. In G. Wagner, F. Werner, and F. De Rango, editors, *Simulation and Modeling Methodologies, Technologies and Applications*, pages 192–213, Cham, 2024. Springer Nature Switzerland.

[6] S. Andersen, S. F. Hafstein, J. J. Palacios Roman, and S. J. A. M. van den Eijnden. Efficient Implementation of Piecewise Quadratic Lyapunov Function Computations for Switched Linear Systems. In G. Gini, R. Precup, and D. P. Filev, editors, *Proceedings of the 21st International Conference on Informatics in Control, Automation and Robotics, ICINCO 2024, Porto, Portugal, November 18-20, 2024, Volume 1*, pages 277–284. SciTePress, 2024.

[7] Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2023. URL `https://www.gurobi.com`.

[8] S. Hafstein. Implementation of simplicial complexes for CPA functions in C++11 using the Armadillo linear algebra library. In *Proceedings of the 3rd International Conference on Simulation and Modeling Methodologies, Technologies and Applications (SIMULTECH)*, pages 49–57, Reykjavik, Iceland, 2013.

[9] S. Hafstein. *Simulation and Modeling Methodologies, Technologies and Applications*, volume 873 of *Advances in Intelligent Systems and Computing*, chapter Fast Algorithms for Computing Continuous Piecewise Affine Lyapunov Functions, pages 274–299. Springer, 2019.

[10] S. Hafstein and E. Hafsteinsson. Verifying Positivity of Piecewise Quadratic Lyapunov Functions. In *Proceedings of the 22nd International Conference on Informatics in Control, Automation and Robotics - Volume 1: ICINCO*, pages 435–444. INSTICC, SciTePress, 2025.

[11] J. Hespanha and A. Morse. Stability of switched systems with average dwell-time. In *Proceedings of the 38th IEEE Conference on Decision and Control (Cat. No.99CH36304)*, volume 3, pages 2655–2660, 1999.

[12] M. Johansson. *Piecewise Linear Control Systems*. PhD thesis, Lund University, Sweden, 1999.

[13] R. E. Kalman and J. E. Bertram. Control system analysis and design via the "second method" of Lyapunov: I. Continuous-time systems. *Journal of Basic Engineering*, 82(2):371–393, 1960.

[14] D. Liberzon. *Switching in systems and control*. Systems & Control: Foundations & Applications. Birkhäuser, 2003.

[15] A. M. Lyapunov. *Obshchaya zadacha ob ustoychivosti dvizheniya [The general problem of the stability of motion]*. Kharkovskoe Matematicheskoe Obshchestvo, Kharkov, 1892. (In Russian).

[16] A. M. Lyapunov. *The general problem of the stability of motion*. Taylor & Francis, London, 1992. Translated by A. T. Fuller.

[17] MATLAB. *9.12.0.2039608 (R2022a) Update 5*. The MathWorks Inc., Natick, Massachusetts, 2022.

[18] J. J. Palacios Roman, S. Hafstein, P. Giesl, S. van den Eijnden, S. Andersen, and W. Heemels. Constructing piecewise quadratic Lyapunov functions with linear programming for continuous-time switched and conewise linear systems. *Automatica*, 186:112838, 2026.

[19] A. Rantzer and M. Johansson. Piecewise linear quadratic optimal control. *IEEE Transactions on Automatic Control*, 45(4):629–637, 2000.

Paper I

Efficient Implementation of Piecewise Quadratic Lyapunov Function Computations for Switched Linear Systems.

S. Andersen, S. Hafstein, J. Roman and S.J.A.M van den Eijnden

Proceedings of the 21st International Conference on Informatics in Control, Automation and Robotics (ICINCO), pp. 277–284, 2024.

Reproduced with permission SciTePress, Science and Technology Publications, Lda.

We describe a linear programming (LP) problem to parameterize continuous and piecewise quadratic (CPQ) Lyapunov functions for switched linear systems. We discuss some algorithms and data-structures for its implementation in C++ and compare the computational efficiency of our implementation to an analogous implementation in MATLAB.

Efficient Implementation of Piecewise Quadratic Lyapunov Function Computations for Switched Linear Systems

Stefania Andersen[1][a], Sigurdur Hafstein[1][b], Juan Javier Palacios Roman[2][c] and Sebastiaan J.A.M. van den Eijnden[2][d e]

[1] *University of Iceland, Faculty of Physical Sciences, Dunhagi 5, 107 Reykjavik, Iceland*

[2] *Eindhoven University of Technology, Department of Mechanical Engineering, Groene Loper 3, 5612 AE Eindhoven*

{*saa20, shafstein*}*@hi.is,* {*j.j.palacios.roman, s.j.a.m.v.d.eijnden*}*@tue.nl*

Keywords: Common Lyapunov Function, Switched Linear Systems, Linear Programming, C++, MATLAB

Abstract: We describe a linear programming (LP) problem to parameterize continuous and piecewise quadratic (CPQ) Lyapunov functions for switched linear systems. We discuss some algorithms and data-structures for its implementation in C++ and compare the computational efficiency of our implementation to an analogous implementation in MATLAB.

1 INTRODUCTION

Switched systems play an important role in modelling in science and engineering (Davrazos and Koussoulas, 2001; Liberzon, 2003; Shorten et al., 2007; Sun and Ge, 2011). In control theory the stability of an equilibrium point is commonly of central importance and is most conveniently dealt with using the Lyapunov stability theory of dynamical systems (Hahn, 1967; Sastry, 1999; Khalil, 2002; Vidyasagar, 2002). It is well known that the existence of a Lyapunov function for a switched system, which is a common Lyapunov function for all the subsystems, is equivalent to its stability and suitable classes of potential Lyapunov functions is a thoroughly researched subject (Dayawansa and Martin, 1999; Goebel et al., 2006; Mason et al., 2006; Shorten et al., 2007; Mason et al., 2022). In this paper we consider the algorithm from (Palacios Roman et al., 2024) to parameterize continuous and piecewise quadratic (CPQ) Lyapunov functions for switched linear systems along with its efficient implementation in the programming language C++. The class of CPQ Lyapunov functions have been considered in (Johansson and Rantzer, 1998), and have recently been successfully used to study the stability of, hybrid integrator-gain systems (HIGS), see e.g. (van den Eijnden et al., 2020; Deenen et al., 2021; van den Eijnden et al., 2022).

The main contribution of this work is the efficient computation of CPQ Lyapunov functions for switched linear systems based on a linear program, where the classical constraints for CPQ Lyapunov functions, which are often expressed in terms of linear matrix inequalities, are formulated as a linear programming (LP) problem.

This paper is organized as follows: In Section 2 we describe how we write the LP problem in our implementation, before we discuss in Section 3 triangulations, CPQ functions, and how we parameterize them. Section 4 deals with the linear constraints we use to compute a suitable parameterization for a CPQ Lyapunov function for a given switched linear system. Then, in Section 5, we give some details on the implementation of the linear constraints and compare the numerical efficiency of our implementation in C++ to a corresponding implementation in MATLAB. We conclude the paper in Section 6.

2 WRITING OUR LP PROBLEM

An LP feasibility problem can be characterized using a matrix $A \in \mathbb{R}^{r \times c}$, a vector $\mathbf{b} \in \mathbb{R}^r$, and an `enum` vector

[a] https://orcid.org/0000-0001-6747-775X

[b] https://orcid.org/0000-0003-0073-2765

[c] https://orcid.org/0009-0004-3145-1119

[d] https://orcid.org/0000-0003-1273-109X

[e] Submitted 06/20/2024. This work was supported in part by the Icelandic Research Fund under Grant 228725-051.

s, $s_i \in$ {LE,GE,EQ}, having r elements. A feasible solution to the LP problem is a vector $\mathbf{x} \in \mathbb{R}^c$ that satisfies

$$[A\mathbf{x}]_i \ [s_i] \ b_i, \quad \text{for all } i = 1,2,\ldots,r,$$

where $[A\mathbf{x}]_i$ is the i-th component of the vector $A\mathbf{x}$. This means that

$$\begin{aligned} [A\mathbf{x}]_i &\le b_i, \ \text{if } s_i \text{ is LE}, \\ [A\mathbf{x}]_i &= b_i, \ \text{if } s_i \text{ is EQ}, \\ \text{and} \quad [A\mathbf{x}]_i &\ge b_i, \ \text{if } s_i \text{ is GE}. \end{aligned}$$

Since the matrix A is usually sparse, we implement it using two vectors, rn and cn, of row and column indices, respectively, and one vector of values val. The vector **b** is implemented by the vector b and the vector of symbols **s** by the enum TypeCon {LE,GE,EQ} vector con.

The vectors rn, cn and val will all have the same number of elements, namely the number of non-zero elements of A, and to set the (i,j)-th element of A to some non-zero value x, i.e. $a_{i,j} :=$x, we write

```
rn[k]=i; cn[k]=j; val[k]=x;
```

A constraint of the LP problem is written in a row of the matrix A. A column of A corresponds to a variable. More exactly, the columns of A contain the coefficients with which the corresponing variable in the vector Variables appears in the constrains of the LP problem. The Variables vector is described below.

2.1 Storing the Variables

To implement a variable of the LP problem we use an Armadillo (Sanderson, 2010) integer vector ivec. For example, the variable $\varphi^{\mathrm{v}}_{k,\ell}$ is ivec {'P',nu,k,l}, where nu, k and l are some numbers of type bint (big integer, i.e. long long). All the variables of the problem are stored together in the sorted vector Variables.

In order to sort the Variables vector we need to define an order relation. We first let the variable's length dictate the relation. If the variables have the same length we compare the first non-equal elements or last elements, which ever comes first. This is implemented with the following binary function:

```
bool varCmp(const ivec &x, const ...
    ivec &y) {
  if (x.size() < y.size()) {
    return true;
  }
  else if (x.size() > y.size()) {
    return false;
  }
  int i, len = x.size();
  for(i=0; i < len-1 && x(i) == ...
    y(i); i++) {};
  return x(i) < y(i);
}
```

We also implement a function, VarID, that obtains the index of a variable in Variables. That is, if var=Variables(i) then VarID(var) returns i. If the variable is not in Variables we return the impossible value -1. The function is as follows:

```
bint VarID(const ivec &v){
  auto found = ...
    equal_range(Variables.begin(), ...
    Variables.end(), v, varCmp);
  if( found.first == found.second ){
    return -1;
  }
  else{
    return found.first - ...
    Variables.begin();
  }
}
```

2.2 Construction of the Matrix A

Since each row of the matrix A corresponds to a constraint, it is preferable to write this matrix in a row-by-row manner. We implement the function, new_aij, that writes a new nonzero element of A or modifies a nonzero value, and a function, close_constr, that "closes" the constraint, such that the next call to new_aij will write an element in the next row. We store the current index of rn, cn and val in the bint variable Index, and the current row number in the int variable ConstrNr. The corresponding functions are as follows:

```
void new_aij(const ivec ...
    &variable, double value){
  bint VID = VarID(variable);
  assert(VID !=-1);
  for (bint i = Index - 1; i >= 0 ...
    && rn[i] == ConstrNr; i--) {
    if (cn[i] == VID) {
      val[i] += value;
      return;
    }
    rn.push_back(ConstrNr);
    cn.push_back(VID);
    val.push_back(value);
    Index++;
  }
}
```

```
void close_constr(TypeCon Type, ...
    double _b){
  con.push_back(Type);
  b.push_back(_b);
  ConstrNr++;
}
```

2.3 Minimal LP Example

As a minimal example for our format, consider the constraints

$$\begin{aligned} 3x - y &\leq 5 \\ x + 2y &= 7. \end{aligned} \tag{1}$$

We need the following variables:

```
using bint = long long;
using namespace std;
using namespace arma;
bint Index;
int ConstrNr;
vector<ivec> Variables
vector<bint> rn, cn;
vector<double> val, b;
enum TypeCon {LE, EQ, GE};
vector<TypeCon> con;
```

The constraints can be implemented as:

```
vector<ivec> Variables={ ...
    ivec{'x'}, ivec{'y'}};
rn[0]=0; cn[0]=0; val[0]=3;
rn[1]=0; cn[1]=1; val[1]=-1;
con[0]=LE; b[0]=5;
rn[2]=1; cn[2]=0; val[2]=1;
rn[3]=1; cn[3]=1; val[3]=2;
con[1]=EQ; b[1]=7;
```

or alternatively with `new_aij` and `close_constr` as:

```
vector<ivec> Variables={ ...
    ivec{'x'}, ivec{'y'}};
new_aij(ivec {'x'},3);
new_aij(ivec {'y'},-1);
close_constr(LE,5);
new_aij(ivec {'x'},1);
new_aij(ivec {'y'},2);
close_constr(EQ,7);
```

Note that `'x'` and `'y'` are interpreted as their ASCII `dec` equivalents, that is 120 and 121 respectively.

The purpose of the structure discussed above is to act as a parser between the problem formulation and the linear solver we use to solve the LP problems, Gurobi (Gurobi Optimization, LLC, 2023). The same example using the Gurobi interface for C would be given by

```
int vars    = 2;
int constrs = 2;
size_t vbeg[]  = {0, 2};
int vlen[]  = {2, 2};
int vind[]  = {0, 1, 0, 1};
double vval[] = {3.0, 1.0, -1.0, ...
    2.0};
char sense[] = ...
    {GRB_LESS_EQUAL,GRB_EQUAL};
double rhs[]   = {5.0, 7.0};
error = GRBXloadmodel(env, ...
    &model, "example", vars, ...
    constrs, 1, 0.0, NULL, sense, ...
    rhs, vbeg, vlen, vind, vval, ...
    NULL, NULL, NULL, NULL, NULL);
```

The syntax of this interface has two main drawbacks; it is rather opaque and one can easily overwrite a previously declared nonzero element of the matrix A accidentally. If, for example, we obtained the $3x$ in (1) via a sum $(x+2x)$ and wrote

```
int vlen[]={3, 2};
int vind[]={0, 0, 1, 0, 1};
double vval[]={1.0, 2.0, 1.0, ...
    -1.0, 2.0};
```

we would overwrite the first instance $1.0 \cdot x$ with $2.0 \cdot x$ rather than adding them together. A parser is easily implemented:

```
// var_size = Variables.size()
// rncnval_size = rn.size()
for(int v=0,j=0;v < var_size;v++){
  vbeg[v]=j;
  for(int k=0;k < rncnval_size;k++){
    if(cn[k] == v){
      vval[j]=val[k];
      vind[j]=rn[k];
      j++;
    }
  }
}
```

3 CPQ FUNCTIONS

In order to parameterize a CPQ function, we use a triangulation of the domain of the function. We will first discuss the specifics of a suitable triangulation $\mathcal{T}$. Then, a prelude on the parameterization of continuous and piecewise affine (CPA) functions will be given, from which the parameterization of CPQ functions

using the triangulation $\mathcal{T}$ will follow naturally. Note that our parameterization of CPQ Lyapunov functions largely follows (Johansson, 1999).

3.1 Triangulation

A triangulation $\mathcal{T}$ with vertices $\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_p\} \subset \mathbb{R}^n$ is a subdivision of a subset of $\mathbb{R}^n$ into simplices $\mathfrak{S}$. A simplex $\mathfrak{S}_\nu$ is defined as

$$\begin{aligned}\mathfrak{S}_\nu &:= \operatorname{co}\{\mathbf{x}_0^\nu, \mathbf{x}_1^\nu, \ldots, \mathbf{x}_n^\nu\} \\ &= \left\{ \mathbf{x} \in \mathbb{R}^n : \mathbf{x} = \sum_{i=0}^{n} \lambda_i \mathbf{x}_i^\nu, \lambda_i \geq 0, \sum_{i=0}^{n} \lambda_i = 1 \right\},\end{aligned}$$

where $\mathbf{x}_i^\nu = \mathbf{v}_{g_\nu(i)}$ for $i = 0, 1, \ldots, n$ and $g_\nu : \{0, 1, \ldots, n\} \to \{1, 2, \ldots, p\}$ is an index-function. For our purposes, we require the triangulation $\mathcal{T}$ to be shape-regular, i.e., every two different simplices $\mathfrak{S}_\nu, \mathfrak{S}_\mu \in \mathcal{T}$ either intersect in a common lower-dimensional face or do not intersect at all. Further, we demand that $\mathbf{x}_0^\nu = \mathbf{0}$ and that the vectors $\mathbf{x}_1^\nu, \mathbf{x}_2^\nu, \ldots, \mathbf{x}_n^\nu$ are linearly independent for all $\mathfrak{S}_\nu \in \mathcal{T}$. Finally, the set theoretic union of all $\mathfrak{S}_\nu \in \mathcal{T}$, denoted $\mathcal{D}_\mathcal{T}$, must be a neighbourhood of the origin of $\mathbb{R}^n$.

An efficient implementation of a triangulation that satisfies these requirements is the triangular fan $(\mathcal{T}_{K,\text{fan}}^{\text{std}})^{\mathbf{F}}$ discussed in (Hafstein, 2019), from here on denoted simply by $\mathcal{T}_K$, where a formula for all $\mathbf{x}_i^\nu$ is given. The vertices $\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_p\}$ of the triangulation $\mathcal{T}_K$ are

$$\{\mathbf{0}\} \cup \left\{ \frac{K}{\|\mathbf{z}\|_2} \mathbf{z} : \mathbf{z} \in \mathbb{Z}^n, \|\mathbf{z}\|_\infty = K \right\},$$

where the scaling parameter $K \in \mathbb{N}$ determines the fineness of the triangulation around zero. The number of simplices in the triangulation $\mathcal{T}_K$ is given by the formula $2^n \cdot K^{n-1} \cdot n!$. The triangulation $\mathcal{T}_1$ in $\mathbb{R}^2$ is depicted in Figure 1.

3.2 Prelude on CPA Functions

To parameterize CPA functions on the whole space $\mathbb{R}^n$, we associate a cone $\mathfrak{C}_\nu$ to each simplex $\mathfrak{S}_\nu \in \mathcal{T}$, defined as

$$\begin{aligned}\mathfrak{C}_\nu &:= \operatorname{cone}\{\mathbf{x}_1^\nu, \mathbf{x}_2^\nu, \ldots, \mathbf{x}_n^\nu\} \\ &= \left\{ \mathbf{x} \in \mathbb{R}^n : \mathbf{x} = \sum_{i=1}^{n} \lambda_i \mathbf{x}_i^\nu, \lambda_i \geq 0 \right\}.\end{aligned}$$

We refer to the vector $\boldsymbol{\lambda} = (\lambda_i)_{i=1,2,\ldots,n} \in \mathbb{R}_+^n$ as the *cone coordinates* of $\mathbf{x}$ in $\mathfrak{C}_\nu$. Additionally, we define the matrix $X_\nu := [\mathbf{x}_1^\nu, \ldots, \mathbf{x}_n^\nu] \in \mathbb{R}^{n \times n}$. Note that for this definition we must assume that the order of the vertices of each $\mathfrak{S}_\nu$ is fixed.

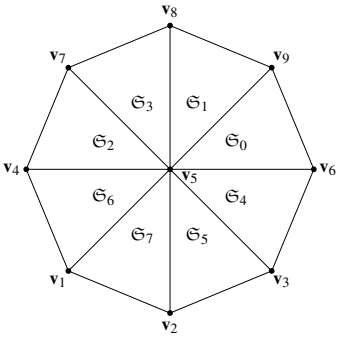


Figure 1: Triangulation $\mathcal{T}_1$ with simplices and vertices indexed

For a CPA function $W : \mathbb{R}^n \to \mathbb{R}$ defined using the triangulation $\mathcal{T}$ we have for every $\mathbf{x} \in \mathfrak{C}_\nu$ that $W(\mathbf{x}) = \mathbf{w}_\nu^T \mathbf{x}$, where

$$\mathbf{w}_\nu^T = \left[W(\mathbf{x}_1^\nu), W(\mathbf{x}_2^\nu), \ldots, W(\mathbf{x}_n^\nu) \right] X_\nu^{-1}.$$

Just note that for all $\mathbf{x} \in \mathfrak{C}_\nu$ we have $\mathbf{x} = X_\nu \boldsymbol{\lambda}$ and thus

$$\begin{aligned}W(\mathbf{x}) &= \mathbf{w}_\nu^T \mathbf{x} \\ &= \left[W(\mathbf{x}_1^\nu), W(\mathbf{x}_2^\nu), \ldots, W(\mathbf{x}_n^\nu) \right] X_\nu^{-1} X_\nu \boldsymbol{\lambda} \\ &= \sum_{i=1}^{n} \lambda_i W(\mathbf{x}_i^\nu).\end{aligned} \tag{2}$$

To implement the computation of W using LP, it is advantageous to define the vector of variables $\Phi \in \mathbb{R}^p$, where $\varphi_j = W(\mathbf{v}_j)$, $j = 1, 2, \ldots, p$. Then $W(\mathbf{x}_i^\nu) = \varphi_{g_\nu(i)}$ for every vertex $\mathbf{x}_i^\nu = \mathbf{v}_j$, $i = 0, 1, \ldots, n$, of some $\mathfrak{S}_\nu \in \mathcal{T}$. Next, we define the vector

$$\Psi^\nu := \left(\varphi_{g_\nu(1)}, \varphi_{g_\nu(2)}, \ldots, \varphi_{g_\nu(n)} \right)^T$$

for every $\mathfrak{S}_\nu \in \mathcal{T}$. Then

$$W(\mathbf{x}) = (\Psi^\nu)^T \boldsymbol{\lambda}$$

by (2), where $\boldsymbol{\lambda} \in \mathbb{R}_+^n$ are the cone coordinates of $\mathbf{x} \in \mathfrak{C}_\nu$.

That W is well-defined and continuous now follows easily from the fact that the vertices $\mathbf{x}_1^\nu, \ldots, \mathbf{x}_n^\nu$ of every $\mathfrak{S}_\nu \in \mathcal{T}$ are linearly independent and that $\mathcal{T}$ is shape-regular: Since $\mathcal{T}$ is shape-regular, $\mathfrak{C}_\nu \cap \mathfrak{C}_\mu = \operatorname{cone}\{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_j\}$, where $j < n$ and the $\mathbf{y}_i$ are the common nonzero vertices of $\mathfrak{S}_\nu$ and $\mathfrak{S}_\mu$, i.e., $\mathbf{y}_i = \mathbf{x}_{k_i}^\nu = \mathbf{x}_{\ell_i}^\mu$ for some $k_i, \ell_i \in \{1, \ldots, n\}$ and $g_\nu(k_i) = g_\mu(\ell_i)$ for all $i = 1, \ldots, j$. Then, for all $\mathbf{x} \in \mathfrak{C}_\nu \cap \mathfrak{C}_\mu$

$$\mathbf{x} = \sum_{i=1}^{j} \lambda_i \mathbf{y}_i = \sum_{i=1}^{j} \lambda_i \mathbf{x}_{k_i}^\nu = \sum_{i=1}^{j} \lambda_i \mathbf{x}_{\ell_i}^\mu,$$

for some unique $\boldsymbol{\lambda}$, because of the linear independence of the vertices. Hence

$$(\Psi^{\nu})^T\boldsymbol{\lambda} = \sum_{i=1}^{j} \lambda_i \varphi_{g_{\nu}(k_i)} = \sum_{i=1}^{j} \lambda_i \varphi_{g_{\mu}(\ell_i)} = (\Psi^{\mu})^T\boldsymbol{\lambda}$$

and W is well-defined on $\mathfrak{C}_{\nu} \cap \mathfrak{C}_{\mu}$ and continuous, since it is the restriction of the continuous functions $\mathbf{x} \mapsto \mathbf{w}_{\nu}^T\mathbf{x}$ and $\mathbf{x} \mapsto \mathbf{w}_{\mu}^T\mathbf{x}$ to $\mathfrak{C}_{\nu} \cap \mathfrak{C}_{\mu}$. Hence, the vector $\boldsymbol{\varphi} \in \mathbb{R}^p$ and the functions g_{ν} connect the different formulas $W(\mathbf{x}) = \mathbf{w}_{\nu}^T\mathbf{x}$, $\mathfrak{S}_{\nu} \in \mathcal{T}$, such that the resulting function

$$W(\mathbf{x}) = \mathbf{w}_{\nu}^T\mathbf{x} \quad \text{if } \mathbf{x} \in \mathfrak{C}_{\nu}$$

is well-defined and continuous.

3.3 Representation of CPQ Functions

For representing a CPQ function $V\colon \mathbb{R}^n \to \mathbb{R}$, i.e., V is continuous and $V(\mathbf{x}) = \mathbf{x}^T P^{\nu}\mathbf{x}$ for a symmetric matrix $P^{\nu} \in \mathbb{R}^{n\times n}$ on each $\mathfrak{S}_{\nu} \in \mathcal{T}$, we can proceed similarly as in the CPA case. However, now we need a $p \times p$ matrix $\Phi \in \mathbb{R}^{p\times p}$ of variables, such that for a simplex $\mathfrak{S}_{\nu} \in \mathcal{T}$ we have

$$P^{\nu} = (X_{\nu}^{-1})^T \Psi^{\nu} X_{\nu}^{-1},$$

where the (k,ℓ)-th entry of the matrix $\Psi^{\nu} \in \mathbb{R}^{n\times n}$ is equal to $\varphi_{g_{\nu}(k),g_{\nu}(\ell)}$, which is the $(g_{\nu}(k),g_{\nu}(\ell))$-th entry of Φ.

Then, for an $\mathbf{x} = \sum_{i=1}^{n} \lambda_i \mathbf{x}_i^{\nu} \in \mathfrak{C}_{\nu}$, i.e., $\mathbf{x} = X^{\nu}\boldsymbol{\lambda}$ with $\boldsymbol{\lambda} \in \mathbb{R}_+^n$, we have

$$\begin{aligned} V(\mathbf{x}) &= \mathbf{x}^T P^{\nu}\mathbf{x} \\ &= \boldsymbol{\lambda}^T X_{\nu}^T (X_{\nu}^{-1})^T \Psi^{\nu} X_{\nu}^{-1} X_{\nu}\boldsymbol{\lambda} \\ &= \boldsymbol{\lambda}^T \Psi^{\nu}\boldsymbol{\lambda} \qquad (3) \\ &= \sum_{k,\ell=1}^{n} \lambda_k\lambda_{\ell}\varphi_{g_{\nu}(k),g_{\nu}(\ell)}. \end{aligned}$$

That V is well-defined and continuous follows similarly as for the CPA case: For all $\mathbf{x} \in \mathfrak{C}_{\nu} \cap \mathfrak{C}_{\mu}$

$$\mathbf{x} = \sum_{i=1}^{j} \lambda_i \mathbf{y}_i = \sum_{i=1}^{j} \lambda_i \mathbf{x}_{k_i}^{\nu} = \sum_{i=1}^{j} \lambda_i \mathbf{x}_{\ell_i}^{\mu},$$

for some $k_i, \ell_i \in \{1,\dots,n\}$ and a unique $\boldsymbol{\lambda}$. Since

$$\begin{aligned} \mathbf{x}^T P^{\nu}\mathbf{x} &= \sum_{r=1}^{j} \lambda_r (\mathbf{x}_{k_r}^{\nu})^T (X_{\nu}^{-1})^T \Psi^{\nu} X_{\nu}^{-1} \sum_{s=1}^{j} \lambda_s \mathbf{x}_{k_s}^{\nu} \\ &= \sum_{r=1}^{j} \lambda_r \mathbf{e}_{k_r}^T \Psi^{\nu} \sum_{s=1}^{j} \lambda_s \mathbf{e}_{k_s} \\ &= \sum_{r,s=1}^{j} \lambda_r\lambda_s \varphi_{g_{\nu}(k_r),g_{\nu}(k_s)}, \end{aligned}$$

where $\mathbf{e}_i$ is the i-th unit vecotor of $\mathbb{R}^n$. Likewise, $\mathbf{x}^T P^{\mu}\mathbf{x} = \sum_{r,s=1}^{j} \lambda_r\lambda_s \varphi_{g_{\mu}(\ell_r),g_{\mu}(\ell_s)}$ and it can be concluded that

$$\mathbf{x}^T P^{\nu}\mathbf{x} = \mathbf{x}^T P^{\mu}\mathbf{x}$$

because $g_{\nu}(k_i) = g_{\mu}(\ell_i)$ for $i = 1,2,\dots,j$. Thus, V is well-defined and continuous.

Remark 1. Note that $g_{\nu}(0)$, i.e. the index of the zero vertex $\mathbf{v}_k = \mathbf{0}$, is never needed in the formulas above.

4 COMPUTING A CPQ LYAPUNOV FUNCTION

Consider a switched linear system with arbitrary switching

$$\dot{\mathbf{x}}(t) = A(t)\mathbf{x}(t), \quad A(t) \in \mathcal{A} \coloneqq \{A_1, A_2, \dots, A_N\},$$

where $A : \mathbb{R}_+ \to \mathcal{A}$ is an arbitrary right-continuous piecewise constant mapping and $N \in \mathbb{N}$ is finite. As discussed in the Introduction, the origin is asymptotically stable for the system, if and only if there exists a Lyapunov function for the system, i.e., a locally Lipschitz function $V\colon \mathbb{R}^n \to \mathbb{R}$ that satisfies

$$\begin{aligned} V(\mathbf{x}) &> 0 && \forall \mathbf{x} \in \mathbb{R}^n\backslash\{\mathbf{0}\}, \qquad (4) \\ V(\mathbf{0}) &= 0, \\ \langle \nabla V(\mathbf{x}), A_i\mathbf{x}\rangle &< 0 && \forall \mathbf{x} \in \mathbb{R}^n\backslash\{\mathbf{0}\}, \\ & && \forall i \in \{1,2,\dots,N\}, \end{aligned}$$

where $\langle \nabla V(\mathbf{x}), A_i\mathbf{x}\rangle$ denotes the inner product of the vectors $\nabla V(\mathbf{x})$ and $A_i\mathbf{x}$. Strictly speaking $\nabla V(\mathbf{x})$ is the Clarke's subdifferential (Clarke, 1990), which in our CPQ case means that for $\mathbf{x} \in \mathfrak{C}_{\nu} \cap \mathfrak{C}_{\mu}$ we need

$$\left\langle \nabla(\mathbf{x}^T P^{\nu}\mathbf{x}), A_i\mathbf{x}\right\rangle < 0 \text{ and } \left\langle \nabla(\mathbf{x}^T P^{\mu}\mathbf{x}), A_i\mathbf{x}\right\rangle < 0.$$

Going into details would go beyond the scope of this paper and we refer the interested reader to the literature (Clarke et al., 1998; Baier et al., 2012).

If V is CPQ and parameterized as in (3), then conditions (4) are equivalent to

$$\begin{aligned} \mathbf{x}^T P^{\nu}\mathbf{x} &> 0 && \forall \mathbf{x} \in \mathfrak{C}_{\nu}\backslash\{\mathbf{0}\} \\ \mathbf{x}^T Q_i^{\nu}\mathbf{x} &< 0 && \forall \mathbf{x} \in \mathfrak{C}_{\nu}\backslash\{\mathbf{0}\} \\ & && \forall i \in \{1,2,\dots,N\} \end{aligned}$$

for all $\mathfrak{S}_{\nu} \in \mathcal{T}$, where $Q_i^{\nu} \coloneqq A_i^T P^{\nu} + P^{\nu} A_i$. We will refer to these conditions as P^{ν} and Q_i^{ν} being positive definite (p.d.) and negative definite (n.d.), respectively, on the cone $\mathfrak{C}_{\nu}$.

4.1 Reexpression of Constraints

Now it is desired to reexpress the condition on Q_i^ν into one that can be verified with LP. Note that for $\mathbf{x} \in \mathfrak{C}_\nu$ we can write $\mathbf{x} = X_\nu \boldsymbol{\lambda}$, where $\boldsymbol{\lambda} \in \mathbb{R}^n_+$ are the cone coordinates of $\mathbf{x}$. Thus we can define a matrix

$$\begin{aligned} B_i^\nu &= X_\nu^T Q_i^\nu X_\nu \\ &= X_\nu^T (A_i^T (X_\nu^{-1})^T \Psi^\nu X_\nu^{-1} + (X_\nu^{-1})^T \Psi^\nu X_\nu^{-1} A_i) X_\nu \\ &= \left(\Psi^\nu X_\nu^{-1} A_i X_\nu\right)^T + \Psi^\nu X_\nu^{-1} A_i X_\nu \\ &= (\Psi^\nu \hat{A}_i^\nu)^T + \Psi^\nu \hat{A}_i^\nu, \end{aligned}$$

where $\hat{A}_i^\nu = X_\nu^{-1} A_i X_\nu$. Then we define a matrix C_i^ν such that $c_{k,k}^{\nu,i} = b_{k,k}^{\nu,i}$ and $c_{k,\ell}^{\nu,i} = \max\{0, b_{k,\ell}^{\nu,i}\}$ for all $k,\ell \in \{1,2,\ldots,n\}$, where $b_{k,\ell}^{\nu,i}$ and $c_{k,\ell}^{\nu,i}$ are the (k,ℓ)-th entries of B_i^ν and C_i^ν, respectively.

Consider the next proposition.

Proposition 1. *If the sum $\sum_{\ell=1}^n c_{k,\ell}^{\nu,i} < 0$ for all $k \in \{1,2,\ldots,n\}$, then Q_i^ν is n.d. on the cone $\mathfrak{C}_\nu$.*

Sketch of proof. We will provide a sketch of the proof here; a detailed proof will be presented in an upcoming paper (Palacios Roman et al., 2024).

Because the off-diagonal elements of C_i^ν are positive, the conditions $\sum_{\ell=1}^n c_{k,\ell}^{\nu,i} < 0$ force the matrix C_i^ν to be a diagonally dominant and n.d. matrix. Because of how C_i^ν is defined from B_i^ν, $\boldsymbol{\lambda}^T C_i^\nu \boldsymbol{\lambda} \geq \boldsymbol{\lambda}^T B_i^\nu \boldsymbol{\lambda}$ for all $\boldsymbol{\lambda} \in \mathbb{R}^n_+$. Thus, B_i^ν is n.d. on the cone $\mathbb{R}^n_+$, from which it follows that Q_i^ν is n.d. on $\mathfrak{C}_\nu$. □

Finally, we can implement the max function in $c_{k,\ell}^{\nu,i} = \max\{0, b_{k,\ell}^{\nu,i}\}$ with the constraints $c_{k,\ell}^{\nu,i} \geq 0$ and $c_{k,\ell}^{\nu,i} \geq b_{k,\ell}^{\nu,i}$. Then, the conditions on C_i^ν become

$$\begin{aligned} c_{k,\ell}^{\nu,i} &\geq \sum_{r=1}^n \left(\psi_{\ell,r} \hat{a}_{r,k}^{\nu,i} + \psi_{k,r} \hat{a}_{r,\ell}^{\nu,i}\right) \\ c_{k,\ell}^{\nu,i} &\geq 0 \end{aligned}$$

for all $k,\ell \in \{1,2,\ldots n\}$ and $k \neq \ell$ and

$$2 \sum_{r=1}^n \psi_{k,r} \hat{a}_{r,k}^{\nu,i} + \sum_{\substack{\ell=1 \\ \ell \neq k}}^n c_{k,\ell} < 0.$$

for all $k \in \{1,2,\ldots,n\}$.

4.2 Constraints of the LP problem

Equipped with the above results, the final constraints for the LP problem can now be summarized. On every simplex $\mathfrak{S}_\nu := \mathrm{co}\{\mathbf{0}, \mathbf{x}_1^\nu, \mathbf{x}_2^\nu, \ldots, \mathbf{x}_n^\nu\}$, we have $X_\nu = [\mathbf{x}_1^\nu, \mathbf{x}_2^\nu, \ldots, \mathbf{x}_n^\nu]$ and index-function g_ν such that $\mathbf{x}_i^\nu = \mathbf{v}_{g_\nu(i)}$. The constraints are then given by:

1. $V(\mathbf{v}_i) > 0$ for all $\mathbf{v}_i \in \mathcal{T}$.
2. For each system define $\hat{A}_i^\nu := X_\nu^{-1} A_i X_\nu$ and C_i^ν such that

$$c_{k,k}^{\nu,i} = 2 \sum_{r=1}^n \varphi_{g_\nu(r),g_\nu(k)} \hat{a}_{r,k}^{\nu,i}, \tag{5}$$

for all $k \in \{1,\ldots,n\}$ and

$$\begin{aligned} c_{k,\ell}^{\nu,i} &\geq \sum_{r=1}^n \left(\varphi_{g_\nu(\ell),g_\nu(r)} \hat{a}_{r,k}^{\nu,i} + \varphi_{g_\nu(k),g_\nu(r)} \hat{a}_{r,\ell}^{\nu,i}\right), \\ c_{k,\ell}^{\nu,i} &\geq 0 \end{aligned} \tag{6}$$

for all $k,\ell \in \{1,\ldots,n\}$ and $k \neq \ell$. Then require for each $\mathfrak{S}_\nu \in \mathcal{T}$ that

$$\sum_{\ell=1}^n c_{k,\ell}^{\nu,i} < 0 \quad \forall k \in \{1,2,\ldots,n\}.$$

The problem variables are the elements of the matrices Φ and C_i^ν.

Remark 2. Note that $V(\mathbf{v}_i) > 0$ for all nonzero vertices $\mathbf{v}_i$ of $\mathcal{T}$ does not ensure that $V(\mathbf{x}) > 0$ for all $\mathbf{x} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$. This can however be checked a posteriori to solving the LP problem by checking whether there exists a solution to the linear matrix inequality (LMI)

$$P^\nu - (X_\nu^{-1})^T U_\nu X_\nu^{-1} \succ 0,$$

where $U_\nu \in \mathbb{R}^{n \times n}_+$ is the problem variable and $A \succ B$ means that $A - B$ is a symmetric and positive definite matrix. This a posteriori check was proposed in (Johansson, 1999, Proposition 4.4).

5 IMPLEMENTATION AND COMPARISON

In our implementation the object corresponding to the simplicial complex $\mathcal{T}_K$ provides the functions `VertexNr(simNo,VertNo)`, to obtain the vertex number analogous to the indexing function $g_\nu(i)$, and `Vertex(simNo,VertNo)` to obtain the vertex vector analogous to $\mathbf{x}_i^\nu$ (see Figure 2). With these functions we can define our variables and construct our constraints discussed in Section 4.2.

Note that since Φ is a sparse symmetric matrix, we only need to define variables for the upper triangular half. Further, we only need $\varphi_{k,\ell}$ if $\mathbf{v}_k$ and $\mathbf{v}_\ell$ are non-zero vertices of the same simplex and then we define `var = ivec {'P', min(k,l), max(k,l)}`. For example for the simplex in Figure 2 we only need $\varphi_{5,8}$, $\varphi_{5,5}$ and $\varphi_{8,8}$. To avoid unnecessary checks and multiple definitions of variables, we first define

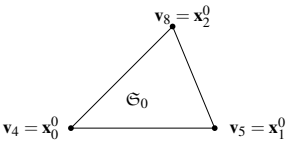


Figure 2: Simplex $\mathfrak{S}_0$ with its vertices indexed. For example, with `simNo=0` and `vertNo=1` the functions `VertexNr(simNo,VertNo)` and `Vertex(simNo,VertNo)` return the `int` 5 and Armadillo vector $[1,0]^T$ respectively.

the variables above the diagonal in this way, and subsequently the diagonal $\varphi_{k,k}$, $k = 1,\ldots,p$. Note that where the index for the zero vertex can be skipped, as it is not used in the constraints.

Furthermore, each C_i^{v} is a symmetric $n \times n$ matrix so we only need the upper triangular half of the elements. In fact, we only need the elements above the diagonal since the diagonal is equal to B_i^{v}, so rather than using $c_{k,k}^{\mathrm{v},i}$ in the constraint we can directly use the right hand side of (5).

In this way we can define each variable, push into the `Variables` vector and then sort it.

All that is left to complete the setup of the LP problem is defining the constraints. This is were the benefit of our structure defined in Section 2 is most apparent since there is no need to keep track of whether a variable appears twice in a sum, as in (6), or not. We can simply go through each constraint using `new_aij` to add the coefficient of each summand and `close_constr` before moving to the next constraint.

As an alternative, we could use a parser such as YALMIP (Löfberg, 2004) in MATLAB (The MathWorks Inc., 2023) to solve for Φ. This would admittedly entail a less convoluted set-up process since we could create matrix variables and don't need to implement the max function with an intermediate variable like C_i^{v}. However, since MATLAB is an interpreted language it doesn't benefit from the efficiency associated with compiled languages, in our case the efficiency of the GNU Compiler `g++`.

For comparison consider the arbitrarily switched linear system discussed in (Andersen et al., 2023; Polanski, 1997; Pyatnitskii, 1971; Brockett, 1966) with subsystems given by

$$A_1 = \begin{bmatrix} 0 & 1 \\ -0.01 & -2 \end{bmatrix} \quad \text{and} \quad A_2 = \begin{bmatrix} 0 & 1 \\ -11.7 & -2 \end{bmatrix}.$$

We can compute a CPQ Lyapunov function with a scaling factor $K \geq 4$ and run the computations twenty times for each K. As can be seen in Figure 3, the average time for each scaling factor using the C++ implementation is roughly $1/200$th of the time needed when using MATLAB. We used the SDPT3 (Toh et al., 1999) solver in the MATLAB

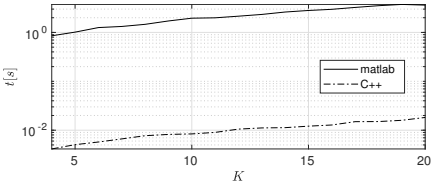


Figure 3: Comparison of the computational time for parameterizing Lyapunov functions with our method in MATLAB and C++; run on i9900K (8 cores) under Linux Mint. Average computation time of twenty tests as a function of triangulation scaling factor K in $\mathcal{T}_K$. Note that the C++ code is ca. 200 times faster than the MATLAB code.

implementation and Gurobi version 11 for the C++ implementation. Additional MATLAB tooling to speed up the computations like the MATLAB Coder can not be utilized to minimize computation time without major changes to YALMIP.

6 CONCLUSIONS

We presented the implementation of a method to compute continuous and piecewise quadratic (CPQ) Lyapunov functions for switched linear systems in C++; such a Lyapunov function is a common Lyapunov function for all the linear subsystems. We compared our implementation to an analogous implementation in MATLAB and showed that the C++ implementation is more than 100 times faster for an example system from the literature. We expect this difference in computational speed to be representative for the general case, as the setup of the linear programming (LP) problem used to parameterize the CPQ Lyapunov function is quite involved and an interpreted computer language like MATLAB is at a great disadvantage in comparison to a compiled language like C++.

REFERENCES

Andersen, S., Giesl, P., and Hafstein, S. (2023). Common Lyapunov functions for switched linear systems: Linear programming-based approach. *IEEE Control Systems Letters*, 7:901–906.

Baier, R., Grüne, L., and Hafstein, S. (2012). Linear programming based Lyapunov function computation for differential inclusions. *Discrete Contin. Dyn. Syst. Ser. B*, 17(1):33–56.

Brockett, R. (1966). The status of stability theory for deterministic systems. *IEEE Trans. Automat. Control*, 11(3):596–606.

Clarke, F. (1990). *Optimization and Nonsmooth Analysis*. Classics in Applied Mathematics. SIAM.

Clarke, F., Ledyaev, Y., and Stern, R. (1998). Asymptotic stability and smooth Lyapunov functions. *J. Differential Equations*, 149:69–114.

Davrazos, G. and Koussoulas, N. (2001). A review of stability results for switched and hybrid systems. In *Proceedings of 9th Mediterranean Conference on Control and Automation*, Dubrovnik, Croatia.

Dayawansa, W. and Martin, C. (1999). A converse Lyapunov theorem for a class of dynamical systems which undergo switching. *IEEE Trans. Automat. Control*, (44):751–760.

Deenen, D., Sharif, B., van den Eijnden, S., Nijmeijer, H., Heemels, M., and Heertjes, M. (2021). Projection-based integrators for improved motion control: Formalization, well-posedness and stability of hybrid integrator-gain systems. *Automatica*, 133:109830.

Goebel, R., Hu, T., and Teel, A. (2006). *Current Trends in Nonlinear Systems and Control. Systems and Control: Foundations & Applications*, chapter Dual Matrix Inequalities in Stability and Performance Analysis of Linear Differential/Difference Inclusions, pages 103–122. Birkhauser.

Gurobi Optimization, LLC (2023). Gurobi Optimizer Reference Manual.

Hafstein, S. (2019). *Simulation and Modeling Methodologies, Technologies and Applications*, volume 873 of *Advances in Intelligent Systems and Computing*, chapter Fast Algorithms for Computing Continuous Piecewise Affine Lyapunov Functions, pages 274–299. Springer.

Hahn, W. (1967). *Stability of Motion*. Springer, Berlin.

Johansson, M. (1999). *Piecewise Linear Control Systems*. PhD thesis: Lund University, Sweden.

Johansson, M. and Rantzer, A. (1998). Computation of piecewise quadratic Lyapunov functions for hybrid systems. *IEEE Trans. Automat. Control*, 43(4):555–559.

Khalil, H. (2002). *Nonlinear Systems*. Pearson, 3. edition.

Liberzon, D. (2003). *Switching in systems and control*. Systems & Control: Foundations & Applications. Birkhäuser.

Löfberg, J. (2004). YALMIP: A toolbox for modeling and optimization in MATLAB. In *In Proceedings of the CACSD Conference*, Taipei, Taiwan.

Mason, P., Boscain, U., and Chitour, Y. (2006). Common polynomial Lyapunov functions for linear switched systems. *SIAM J. Control Optim.*, 45(1):226–245.

Mason, P., Chitour, T., and Sigalotti, M. (2022). On universal classes of Lyapunov functions for linear switched systems. arXiv:2208.09179.

Palacios Roman, J., Hafstein, S., Giesl, P., van den Eijnden, S., Andersen, S., and Heemels, M. (2024). Constructing continuous piecewise quadratic Lyapunov functions with linear programming. [Manuscript in preparation].

Polanski, A. (1997). Lyapunov functions construction by linear programming. *IEEE Trans. Automat. Control*, 42:1113–1116.

Pyatnitskii, Y. (1971). Absolute stability criterion for the 2nd order nonlinear controlled systems with one nonlinear nonstationary element. *Autom. Remote Control*, 32(1):1–11.

Sanderson, C. (2010.). Armadillo: An open source C++ linear algebra library for fast prototyping and computationally intensive experiments. Technical report, NICTA.

Sastry, S. (1999). *Nonlinear Systems: Analysis, Stability, and Control*. Springer.

Shorten, R., Wirth, F., Mason, O., Wulff, K., and King, C. (2007). Stability criteria for switched and hybrid systems. *SIAM Review*, 49(4):545–592.

Sun, Z. and Ge, S. (2011). *Stability Theory of Switched Dynamical Systems*. Communications and Control Engineering. Springer.

The MathWorks Inc. (2023). MATLAB version: 23.2.0 (r2023b).

Toh, K. C., Todd, M. J., and Tütüncü, R. H. (1999). SDPT3 - a matlab software package for semidefinite programming, version 1.3. *Optimization Methods and Software*, 11(1-4):545–581.

van den Eijnden, S., Heemels, M., Nijmeijer, H., and Heertjes, M. (2022). Stability and performance analysis of hybrid integrator–gain systems: A linear matrix inequality approach. *Nonlinear Analysis: Hybrid Systems*, 45:101192.

van den Eijnden, S., Heertjes, M., Heemels, M., and Nijmeijer, H. (2020). Hybrid integrator-gain systems: A remedy for overshoot limitations in linear control? *IEEE Control Systems Letters*, PP:1–1.

Vidyasagar, M. (2002). *Nonlinear System Analysis*. Classics in Applied Mathematics. SIAM, 2. edition.

Paper II

Numerical Computations of Lyapunov Functions for Switched Linear Systems

S. Andersen, E. August, S. Hafstein and J. Piccini

Simulation and Modeling Methodologies, Technologies and Applications, pp. 192–213, 2024.

Reproduced with permission from Springer Nature

We compare different optimisation methods for the computation of Lyapunov functions for switched linear systems. In particular, we compare methods based on linear programming to methods based on semi-definite optimisation. For the linear programming methods, we describe two approaches aimed at reducing the size of the linear programme. This is done by using different kinds of coordinate transforms. One attempts to increase the rotational symmetry of the solution trajectories of the sub-systems. The other one rotates the areas of maximum curvature of possible Lyapunov functions to where the triangulation is closer meshed. We investigate the effect of these coordinate transforms on several examples and compare their efficiency.

Numerical computations of Lyapunov functions for switched linear systems*

Stefania Andersen[1][0000−0001−6747−775X], Elias August[2][0000−0001−9018−5624], Sigurdur Hafstein[1][0000−0003−0073−2765], and Jacopo Piccini[2][0000−0002−4180−8140]

[1] University of Iceland, Faculty of Physical Sciences, Dunhagi 5, 107 Reykjavik, Iceland {saa20,shafstein}@hi.is

[2] Reykjavik University, Department of Engineering, Menntavegur 1, 102 Reykjavik, Iceland
{eliasaugust, jacopop}@ru.is

Abstract. We compare different optimisation methods for the computation of Lyapunov functions for switched linear systems. In particular, we compare methods based on linear programming to methods based on semi-definite optimisation. For the linear programming methods, we describe two approaches aimed at reducing the size of the linear programme. This is done by using different kinds of coordinate transforms. One attempts to increase the rotational symmetry of the solution trajectories of the sub-systems. The other one rotates the areas of maximum curvature of possible Lyapunov functions to where the triangulation is closer meshed. We investigate the effect of these coordinate transforms on several examples and compare their efficiency.

Keywords: Switched Systems, Lyapunov Functions, Semidefinite Programming, Linear Programming, Preconditioning.

1 Introduction

For the understanding of dynamical systems in science and engineering, one often uses Ordinary Differential Equations (ODEs) for modelling, that is, the temporal behaviour of the state variables is quantified through a differential equation of the following form,

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^n, \tag{1}$$

where $\mathbf{x}$ is the so called state vector and the vector valued function $\mathbf{f}$ describes the dynamics of the system.

Switched systems are used for modelling in many different fields [25]. They describe systems, whose dynamics are affected by instantaneous changes, by considering a set of continuous-time sub-systems and a rule governing the switching

* Submitted 11/15/2023. This work was supported in part by the Icelandic Research Fund under Grant 228725-051 and has received funding from European Union's Horizon 2020 research and innovation programme under grant agreement no. 965417.

between them. Examples for their use range from the biological sciences [15], for example, the dynamics of human body thermoregulation during sleep abruptly changes when sleep transitions from non-rapid eye movement (REM) sleep to REM sleep [22], to mechanical engineering, where one example is the dynamics of an engine with shifting gears [40]. Other engineering applications are power electronics, automotive control, robotics, and air traffic control [44], to name a few. Analysing switched systems is also important in the field of hybrid systems [25], where the switching is due to the interaction between the dynamics of individual continuous-time sub-systems and the discrete-time dynamics of the switching [18]. The differential inclusions associated with uncertainty quantification in continuous-time systems is another common source of switched systems [5, 13].

A switched linear system has the following form,

$$\dot{\mathbf{x}} = A_m \mathbf{x}, \quad m \in \mathcal{P}, \quad \mathbf{x} \in \mathbb{R}^n. \tag{2}$$

The switching between the systems $\dot{\mathbf{x}} = A_m \mathbf{x}$ is modelled through a switching signal $\sigma \colon [0, \infty) \to \mathcal{P}$. In this paper, we will only consider arbitrary switching between a finite number of sub-systems; that is, $\mathcal{P}$ is finite and $\sigma \colon [0, \infty) \to \mathcal{P}$ is arbitrary, except for the (technical) assumption that the number of switching times is finite on any finite time-interval.

Parts of the results in this paper were presented at the International Conference on Informatics in Control, Automation and Robotics (ICINCO) in Rom, Italy, 2023, and published in the corresponding conference proceedings [3]. It is organised as follows. In Section 2, we recall results on stability of equilibrium points, switched systems, linear programming (LP), and semidefinite programming (SDP). In sections 3 and 4, we discuss our LP approach and the canonical SDP approach, respectively, to compute Lyapunov functions for switched linear system. In Section 4, we compare the LP and the SDP approaches using numerous examples. In Section 5, we discuss the AngleAnalysis app, used to aid the LP approach, before we conclude the paper in Section 6.

2 Preliminaries

In this section, we recall some results and definitions used in this paper regarding stability of equilibrium points, switched system, LP, and SDP.

2.1 Stability of an Equilibrium Point

When modelling a dynamical system using (1), the trajectory, or solution curve, is given by $t \mapsto \boldsymbol{\phi}(t, \boldsymbol{\xi})$, where $\boldsymbol{\xi}$ is the initial state; that is, $\boldsymbol{\phi}(0, \boldsymbol{\xi}) = \boldsymbol{\xi}$. In the following, we provide the definitions of an equilibrium point for system (1) and of the stability of an equilibrium point. We denote the Euclidian norm of a vector $\mathbf{x} \in \mathbb{R}^n$ by $\|\mathbf{x}\|$.

Definition 1. *A point* $\mathbf{x}_0$ *is an* equilibrium point *for* (1) *if* $\mathbf{f}(\mathbf{x}_0) = \mathbf{0}$. *Without loss of generality, we assume that the equilibrium point of interest is located at*

the origin; otherwise, we can rewrite (1) *in a new set of coordinates given by* $\mathbf{z} = \mathbf{x} - \mathbf{x}_0$.

Definition 2. *The origin is called* asymptotically stable *for* (1) *if*

(i) for every $\varepsilon > 0$ *there exists a* $\delta > 0$*, such that if* $\|\boldsymbol{\xi}\| < \delta$ *then for every* $t \geq 0$ *we have* $\|\boldsymbol{\phi}(t, \boldsymbol{\xi})\| < \varepsilon$ *and*

(ii) there exists a $\delta > 0$*, such that if* $\|\boldsymbol{\xi}\| < \delta$ *then* $\lim_{t\to\infty} \|\boldsymbol{\phi}(t, \boldsymbol{\xi})\| = 0$.

The origin is called exponentially stable *if there exist constants* $\alpha > 0$ *and* $c \geq 1$ *such that*

$$\|\boldsymbol{\phi}(t, \boldsymbol{\xi})\| \leq c\|\boldsymbol{\xi}\| \exp[-t\alpha] \quad \textit{for all } t \geq 0. \tag{3}$$

2.2 Switched Systems

When modelling switching behaviour by switching between distinct $\mathbf{f}_m(\mathbf{x})$, $m \in \mathcal{P}$ where $\mathcal{P}$ is some index set, such that

$$\dot{\mathbf{x}} = \mathbf{f}_{\sigma(t)}(\mathbf{x}), \quad \sigma : [0, \infty[\mapsto \mathcal{P}, \tag{4}$$

we call (4) a switched system with switching signal σ and sub-systems $\mathbf{f}_m$. On first sight, one might be tempted to derive the stability of the equilibrium for (4) by considering its stability for each sub-system. However, as Figure 1 shows, switching can either stabilise two previously unstable systems or destabilise two stable ones. Consequently, we need to consider the shape of the switching signal when evaluating the stability of an equilibrium point of (4). There are several possibilities to adapt the stability concepts from definitions 1 and 2 to switched system (4). However, since we assume that the switching signal is an arbitrary one, besides having a finite number of switching events on a finite time interval, we do not need to include it in the stability analysis. Under this assumption, a necessary condition for the asymptotic stability of an equilibrium point for (4) is that the sub-systems $\dot{\mathbf{x}} = \mathbf{f}_m(\mathbf{x})$ all have the same equilibrium point and that it is asymptotically stable for each one of them. That is, for $\mathbf{x}_0 = \mathbf{0}$, we demand $\mathbf{f}_m(\mathbf{0}) = \mathbf{0}$ for every $m \in \mathcal{P}$ and that the conditions of Definition 2 hold for all $\boldsymbol{\phi}(t, \boldsymbol{\xi}) = \boldsymbol{\phi}_\sigma(t, \boldsymbol{\xi})$, where $t \mapsto \boldsymbol{\phi}_\sigma(t, \boldsymbol{\xi})$ is the solution curve of (4) for a particular admissible switching signal σ.

Note that in science and engineering, one often linearises the system given by (1) and it becomes

$$\dot{\mathbf{x}} = A\mathbf{x}, \quad A := D\mathbf{f}(\mathbf{0}), \quad \mathbf{x} \in \mathbb{R}^n, \tag{5}$$

where $D\mathbf{f}(\mathbf{0}) \in \mathbb{R}^{n\times n}$ denotes the Jacobian-matrix of $\mathbf{f}$ at the origin. According to the Hartman-Grobman theorem [19], the solutions of linearised system (5) have the same qualitative behaviour close to the origin as system (1), whenever the real parts of the eigenvalues of $D\mathbf{f}(\mathbf{0})$ are nonzero. The solution to the linear system is given by the matrix exponential $\exp[At]$; that is, a solution starting at $\boldsymbol{\xi} \in \mathbb{R}^n$ at time zero will be at $\exp[At]\boldsymbol{\xi}$ at time t or $\boldsymbol{\phi}(t, \boldsymbol{\xi}) := \exp[At]\boldsymbol{\xi}$. The

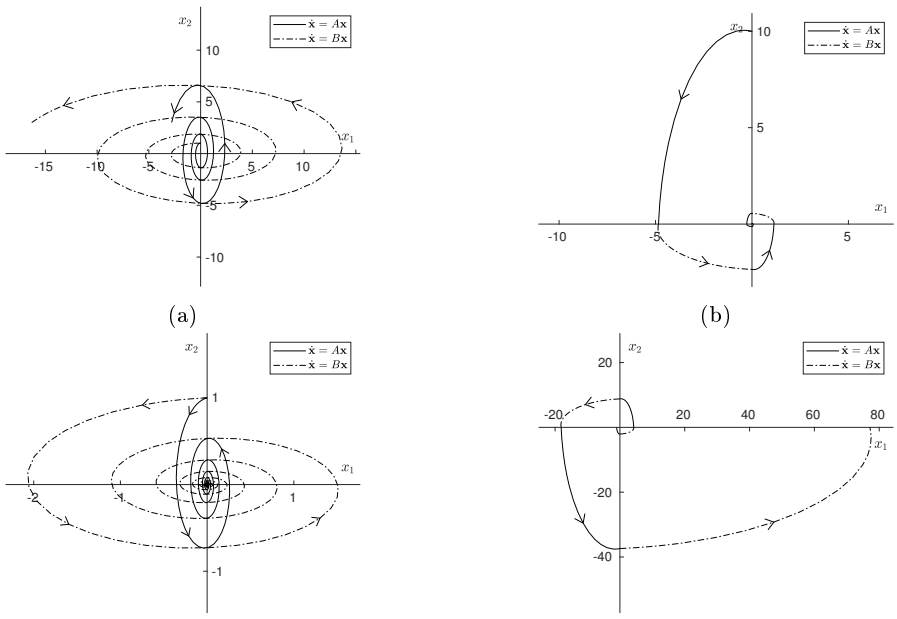


Fig. 1: Trajectories of (a) two unstable systems starting at $\mathbf{x} = [0, 1]^T$ with (b) a stable switching between those two systems starting $\mathbf{x} = [0, 10]^T$ and trajectories of (c) two stable systems starting at $\mathbf{x} = [0, 1]^T$ with (d) an unstable switching between those two systems starting $\mathbf{x} = [-1, 0]^T$. The figure is adapted from [2].

matrix valued function $t \mapsto \exp[At]$ is called the transition matrix of system (5). Note that asymptotic stability of the origin of linear system (5) is equivalent to its exponential stability. Moreover, the origin is asymptotically stable if and only if matrix A is Hurwitz; that is, the real part of all eigenvalues of A is negative.

Now, the transition matrix for (2) with switching signal σ is given by

$$t \mapsto \exp[A_{\sigma(t_k)}(t - t_k)] \cdot \exp[A_{\sigma(t_{k-1})}(t_k - t_{k-1})] \cdot \\ \cdots \exp[A_{\sigma(t_1)}(t_2 - t_1)] \cdot \exp[A_{\sigma(t_0)}(t_1 - t_0)],$$

where $t_0, t_1, t_2, \ldots, t_k$ are the switching times and

$$t > t_k > t_{k-1} > \ldots > t_1 > t_0 = 0.$$

Moreover, for switched system (2) under arbitrary switching, the origin is exponentially stable if and only if for every switching signal σ with a finite number of switching times on every finite interval, there exist constants $\alpha > 0$ and $c \geq 1$,

independently of σ, such that inequality (3) holds for

$$\begin{aligned}\boldsymbol{\phi}_\sigma(t,\boldsymbol{\xi}) := &\exp[A_{\sigma(t_k)}(t-t_k)]\cdot\exp[A_{\sigma(t_{k-1})}(t_k-t_{k-1})]\\ &\cdots\exp[A_{\sigma(t_1)}(t_2-t_1)]\cdot\exp[A_{\sigma(t_0)}(t_1-t_0)]\boldsymbol{\xi}.\end{aligned}$$

Notably, even for planar systems ($n = 2$) and switching between only two different systems ($\mathcal{P} = \{1, 2\}$), unless matrices A_m have some very special structure, for instance, if they commute [1], determining stability is an open problem. To understand why, note that exponential stability of the origin is characterised by matrices A_1 and A_2 and the origin is exponentially stable if and only if all pairwise convex combinations of A_1, A_2, A_1^{-1}, and A_2^{-1} are Hurwitz, a property which is difficult to prove [14, 24].

A sufficient condition for exponential stability of the origin of (2) is the existence of a quadratic common Lyapunov function (QCLF). That is, there exists an $n \times n$ symmetric positive definite matrix P, denoted by $P \succ 0$, such that the following linear matrix inequality (LMI) holds [23, 9]:

$$PA_m + A_m^{\mathrm{T}}P \prec 0, \quad \forall m \in \mathcal{P}. \tag{6}$$

However, even for $n = |\mathcal{P}| = 2$ the origin of an arbitrarily switched system can be exponentially stable, while a QCLF does not exist for the system. For published work on the existence of an common Lyapunov function (CLF), see for example [30, 31, 17, 27, 28]. For a review on different existing methods to compute such a CLF, if it exists, see [16]. One method to compute a CLF is to use LP to parametrise piecewise linear Lyapunov functions [11, 12, 6–8, 45, 37, 38, 33, 34, 46, 4]. Another one is to use SDP to parametrise higher-order polynomial Lyapunov functions [39, 36]. In the next two subsection we give a concise description of LP and SDP as needed in this paper.

2.3 Linear Programming

In LP one seeks to solve the following minimisation problem:

$$\text{minimise} \quad \mathbf{c}^T\mathbf{x} \tag{7}$$

$$\text{subject to} \quad A\mathbf{x} \leq \mathbf{b} \text{ and } \mathbf{x} \geq \mathbf{0}. \tag{8}$$

Here $A \in \mathbb{R}^{n\times m}$, $\mathbf{c} \in \mathbb{R}^m$, and $\mathbf{b} \in \mathbb{R}^n$ are given and one wants to find a feasible vector $\mathbf{x} \in \mathbb{R}^m$ – that is, such that $A\mathbf{x} \leq \mathbf{b}$ and $\mathbf{x} \geq \mathbf{0}$ holds, where the inequalities are to be understood component-wise – that minimises $\mathbf{c}^T\mathbf{x}$. Typically m is much larger than n. Note that there are many other equivalent formulations of an LP problem. There exist very efficient methods to solve LP problems and mature solvers are available. In this work, we use Gurobi [20], which is a commercial solver but can be used free of charge within academia.

2.4 Semidefinite Programming

In SDP, we replace the nonnegative orthant constraint of LP by the cone of positive semidefinite matrices and pose the following minimisation problem [43]:

$$\begin{aligned} \text{minimise} \quad & \mathbf{c}^T\mathbf{x} \\ \text{subject to} \quad & F(\mathbf{x}) \succeq 0, \text{ where} \\ & F(\mathbf{x}) = F_0 + \sum_{i=1}^{n} x_i F_i. \end{aligned} \tag{9}$$

Here, $\mathbf{x} \in \mathbb{R}^n$ is the free variable. The so-called problem data, which are given a priori, are vector $\mathbf{c} \in \mathbb{R}^n$ and symmetric matrices $F_j \in \mathbb{R}^{m\times m}$, $j = 0, 1, \ldots, n$. The *dual* problem associated with the semidefinite program given by (9) is [43]:

$$\begin{aligned} \text{maximise} \quad & -\text{trace}(F_0 Z) \\ \text{subject to} \quad & \text{trace}(F_i Z) = c_i, \ i = 1, 2, \ldots, n \\ & Z \succeq 0. \end{aligned} \tag{10}$$

Here, the free variable is the symmetric matrix $Z \in \mathbb{R}^{m\times m}$. Note that solutions of primal problem (9) provides upper bounds on solutions of the dual and vice versa. This is called *weak duality* and holds since

$$\begin{aligned} \mathbf{c}^T\mathbf{x} + \text{trace}(F_0 Z) &= \sum_{i=1}^{n} c_i x_i + \text{trace}(F_0 Z) = \sum_{i=1}^{n} \text{trace}(F_i Z) x_i + \text{trace}(F_0 Z) \\ &= \text{trace}\left(\left[\sum_{i=1}^{n} F_i x_i + F_0\right] Z\right) \geq 0. \end{aligned}$$

The last inequality holds true because of self-duality of the positive semidefinite cone [35]. If the inequality holds strictly then we speak of *strong duality*.

Programmes of this type can be solved efficiently using interior-point methods. The interested reader is referred to reference [43] and the excellent textbook by the same authors [10]. Moreover, convexity of the set of symmetric positive semidefinite matrices in (9) implies that the minimisation problem has a global minimum, if it is feasible.

3 Lyapunov functions by Linear Programming

In the LP approach, presented in [4], for computing piecewise linear CLF for the switched linear system (2), a neighbourhood of the origin must first be triangulated. Then, an LP is created. Its variables are the values of the piecewise linear CLF to be parametrised for the system and its constraints secure that a feasible solution delivers a function that fulfils the conditions of a CLF. The number of the constraints of the LP problem is given by the number of simplices in the triangulation multiplied with the dimension of the state space. The

Table 1: Some examples of the number of simplices in the triangulation $\mathcal{T}_K$ in different dimensions n; see formula (11)

n	K	Number of simplices in $\mathcal{T}_K$
2	5	40
2	10	80
2	50	400
2	100	800
3	5	1,200
3	10	4,800
3	50	120,000
3	100	480,000
4	5	48,000
4	10	384,000
4	50	48,000,000
4	100	384,000,000
5	5	2,400,000
5	10	38,400,000
5	50	$2.5 \cdot 10^{10}$
5	100	$3.84 \cdot 10^{11}$

vertices of the simplices of the triangulation are first put on the integer grid $\mathbf{z} \in \mathbb{Z}^n$, $\|\mathbf{z}\|_\infty = K$, where $K \in \mathbb{N}$ determines the resolution, or fineness, of the triangulation and $\|\mathbf{z}\|_\infty := \max_{i=1,2,\ldots,n} |z_i|$. The triangulation with resolution parameter K is denoted $\mathcal{T}_K$. We will later map it to the triangulation $\mathcal{T}_K^{\mathbf{F}}$ for computational reasons, but the number of simplices will stay the same.

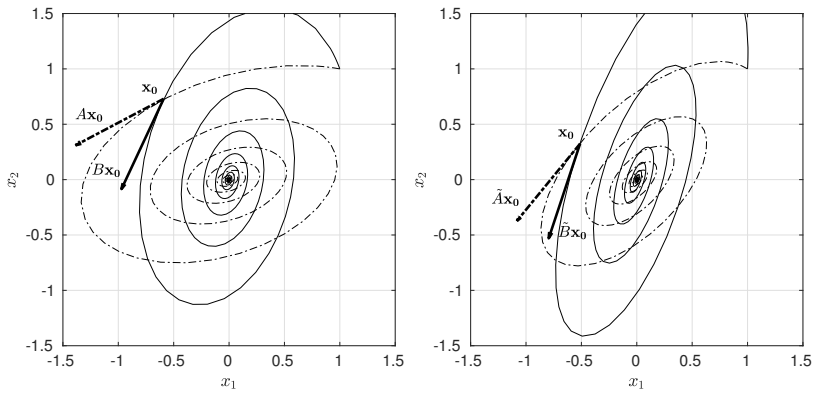
To count the total number of simplices in the triangulation $\mathcal{T}_K$ for a given dimension n, first, note that the triangulation subspace has $2n$ sides. Each side consists of $(2K)^{n-1}$ hypercubes of dimension $n-1$ and each hypercube leads to $(n-1)!$ simplices. To see this, note that the k-dimensional hypercube $[0,1]^k$ is cut into simplices according to $0 \leq x_{p(1)} \leq x_{p(2)} \leq \ldots \leq x_{p(k)} \leq 1$ for every permutation p of $\{1, 2, \ldots, k\}$ and that the $(n-1)$ dimensional hypercubes are cut accordingly. Thus, the number of simplices as a function of dimension n and resolution parameter K is given by: [3]

$$\text{Number of simplices} = 2n(2K)^{n-1}(n-1)! = 2^n K^{n-1} n! \tag{11}$$

We can see that this is a case of the curse of dimensionality, as the number of simplices grows very fast with dimension. For an example of the number of simplices needed, see Table 1.

In [21], it was shown that the numerical approach from [4] is always able to compute a CLF if one exists. However, as shown in Table 1, if the LP approach from [4] is to be applicable in dimensions larger than $n = 3$ or $n = 4$ then some reduction in the number of simplices is clearly needed. A promising preconditioning approach was presented in [4]. It attempts to increase the rotational symmetry of the solutions of the sub-systems (see Figure 2). In more

[3] Note that there is an error in [4] where $2n$ was erroneously replaced by 2^n



(a) Angle is 37.11°. (b) Angle is now 20.97°.

Fig. 2: Exemplary angle between (a) vector fields $\mathbf{x} \mapsto A\mathbf{x}$ and $\mathbf{x} \mapsto B\mathbf{x}$ and (b) the transformed (or preconditioned) vector fields $\mathbf{x} \mapsto \tilde{A}\mathbf{x}$ and $\mathbf{x} \mapsto \tilde{B}\mathbf{x}$ at a point $\mathbf{x}_0$. The figure is adapted from [3].

detail, the idea is to make a coordinate transform such that $V(\mathbf{x}) = \|\mathbf{x}\|$ is closer to fulfilling the conditions of a CLF. To compute such a coordinate transform, we first solve the Lyapunov equation for each sub-system $\dot{\mathbf{x}} = A_m\mathbf{x}$, that is, we solve

$$A_m^T P_m + P_m A_m = -I_n \quad \text{for all } m \in \mathcal{P},$$

where I_n denotes the $n \times n$ identity matrix. Since each individual sub-system $\dot{\mathbf{x}} = A_m\mathbf{x}$ has an exponentially stable equilibrium at the origin, the matrices A_m are Hurwitz and these equations all have a positive definite solution $P_m \succ 0$. Thus, for each sub-system $\dot{\mathbf{x}} = A_m\mathbf{x}$, the function $V_m(\mathbf{x}) = \mathbf{x}^T P_m \mathbf{x}$ is a quadratic Lyapunov function. We then define the matrix

$$R := \sum_{m \in \mathcal{P}} \lambda_m P_m$$

as a convex combination of the P_m, that is, $\lambda_m \geq 0$ for all $m \in \mathcal{P}$ and $\sum_{m \in \mathcal{P}} \lambda_m = 1$. Note that R is also symmetric and positive definite, that is, $R \succ 0$.

Next, we use the coordinate transform $\mathbf{x} \mapsto R^{\frac{1}{2}}\mathbf{x}$; that is, we replace the matrices A_m by $\widetilde{A}_m := R^{\frac{1}{2}} A_m R^{-\frac{1}{2}}$. Recall that for $R \succ 0$ there is exactly one symmetric and positive definite matrix $R^{\frac{1}{2}} \succ 0$ such that $R = R^{\frac{1}{2}} R^{\frac{1}{2}}$. We denote the inverse of $R^{\frac{1}{2}}$ by $R^{-\frac{1}{2}}$ and since $R^{-\frac{1}{2}}$ is symmetric and for $\mathbf{x} \neq \mathbf{0}$ we have

$$\mathbf{x}^T R^{-\frac{1}{2}} \mathbf{x} = (R^{\frac{1}{2}}\mathbf{y})^T R^{-\frac{1}{2}} R^{\frac{1}{2}} \mathbf{y} = \mathbf{y}^T R^{\frac{1}{2}} \mathbf{y} > 0$$

where $\mathbf{y} = R^{-\frac{1}{2}}\mathbf{x} \neq \mathbf{0}$, we also have $R^{-\frac{1}{2}} \succ 0$. We refer to this procedure as *preconditioning* switched system (2). Note that if $|\mathcal{P}| = 1$ then $V(\mathbf{x}) = \mathbf{x}^T I_n \mathbf{x} =$

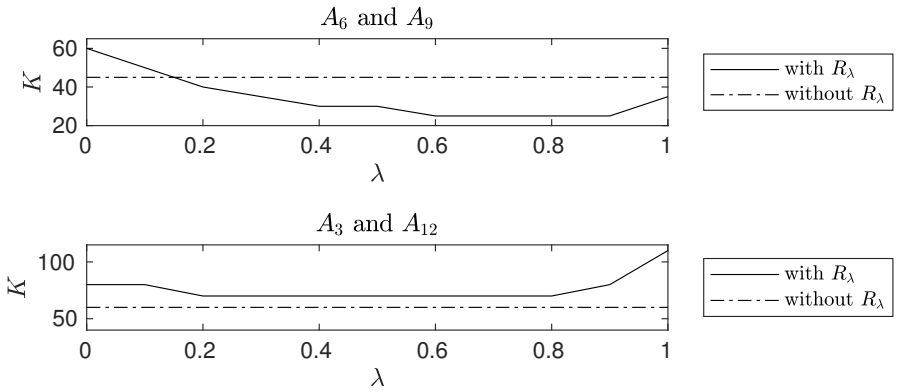


Fig. 3: Lowest resolution factor K that delivers a piecewise linear CLF as a function of λ for two examples of three-dimensional systems. The number of simplices needed is $64K^2$. While $64 \cdot 45^2 = 129,600$ simplices are needed without preconditioning, we get a solution with $64 \cdot 25^2 = 40,000$ simplices for $\lambda \in [0.6, 0.9]$ for $\mathcal{P} = \{6, 9\}$ (upper figure). For $\mathcal{P} = \{3, 12\}$, we have an atypical case, for which preconditioning is counterproductive (lower figure). The first example possesses a QCLF while the second does not. The figure is adapted from [3], where also A_m is defined for $m = 3, 6, 9, 12$.

$\|\mathbf{x}\|^2$ is a Lyapunov function for the transformed system. Furthermore, a Lyapunov function for the transformed systems is easily transformed back using the inverse $\mathbf{x} \mapsto R^{-\frac{1}{2}}\mathbf{x}$ of the coordinate transform $\mathbf{x} \mapsto R^{\frac{1}{2}}\mathbf{x}$. In Figure 2, we show how trajectories of two systems are altered through preconditioning.

In the majority of cases, one can compute a piecewise linear CLF for switched system (2) with fewer simplices when using this preconditioning. That is, one can compute a CLF using a lower value for K in the triangulation $\mathcal{T}_K$ (or $\mathcal{T}_K^{\mathbf{F}}$). However, it is unfortunately not transparent how to choose the optimal value of parameter λ_m. In Figure 3, we investigate this for two cases from the set of three-dimensional linear systems defined in the appendix of [3]. In both cases, we consider the two systems $\dot{\mathbf{x}} = A_r\mathbf{x}$ and $\dot{\mathbf{x}} = A_s\mathbf{x}$, the coordinate transform

$$R_\lambda = \lambda P_r + (1 - \lambda)P_s,$$

which corresponds to $\lambda_r = \lambda$ and $\lambda_s = 1 - \lambda$, and report the lowest resolution parameter K in the triangulation $\mathcal{T}_K$, for which we could compute a piecewise linear CLF for the system. For $r = 6$ and $s = 9$, we see the typical case where preconditioning allows us to compute a CLF using considerably fewer simplices. However, there are also cases, where preconditioning is not advantageous or even counterproductive, as it is for $r = 3$ and $s = 12$. While usually preconditioning has the effect of reducing the number of simplices needed, choosing an optimal value for parameter λ_m is an open problem. Note that this problem is of much practical value for systems of higher dimensions, where the number of simplices

is the limiting factor to for computing a CLF. To better understand the effect of preconditioning, we developed the AngleAnalysis app discussed in Section 5.

3.1 Preconditioning with Rotation

As previously discussed, we first triangulate the hyper-cube $[-K, K]^n$ in the triangulation $\mathcal{T}_K$. For numerical reasons, it is advantageous to map the vertices of $\mathcal{T}_K$ using the mapping

$$\mathbf{F} : \mathbb{R}^n \to \mathbb{R}^n, \quad \mathbf{F}(\mathbf{0}) = \mathbf{0}, \quad \text{and} \quad \mathbf{F}(\mathbf{x}) = \frac{\|\mathbf{x}\|_\infty}{\|\mathbf{x}\|}\mathbf{x} \quad \text{if } \mathbf{x} \neq \mathbf{0}.$$

That is, we map every vertex on the surface of the n-dimensional hyper-cube $[-K, K]^n$ to the surface of the n-dimensional hyper-sphere with radius K.[4] We refer to this new triangulation as $\mathcal{T}_K^{\mathbf{F}}$ to stress the use of mapping $\mathbf{F}$. In this paper, we use this triangulation for all computations. Thus, we often do not make the distinction and rather refer to it as $\mathcal{T}_K$. The process of mapping the vertices is visualised in Figure 4 for $n = 2$ and in Figure 5 for $n = 3$. In Figure 4 and Figure 5, we observe that the density of vertices varies in different regions. As a result, the orientation of the hyper-sphere might be critical with respect to ease of computing a CLF. Intuitively, in two dimensions, if the semi-major axis of the ellipse given by a level set of the CLF lies on the line connecting two opposite corners of the square used for triangulation, shown in the right panel of Figure 4, then the K necessary for successfully computing the CLF should be minimal. As a proof of concept we test the effect of rotating sub-systems A_m such that the resulting CLF is as just mentioned. This is visualised in Figure 6.

In the following, we explain how we compute the rotation. If matrix $P \succ 0$ defines the QCLF for an arbitrarily switched system (and obtained, for example, using the SDP approach), then we consider the eigenvalue decomposition of matrix P given by $P = U\Lambda U^T$, where Λ is diagonal and consists of the eigenvalues of P in ascending order and U is orthogonal. We then map the first column of U to the line $x_1 + x_2 + \ldots + x_n = 0$. Specifically, we apply the QR-decomposition to the vector of ones of length n, set $R = U^T Q$, and replace A_m by $\widetilde{A}_m = RA_mR^T$. Note that since both U and Q are orthogonal matrices, so are R and $R^{-1} = R^T$. Using this coordinate transform results in the area of largest curvature of the ellipsoid given by $\mathbf{x}^T P\mathbf{x} = c$ for a constant $c > 0$, or in other words the cone around the eigenvector corresponding to the smallest eigenvalue of P, to be transferred in the direction $(1, 1, \ldots, 1)$, where the simplices of the triangulation are of maximal density.

We tested this concept on the sets of two-dimensional and three-dimensional systems, for which we computed QCLF in [3]. The results are promising. The highest resolution factor in the two-dimensional case was $K = 5$ so there isn't much room for improvement. However, for only 2% of the cases the rotation was detrimental, for 88% the rotation was inconsequential, and for 10% the rotation was beneficial. In Figure 6, we can see what effect this rotation has on

[4] Note that there is a typo in [4] where the fraction is reversed.

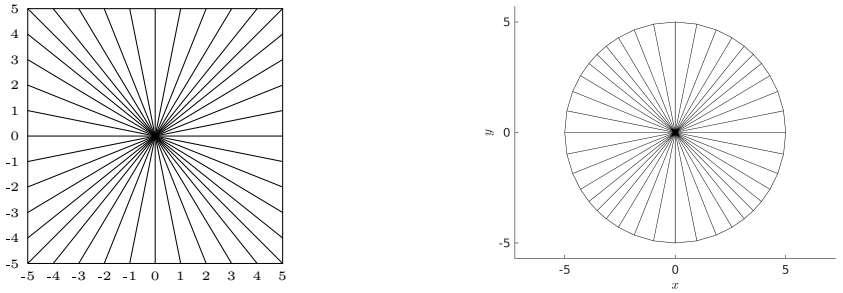


Fig. 4: The figure shows the process of creating a $\mathcal{T}_5$ triangulation in two dimensions. The triangulation of $[-5,5]^2$ is shown in the left panel and the mapping of the square to a circle is shown in the right panel. The figure is adapted from [2].

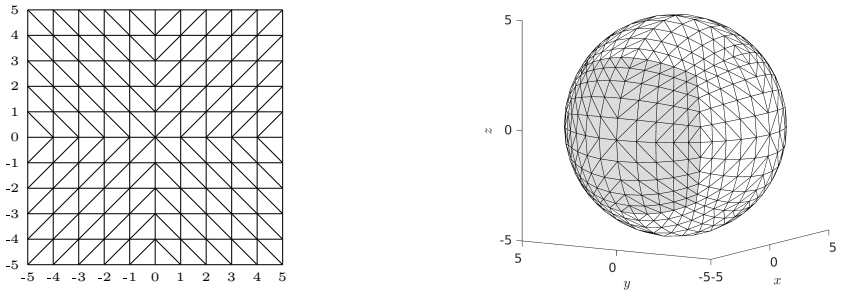


Fig. 5: The figure shows the process of creating a $\mathcal{T}_5$ triangulation in three dimensions. One face of the triangulation of $[-5,5]^3$ is shown in the left panel and the mapping of the cube to a sphere is shown in the right panel, where the projection of the face has been coloured grey. The figure is adapted from [2].

the computed CLF when considering an arbitrarily switched system with the two-dimensional sub-systems $\dot{\mathbf{x}} = A_{14}\mathbf{x}$ and $\dot{\mathbf{x}} = A_{17}\mathbf{x}$ defined in Section 4.

The total number of three dimensional switched systems in [3] with a QCLF was fairly low, but the results still clearly show the benefit of using the rotation. In 18.2% of the cases it was detrimental, in 40.9% it was inconsequential, and in 40.9% the rotation was beneficial and reduced the number of simplices needed to compute a piecewise linear CLF. An overview of the results can be found in Table 2. There is one outlier marked by $\Delta K = -21$ for one combination of three-dimensional systems, which means that we needed a resolution parameter K for the rotated system that was 21 larger than when not using the rotation. It can be explained by the fact that two eigenvalues are much smaller than the third one and we cannot project both of them to a diagonal, where the simplices are denser. In such a case, one can get unlucky when rotating, because, although one area of large curvature is mapped to where simplices are densely distributed, other

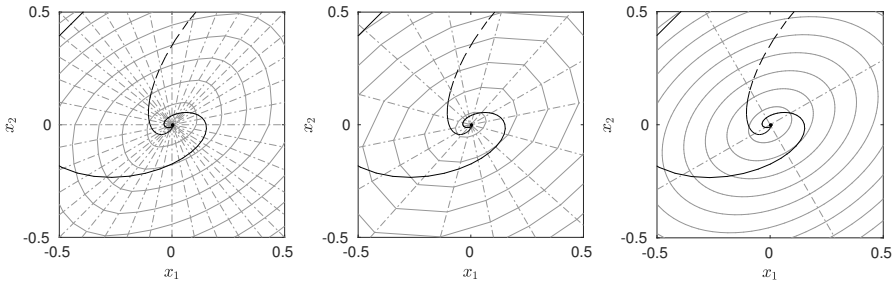


Fig. 6: Trajectories of A_{14} (black and solid), A_{17} (black and dashed), level sets of a CLF (gray and solid) and the triangulation (grey and dashed). Without rotating the coordinate system a resolution factor $K = 5$ is needed (left). When rotating before computing the CLF a resolution factor $K = 2$ is needed (center). For comparison the quadratic CLF found with SDP (right) along with the semi-minor and major axes (gray and dashed).

ones are actually mapped to areas where simplices are not densely distributed. In the majority of cases, however, the number of simplices is the same or less.

4 Computing Lyapunov Functions using SDP or LP

In this section, we compare different solvers for the established SDP approach to compute QCLF. Furthermore, we compare our LP approach from [4] to compute piecewise linear CLFs to the SDP approach. For the comparison, we construct twenty 2×2 matrices, which we denote by

$$A_{k+4(j-1)} = V_j E_k V_j^{-1}, \tag{12}$$

where $j = 1, 2, 3, 4, 5$, $k = 1, 2, 3, 4$,

$$\theta = \begin{bmatrix} 0 & \frac{9\pi}{40} & \frac{9\pi}{20} & \frac{27\pi}{40} & \frac{9\pi}{10} \end{bmatrix}, \quad d = \begin{bmatrix} 1 & 2.1544 & 4.6416 & 10 \end{bmatrix},$$

$$R(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix},$$

$$V_j = \begin{bmatrix} R(\theta_j) \begin{bmatrix} 1 \\ 1 \end{bmatrix} & R\left(\theta_j + \frac{\pi}{3}\right) \begin{bmatrix} 1 \\ 1 \end{bmatrix} \end{bmatrix} \quad \text{and} \quad E_k = \begin{bmatrix} -1 & -d_k \\ d_k & -1 \end{bmatrix}.$$

We then consider the switched systems defined by (2) with these A_m and $\emptyset \neq \mathcal{P} \subseteq \mathcal{Q} = \{1, 2, \ldots, 20\}$. Note that the origin is exponentially stable for each individual system, since the eigenvalues of (12) are given by $\lambda = -1 \pm d_k\sqrt{-1}$. Moreover, they all have the same convergence rate. However, the spiralling rate is different and depends on k. System trajectories, starting at $(1, 1)$, are shown in Figure 7.

Table 2: The effect of preconditioning with rotation on the lowest resolution parameter K needed to compute a CLF for sets of two-dimensional ($n = 2$) and three-dimensional ($n = 3$) systems from [3]. $\Delta K := K_{\text{no R}} - K_{\text{with R}}$ and thus $\Delta K > 0$ means that we needed fewer simplices when we used the rotation and $\Delta K < 0$ means that we needed more simplices when we used the rotation.

ΔK	# cases $n = 2$	# cases $n = 3$
-21	$\star$	1
-4	$\star$	1
-2	$\star$	1
-1	28	1
0	1126	9
1	80	4
2	2	3
3	3	$\star$
4	40	$\star$
5	$\star$	1
14	$\star$	1
Total:	1279	22

4.1 Numerical Study: SDP

In this section, we compare different solvers and different tolerance parameters to compute QCLF for the switched system by solving LMI. We start by investigating the stability of all switched linear systems defined by (2) and all the subsets $\mathcal{P} \neq \emptyset$ of an a priori fixed superset $\mathcal{Q}$. In more detail, for each switched system corresponding to a given subset $\mathcal{P} \subseteq \mathcal{Q}$, we use SDP to search for a QCLF that solves the LMIs given by

$$P - \varepsilon I_n \succeq 0, \tag{13}$$

$$A_m^T P + P A_m + \varepsilon I_n \preceq 0, \quad \forall m \in \mathcal{P} \subseteq \mathcal{Q}. \tag{14}$$

Here, $\varepsilon > 0$ is a small constant to force positive definiteness.

To speed up the computations, we use two simple tricks. First, if there is not a solution for a given subset $\mathcal{P} \subset \mathcal{Q}$, then there cannot be a solution for any superset P^*, $\mathcal{Q} \supseteq \mathcal{P}^* \supseteq \mathcal{P}$. Second, a solution to the LMI given by (13)-(14) cannot exist for a particular subset $\mathcal{P}$ if there is a matrix in the cone defined by the matrices A_m, $m \in \mathcal{P}$, that is not Hurwitz, or more specifically, if for some choice of λ_m matrix $\widetilde{A} := \sum_{m \in \mathcal{P}} \lambda_m A_m$ is not Hurwitz. The reason for this is that

$$\widetilde{A}^T P + P \widetilde{A} \prec 0 \tag{15}$$

has a solution $P \succ 0$ if and only if $\widetilde{A}$ is Hurwitz and

$$\widetilde{A}^T P + P \widetilde{A} = \sum_{m \in \mathcal{P}} \lambda_m \left(A_m^T P + P A_m \right). \tag{16}$$

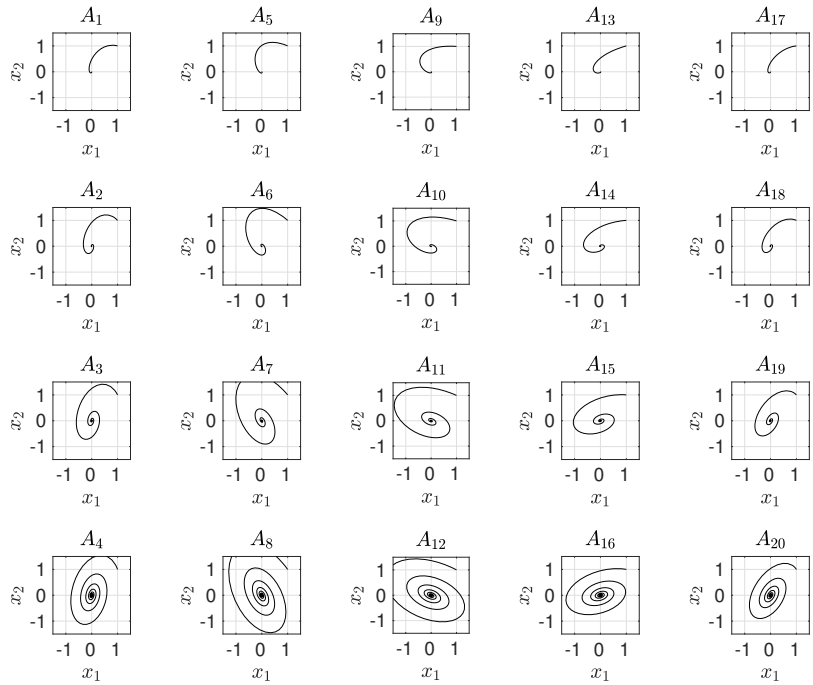


Fig. 7: Trajectories of the systems $\dot{\mathbf{x}} = A_m\mathbf{x}$, $m = 1, 2, \ldots, 20$, starting at $(1, 1)$. The matrices A_m are defined in (12). The figure is adapted from [3].

Thus, if $\widetilde{A}$ is not Hurwitz then we know that there cannot exist a solution to (13)-(14), because otherwise the left-hand-side of (16) is negative definite with the $P \succ 0$ from the solution, which is contradictory to (15) not having a solution $P \succ 0$. Hence, as a computationally very cheap test to eliminate unnecessary computations, for a subset $\mathcal{P}$, we first verify whether $\sum_{m\in\mathcal{P}} A_m$ is Hurwitz or not. If it is not Hurwitz then we do not need to attempt to solve (13)-(14) for this particular $\mathcal{P}$.

For our computations, we use the MATLAB [29] toolbox YALMIP [26] to search for a solution for all possible $\mathcal{P} \subseteq \{1, 2, \ldots, 20\}$ using the solvers SeDuMi [41], MOSEK [32], and SDPT3 [42]. For all solvers, we use their default parameters and tolerances. To identify false positives, we additionally verify that a solution reported by a solver truly satisfies (6). This is necessary because solvers sometimes report solutions to problems that do not really fulfill the constraints, particularly, if the problems are numerically badly conditioned.

For some subsets $\mathcal{P} \subseteq \mathcal{Q}$, the origin is an exponentially stable equilibrium of the corresponding switched system, for instance, if we choose only singlets. However, there are subsets $\mathcal{P}$, for which the origin is not exponentially stable. Furthermore, in some cases there are subsets $\mathcal{P}$, for which the origin is exponen-

Table 3: Overview of the results in two dimensions. PT: Solution reported and passed test, FP: False positive (solution reported but did not pass test), I: Infeasible, and T: Total number of LMIs problems. The tolerance parameter $\varepsilon = 10^{-3}$ appears reasonable and $\varepsilon = 10^{-16}$ does not. Note the effect of the simplification of not considering supersets of a set $\mathcal{P}$, for which the LMI problem is infeasible. With $\varepsilon = 10^{-3}$ only 1,366 LMIs have to be solved instead of $2^{20} - 1 = 1,047,296$, i.e. for every subsets $\mathcal{P} \neq \emptyset$ of $\mathcal{Q}$.

Solver	ε	PT	FP	I	T
MOSEK		1279	0	87	1366
SDPT3	10^{-3}	1279	0	87	1366
SeDuMi		1279	0	87	1366
MOSEK		1279	1,047,296	0	1,048,575
SDPT3	10^{-16}	1279	1,047,296	0	1,048,575
SeDuMi		1279	0	87	1366

tially stable, but there does not exist a QCLF for the corresponding switched linear system. We summarise the results of our computations in Table 3.

If we choose a sensible tolerance parameter, such as $\varepsilon = 10^{-3}$, then we obtain the same results independently of the solver used. If we choose a very low tolerance parameter such as $\varepsilon = 10^{-16}$, which is numerically close to zero, then we obtain, possibly as one would expect, more solutions. For two-dimensional systems, the results are not really useful, since the solvers MOSEK and SDPT3 report all LMIs as feasible, while solver SeDuMi delivers exactly the same results as when we use $\varepsilon = 10^{-3}$. Surprisingly however, $P \neq 0$, as one might expect for such a low tolerance. Importantly, for very badly conditioned five-dimensional systems, we showed in [3] that one obtains many more true solutions by using the very low tolerance parameter $\varepsilon = 10^{-16}$ and then verifying the solutions, than by using a more reasonable tolerance parameter $\varepsilon = 10^{-3}$. Thus, we propose to first use a tolerance that seems unreasonably low, like $\varepsilon = 10^{-16}$, and then verify the solutions obtained, since checking for false positives is computationally very cheap and we manage to obtain solutions that are missed with a more sensible tolerance parameter ε.

4.2 Numerical Study: LP vs. SDP

As we have discussed earlier in this paper, even if the origin of the switched system given by (2) is exponentially stable, a QCLF does not necessarily exist. However, the origin is an exponentially stable equilibrium for the switched system (2) if and only if there exists a piecewise linear CLF [17, 27, 28]. Moreover, a piecewise linear CLF can always be computed with our LP method from [4] as shown in [21]. Thus, we expect that by using the LP approach, we will be able to prove exponential stability of (2) for more subsets $\mathcal{P} \subseteq \mathcal{Q} = \{1, 2, \ldots, 20\}$ than by solving an LMI.

For our investigation, we compare the results from the last section, where we searched for a QCLF by means of SDP for two-dimensional systems, with the LP

Table 4: Ratio, in percentage, of the two-dimensional switched systems (1) successfully solved using SDP to problems successfully solved using our LP method, for matrices (12). Note that # refers to number of systems.

$\mathcal{Q}$	# solved (SDP)	# solved (LP)	SDP/LP	LP/tot.	total #
1	20	20	100%	100%	20
2	104	142	73.24%	74.74%	190
3	260	522	49.81%	45.79%	1,140
4	370	1092	33.88%	22.54%	4,845
5	316	1458	21.67%	9.404%	15,504
6	160	1261	12.69%	3.253%	38,760
7	44	696	6.322%	0.8978%	77,520
8	5	233	2.146%	0.1850%	125,970
9	0	42	0%	0.02501%	167,960
10	0	3	0%	0.001624%	184,756
11	0	0	$\star$	0%	167,960

Table 5: Ratio, in percentage, of the three-dimensional switched linear systems from [3] successfully solved using SDP to problems successfully solved using our LP method, where # refers to number of systems.

$\mathcal{Q}$	# solved (SDP)	# solved (LP)	SDP/LP	LP/tot.	total #
1	12	12	100%	100%	12
2	9	10	90%	15.15%	66
3	1	3	33.33%	1.36%	220
4	0	0	$\star$	0	495

approach from [4], where we search for a piecewise linear CLF using the Gurobi solver [20]. As shown in Table 4, the LP approach increasingly outperforms the SDP approach for larger $|\mathcal{P}|$. For instance, for $|\mathcal{P}| = 2$, the SDP approach guarantees exponential stability for only 75% of the systems, for which the LP approach guarantees it. For $|\mathcal{P}| = 3$, the percentage decreases to 50%, for $|\mathcal{P}| = 4$, to 33%, etc. For completeness, we also include Table 5 from [3], which shows similar results for three-dimensional systems. The results are comparable to the two-dimensional case.

5 The AngleAnalysis App

As we discussed thoroughly in Section 3, it is by no means easy to determine a good matrix R for preconditioning the switched linear system (1) for the computation of a piecewise linear CLF using LP. To offer some visual guidance for the preconditioning and in order to understand the challenge better, we developed the MATLAB application AngleAnalysis. It visualises the different local angles between two linear vector fields over the state space. That is, for the

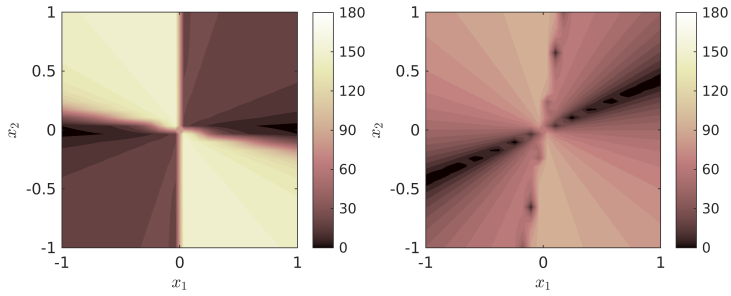


Fig. 8: Angle visualisation for matrices from [4] with $a = 0.1$ and $b = 13.25$ without preconditioning (left) and with $\lambda_a = 0.2$ and $\lambda_b = 0.8$ (right). A piecewise linear CLF was obtained for the preconditioned system using a much fewer simplices than without (lower K). The figure is adapted from [3].

systems $\dot{\mathbf{x}} = A\mathbf{x}$ and $\dot{\mathbf{x}} = B\mathbf{x}$, we visualise the angle

$$\angle(A\mathbf{x}, B\mathbf{x}) := \arccos\left(\frac{\langle A\mathbf{x}, B\mathbf{x}\rangle}{\|A\mathbf{x}\|\,\|B\mathbf{x}\|}\right) \tag{17}$$

as a function of $\mathbf{x}$, where $A, B \in \mathbb{R}^{n\times n}$, $n = 2, 3$ (see Figure 2 (a)). Note that $\langle \mathbf{x}, \mathbf{y}\rangle$ denotes the Euclidian scalar product of the vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$. In the following, we explain how the angle in (17) relates to the stability of the origin.

First, note that function $V\colon \mathbb{R}^n \to [0, \infty)$ is a Lyapunov function for the switched linear system (2) if and only if $V(\mathbf{0}) = 0$, $V(\mathbf{x}) > 0$ for $\mathbf{x} \neq \mathbf{0}$, and the directional derivative along any system trajectory is strictly negative, that is, $\langle \nabla V(\mathbf{x}), A_m\mathbf{x}\rangle < 0$ for all $m \in \mathcal{P}$ and $\mathbf{x} \neq \mathbf{0}$. It follows that for asymptotic stability of the switched system $\dot{\mathbf{x}} = A_m\mathbf{x}$, $\mathcal{P} = \{1, 2\}$, $A_1 = A$ and $A_2 := B$, the conditions $\langle \nabla V(\mathbf{x}), A\mathbf{x}\rangle < 0$ and $\langle \nabla V(\mathbf{x}), B\mathbf{x}\rangle < 0$ for all $\mathbf{x} \neq \mathbf{0}$ must hold. This implies that $\angle(\nabla V(\mathbf{x}), A\mathbf{x}) > 90°$ and $\angle(\nabla V(\mathbf{x}), B\mathbf{x}) > 90°$ must hold for all $\mathbf{x} \neq \mathbf{0}$. Then, intuitively, this condition is more difficult to satisfy if (17) tends to be large or, conversely, the smaller (17) the more "space" there is for the gradient $\nabla V(\mathbf{x})$. Hence, it should be easier to find a function $V\colon \mathbb{R}^n \to [0, \infty)$, whose gradient fulfils $\angle(\nabla V(\mathbf{x}), A\mathbf{x}) > 90°$ and $\angle(\nabla V(\mathbf{x}), B\mathbf{x}) > 90°$ for all $\mathbf{x} \neq \mathbf{0}$ if (17) is small for all $\mathbf{x} \neq \mathbf{0}$. However, note that this is only a local condition and even though the angle (17) is small, but not zero, for all $\mathbf{x} \neq \mathbf{0}$, this does not necessarily mean that the equilibrium is asymptotically stable for the switched system.

Given matrix $A \in \mathbb{R}^{n\times n}$ and matrix $B \in \mathbb{R}^{n\times n}$, the AngleAnalysis app visualises the angle given by (17) as a function of $\mathbf{x}$ through a heat-map, where small angles are represented by dark colours and large angles by bright colours. For $n = 2$, the heat-map is plotted in the square $[-1, 1]^2$. For $n = 3$, the heat-map is plotted in the square $[-1, 1]^2 \times \{x_3\}$, where the value of parameter x_3 is chosen by the user from the interval $[-1, 1]$.

To provide further insight, next, we look at two extreme cases. For instance, if there exists a $\hat{\mathbf{x}} \in \mathbb{R}^n$, for which (17) is 180°, then there exists a constant $c > 0$

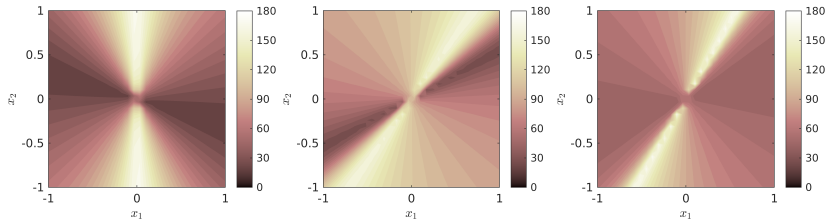


Fig. 9: Angle visualisation for A_3 and A_{12} at $x_3 = 0$. Without preconditioning (left), with $\lambda_6 = 0.7$ and $\lambda_9 = 0.3$ (center) and with $\lambda_6 = 0$ and $\lambda_9 = 1$ (right). Corresponding angle data can be found in Table 6. The figure is adapted from [3].

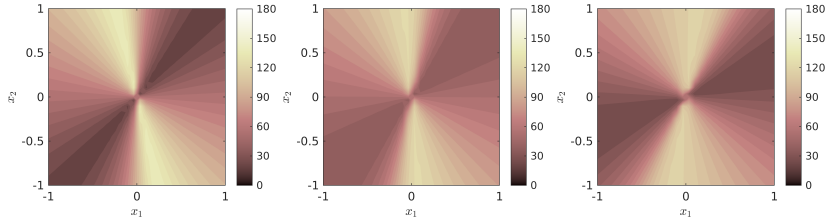


Fig. 10: Angle visualisation for A_3 and A_{12} at $x_3 = 0$. Without preconditioning (left), with $\lambda_3 = 0.7$ and $\lambda_{12} = 0.3$ (center), and with $\lambda_3 = 1$ and $\lambda_{12} = 0$ (right). Corresponding angle data can be found in Table 7. This figure is adapted from [3].

such that $A\hat{\mathbf{x}} = -cB\hat{\mathbf{x}}$ and a Lyapunov function V, such that

$$\langle \nabla V(\mathbf{x}), A\mathbf{x} \rangle < 0 \quad \text{and} \quad \langle \nabla V(\mathbf{x}), B\mathbf{x} \rangle < 0 \tag{18}$$

holds for all $\mathbf{x} \neq \mathbf{0}$, cannot exists. To see this, assume for the sake of the argument that (18) holds. Then this leads to the following contradiction:

$$0 > \langle \nabla V(\mathbf{x}), A\mathbf{x} \rangle = \langle \nabla V(\mathbf{x}), -cB\mathbf{x} \rangle = -c \langle \nabla V(\mathbf{x}), B\mathbf{x} \rangle > 0$$

for $\mathbf{x} = \hat{\mathbf{x}}$. Hence, if there exists a $\hat{\mathbf{x}} \in \mathbb{R}^n$, for which (17) is 180°, then a CLF for systems $\dot{\mathbf{x}} = A\mathbf{x}$ and $\dot{\mathbf{x}} = B\mathbf{x}$ does not exist. Another extreme case is given by (17) being 0° for all $\mathbf{x} \in \mathbb{R}^n$. Then, a Lyapunov function guarantees exponential stability of $\dot{\mathbf{x}} = A\mathbf{x}$ if and only if it guarantees exponential stability of $\dot{\mathbf{x}} = B\mathbf{x}$. In fact, $A = cB$ for some $c > 0$.

For some systems, the app clearly indicates that we can expect a reduction in the number of simplices after preconditioning, see Figure 8 and the reference [4]. In other cases the benefits of preconditioning do not become clear. For the 3×3 matrices A_m from the appendix of [3] we show in Figure 9 and Figure 10 the values of (17) by means of a heat-map with preconditioning and without preconditioning. Table 6 and Table 7 show the corresponding angle statistics.

Table 6: Maximum, minimum, mean, and standard deviation of (17) for matrix A_6 and matrix A_9 at $(x_1, x_2, x_3) \in [-1, 1]^3$ where x_1, x_2, and x_3 are 30 evenly spaced points.

λ_6	λ_9	Max	Min	Mean	Std	K
N/A	N/A	176.0	2.97	77.5	43.0	45
0.7	0.3	176.5	1.02	105.3	30.4	25
0	1	176.8	0.83	82.5	37.7	60

Table 7: Maximum, minimum, mean, and standard deviation of (17) for matrix A_3 and matrix A_{12} at $(x_1, x_2, x_3) \in [-1, 1]^3$ where x_1, x_2, and x_3 are 30 evenly spaced points.

λ_3	λ_{12}	Max	Min	Mean	Std	K
N/A	N/A	143.0	1.04	74.2	25.2	60
0.7	0.3	136.3	1.14	70.0	20.0	70
1	0	151.9	1.38	64.2	23.0	110

In both case, the app does not seem to provide support for deciding about the parameters in the preconditioning. The reason is that, although it is not difficult to precondition the system such that locally the gradients $\nabla V(\mathbf{x})$ fulfil $\angle(\nabla V(\mathbf{x}), A\mathbf{x}) > 90°$ and $\angle(\nabla V(\mathbf{x}), B\mathbf{x}) > 90°$, $V \colon \mathbb{R}^n \to [0, \infty)$ is a continuous function on $\mathbb{R}^n$ and the problem of the existence of a CLF is highly non-local.

6 Conclusions

We discussed methods to compute Lyapunov functions for switched linear systems using semidefinite programming (SDP) and linear programming (LP). We compared the SDP solvers SeDuMi, MOSEK, and SDPT3 for computing quadratic common Lyapunov functions (QCLFs) for a variety of sets of linear systems. Such common Lyapunov functions (CLFs) are also Lyapunov functions for the corresponding switched linear systems. Furthermore, we tried different tolerance parameters $\varepsilon > 0$ to force positive definiteness of matrix A through $A \succeq \varepsilon I_n$ and obtained the surprising result that using very small ε enhanced the performance of the approach in some examples. Hence, it is advisable to use a very low $\varepsilon > 0$, such as $\varepsilon = 10^{-16}$, and then verify solutions afterwards.

We compared computing QCLF with SDP to computing piecewise linear CLF using the LP approach from [4]. As expected, since switched linear systems with an asymptotically stable equilibrium at the origin do not necessarily possess a QCLF, the LP approach is able to assert stability for more switched systems than the SDP approach. In fact, it was proved in a recent publication [21] that our LP approach always succeeds in computing a CLF, if enough simplices are used in the triangulation, that is, if resolution parameter K is large enough. A drawback of the LP approach is that a triangulation of the state space is needed, which can result in a very high number of simplices in higher dimensions. We described different preconditioning methods for our LP method, including a new method based on rotations that has not been studied before, to better deal with this problem. Unfortunately, it is not clear how to choose the optimal parameters for the preconditioning methods. Finally, we developed the MATLAB app AngleAnalysis to get visual information about the effect of the preconditioning and gave examples of its use. In some cases, it gives a clear indication of which parameters are good, but in other cases this is not as clear.

In summary, we presented state-of-the art approaches to determine the asymptotic stability of the origin for switched linear systems. The traditional SDP approach is faster, but it is often not able to assert the asymptotic stability of the origin. Our LP approach is computationally more demanding, but is able to assert asymptotic stability of the origin for many more switched linear systems. In theory, it can always assert the asymptotic stability of an asymptotically stable equilibrium given sufficient computational power. For future research, it seems promising to extend the SDP approach to higher-order polynomial Lyapunov functions, as this approach is less conservative. Furthermore, as shown in this paper, preconditioning can greatly accelerate the LP approach. We will further investigate how to choose optimal parameters for the preconditioning.

Acknowledgements This work was supported in part by the Icelandic Research Fund under Grant 228725-051 and has received funding from European Union's Horizon 2020 research and innovation programme under grant agreement no. 965417.

References

1. Agrachev, A., Liberzon, D.: Lie-algebraic stability criteria for switched systems. SIAM J. Control Optim. **40**(1), 253–269 (2001)
2. Andersen, S.: Common Lyapunov Functions for Switched Linear Systems: Computations and comparisons of numerical approaches. Master's thesis, University of Iceland, Dunhagi 5, 107 Reykjavík, Iceland (10 2023)
3. Andersen., S., August., E., Hafstein., S., Piccini., J.: Lyapunov function computation for linear switched systems: Comparison of SDP and LP approaches. In: Proceedings of the 13th International Conference on Simulation and Modeling Methodologies, Technologies and Applications - SIMULTECH. pp. 61–70. INSTICC, SciTePress (2023). https://doi.org/10.5220/0012084100003546
4. Andersen, S., Giesl, P., Hafstein, S.: Common Lyapunov functions for switched linear systems: Linear programming-based approach. IEEE Control Systems Letters **7**, 901–906 (2023). https://doi.org/10.1109/LCSYS.2022.3228857
5. Aubin, J.P., Cellina, A.: Differential Inclusions. Springer (1984)
6. Blanchini, F.: Nonquadratic Lyapunov functions for robust control. Automatica **31**(3), 451–461 (1995)
7. Blanchini, F., Carabelli, S.: Robust stabilization via computer-generated Lyapunov functions: an application to a magnetic levitation system. In: Proceedings of the 33th IEEE Conference on Decision and Control. vol. 2, pp. 1105–1106 (Dec 1994)
8. Blanchini, F., Miani, S.: Set-theoretic methods in control. Systems & Control: Foundations & Applications, Birkhäuser (2008)
9. Boyd, S., El Ghaoui, L., Feron, E., Balakrishnan, V.: Linear matrix inequalities in system and control theory, SIAM Studies in Applied Mathematics, vol. 15. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA (1994). https://doi.org/10.1137/1.9781611970777, http://dx.doi.org/10.1137/1.9781611970777
10. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press (2004)
11. Brayton, R., Tong, C.: Stability of dynamical systems: A constructive approach. IEEE Trans. Circuits and Systems **26**(4), 224–234 (Apr 1979). https://doi.org/10.1109/TCS.1980.1084749
12. Brayton, R., Tong, C.: Constructive stability and asymptotic stability of dynamical systems. IEEE Trans. Circuits and Systems **27**(11), 1121–1130 (Nov 1980). https://doi.org/10.1109/TCS.1980.1084749
13. Clarke, F.: Optimization and Nonsmooth Analysis. Classics in Applied Mathematics, SIAM (1990)
14. Cohen, N., Lewkowicz, I.: A necessary and sufficient criterion for the stability of a convex set of matrices. IEEE Trans. Automat. Control **38**(4), 611–615 (1993)
15. El-Farra, N., Gani, A., Christofides, P.: A switched systems approach for the analysis and control of mode transitions in biological networks. In: Proceedings of the 2005 American Control Conference. pp. 3247–3252 (2005)
16. Giesl, P., Hafstein, S.: Review of computational methods for Lyapunov functions. Discrete Contin. Dyn. Syst. Ser. B **20**(8), 2291–2331 (2015)
17. Goebel, R., Hu, T., Teel, A.: Current Trends in Nonlinear Systems and Control. Systems and Control: Foundations & Applications, chap. Dual Matrix Inequalities in Stability and Performance Analysis of Linear Differential/Difference Inclusions, pp. 103–122. Birkhauser (2006)

18. Goebel, R., Sanfelice, R., Teel, A.: Hybrid Dynamical Systems. Princeton University Press (2012)
19. Grüne, L., Junge, O.: Gewöhnliche Differentialgleichungen. Springer, Wiesbaden, 2nd edn. (2016)
20. Gurobi Optimization, LLC: Gurobi Optimizer Reference Manual (2023), https://www.gurobi.com
21. Hafstein, S.: Lyapunov functions for switched linear systems: Proof of convergence for an LP computational approach. IEEE Control Systems Letters **7**, 3283–3288 (2023). https://doi.org/10.1109/LCSYS.2022.3228857
22. Harding, E.C., Franks, N.P., Wisden, W.: Sleep and thermoregulation. Current Opinion in Psychology **15**, 7–13 (2020)
23. Khalil, H.: Nonlinear Systems. Pearson, 3. edn. (2002)
24. Liberzon, D.: Switching in systems and control. Systems & Control: Foundations & Applications, Birkhäuser (2003)
25. Liberzon, D.: Switching in Systems and Control. Birkhäuser Boston (2003). https://doi.org/10.1007/978-1-4612-0017-8
26. Löfberg, J.: YALMIP: A toolbox for modeling and optimization in MATLAB. In: In Proceedings of the CACSD Conference. Taipei, Taiwan (2004)
27. Mason, P., Boscain, U., Chitour, Y.: Common polynomial Lyapunov functions for linear switched systems. SIAM J. Control Optim. **45**(1), 226–245 (2006)
28. Mason, P., Chitour, T., Sigalotti, M.: On universal classes of Lyapunov functions for linear switched systems (2022), arXiv:2208.09179
29. MATLAB: 9.12.0.2039608 (R2022a) Update 5. The MathWorks Inc., Natick, Massachusetts (2022)
30. Molchanov, A., Pyatnitskiy, E.: Lyapunov functions that specify necessary and sufficient conditions of absolute stability of nonlinear nonstationary control systems I, II. Automat. Remote Control **47**, 344–354, 443–451 (1986)
31. Molchanov, A., Pyatnitskiy, E.: Criteria of asymptotic stability of differential and difference inclusions encountered in control theory. Systems Control Lett. **13**(1), 59–64 (1989). https://doi.org/http://dx.doi.org/10.1016/0167-6911(89)90021-2, http://www.sciencedirect.com/science/article/pii/0167691189900212
32. MOSEK ApS: The MOSEK optimization toolbox for MATLAB manual. Version 9.0. (2019), http://docs.mosek.com/9.0/toolbox/index.html
33. Ohta, Y.: On the construction of piecewise linear Lyapunov functions. In: Proceedings of the 40th IEEE Conference on Decision and Control. vol. 3, pp. 2173–2178 (Dec 2001)
34. Ohta, Y., Tsuji, M.: A generalization of piecewise linear Lyapunov functions. In: Proceedings of the 42nd IEEE Conference on Decision and Control. vol. 5, pp. 5091–5096 (Dec 2003). https://doi.org/10.1109/CDC.2003.1272443
35. Parrilo, P.A.: Semidefinite programming relaxations for semialgebraic problems. Math. Program., Ser. B **96**, 293–320 (2003)
36. Piccini, J., August, E., Hafstein, S., Andersen, S.: Sufficient stability conditions for a class of switched systems with multiple steady states. IEEE Control Systems Letters **7**, 2653–2658 (2023)
37. Polanski, A.: Lyapunov functions construction by linear programming. IEEE Trans. Automat. Control **42**, 1113–1116 (1997)
38. Polanski, A.: On absolute stability analysis by polyhedral Lyapunov functions. Automatica **36**, 573–578 (2000)
39. Prajna, S., Papachristodoulou, A.: Analysis of switched and hybrid systems - beyond piecewise quadratic methods. In: Proceedings of the 2003 American Control Conference. pp. 2779–2784 (2003)

40. Shorten, R., Wirth, F., Mason, O., Wulff, K., King, C.: Stability criteria for switched and hybrid systems. SIAM Review **49**(4), 545–592 (2007)
41. Sturm, J.: Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. Optimization Methods and Software **11–12**, 625–653 (1999), version 1.05 available from `http://fewcal.kub.nl/sturm`
42. Toh, K.C., Todd, M.J., Tütüncü, R.H.: Sdpt3 - a matlab software package for semidefinite programming, version 1.3. Optimization Methods and Software **11**(1-4), 545–581 (1999). https://doi.org/10.1080/10556789908805762
43. Vandenberghe, L., Boyd, S.: Semidefinite programming. SIAM Review **38**, 49–95 (1996)
44. Veer, S., Poulakakis, I.: Switched systems with multiple equilibria under disturbances: Boundedness and practical stability. IEEE Transactions on Automatic Control **65**(6), 2371–2386 (2020)
45. Wang, K., Michel, A.: On the stability of a family of nonlinear time-varying system. IEEE Trans. Circuits and Systems **43**(7), 517–531 (Jul 1996)
46. Yfoulis, C., Shorten, R.: A numerical technique for the stability analysis of linear switched systems. Int. J. Control **77**(11), 1019–1039 (2004)

Paper III

Investigating Average Dwell-Time Constraints on Multiple Lyapunov Functions for Switched Linear Systems

S. Andersen, S. Hafstein

The 13th IFAC Symposium on Nonlinear Control Systems, pp. 150–155, 2025.

This article is reproduced in full with permission from Elsevier.

We present a new method to compute the minimum average dwell-time needed to assert the global exponential stability of the equilibrium at the origin for a switched linear system. The method attempts to compute compatible Lyapunov functions for the individual subsystems of the switched system using linear programming or linear matrix inequalities optimization problems. We test the method on four examples to demonstrate its applicability.

Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 59-19 (2025) 150–155

IFAC Papers Online CONFERENCE PAPER ARCHIVE

Investigating Average Dwell-Time Constraints on Multiple Lyapunov Functions for Switched Linear Systems

Stefania Andersen * **Sigurdur Hafstein** *

* *University of Iceland, Faculty of Physical Sciences, Dunhagi 5, 107 Reykjavik, Iceland (saa20@hi.is, shafstein@hi.is)*

Abstract: We present a new method to compute the minimum average dwell-time needed to assert the global exponential stability of the equilibrium at the origin for a switched linear system. The method attempts to compute compatible Lyapunov functions for the individual subsystems of the switched system using linear programming or linear matrix inequalities optimization problems. We test the method on four examples to demonstrate its applicability.

Copyright © 2025 The Authors. This is an open access article under the CC BY-NC-ND license (https://creativecommons.org/licenses/by-nc-nd/4.0/)

Keywords: Switched systems, minimum average dwell-time, Lyapunov function, numerical method.

1. INTRODUCTION

We consider the n-dimensional switched linear system

$$\dot{\mathbf{x}} = A_{\sigma(t)}\mathbf{x}, \quad \sigma : [0,\infty) \to \mathcal{P} := \{1, 2, \dots, P\}, \tag{1}$$

where $A_i \in \mathbb{R}^{n\times n}$ and the origin is globally exponentially stable (GES) for all the individual subsystems $\dot{\mathbf{x}} = A_i\mathbf{x}$, $i \in \mathcal{P}$, and where the switching signal σ is right-continuous and only has a finite number of discontinuity-points on every finite time-interval. We are interested in the stability of the equilibrium at the origin for the switched system. Note that even though the origin is GES for all the individual subsystems, it might not be stable for the arbitrary switched system (1), see e.g. Remark 2.3 in Liberzon (2003) for a counterexample. However, by limiting the allowed rate of switching sufficiently, the origin will always be GES for the switched system (1).

An appropriate concept to discuss the allowed rate of switching is the so-called average dwell-time, see Hespanha and Morse (1999) or, e.g., Chapter 3.2 in Liberzon (2003). We say that a switching signal σ has an average dwell-time $\tau > 0$, if there exists $N_0 > 0$, such that

$$N_\sigma(t,s) \le N_0 + \frac{t-s}{\tau} \quad \text{for all } t \ge s \ge 0,$$

where $N_\sigma(t,s)$ denotes the number of discontinuity-points of σ on the open interval (s,t), i.e. $N_\sigma(t,s)$ is the number of times that σ switches value between times s and t. The set of all switching signals with average dwell-time τ is denoted by Σ_τ and we talk about the switched system (1) with average dwell-time τ, if σ in (1) can be an arbitrary element from Σ_τ. When we say that the origin is GES for (1) with average dwell-time τ, we mean that there exist constants $C \ge 1$ and $\beta > 0$, such that

$$\|\mathbf{x}(t)\| \le C\,\mathrm{e}^{-\beta t}\,\|\mathbf{x}(0)\|$$

for all solutions $\mathbf{x}(\cdot)$ to (1) with $\sigma \in \Sigma_\tau$; $\|\cdot\|$ denotes the Euclidian norm.

⋆ This work was supported in part by the Icelandic Research Fund under Grant 228725-051.

An interesting question is now:

> What is the minimum average dwell-time τ, such that the origin is GES for the switched system (1)?

A sufficient condition is given in terms of Lyapunov functions for the subsystems in the following theorem adapted from Theorem 3.2 in Liberzon (2003); recall that $\mathcal{K}_\infty$ is the set of strictly increasing, unbounded, continuous functions $[0,\infty) \to [0,\infty)$, that are zero at zero.

Theorem 1. Assume that for each $i \in \mathcal{P}$, there exists a locally Lipschitz continuous function $V_i\colon \mathbb{R}^n \to \mathbb{R}$, such that

$$\underline{a}^*(\|\mathbf{x}\|) \le V_i(\mathbf{x}) \le \overline{a}^*(\|\mathbf{x}\|), \quad \forall i \in \mathcal{P}, \tag{2a}$$
$$D^+V_i(\mathbf{x}, A_i\mathbf{x}) \le -\alpha V_i(\mathbf{x}), \quad \forall i \in \mathcal{P}, \tag{2b}$$
$$V_i(\mathbf{x}) \le \mu V_j(\mathbf{x}), \quad \forall i,j \in \mathcal{P}, \tag{2c}$$

where $\overline{a}^*, \underline{a}^* \in \mathcal{K}_\infty$, $\alpha > 0$, and $\mu \ge 1$. Then the origin is GES for the switched system (1) with average dwell-time τ, if τ satisfies

$$\tau > \frac{\ln(\mu)}{\alpha}. \tag{3}$$

Here,

$$\begin{aligned} D^+V_i(\mathbf{x}, A_i\mathbf{x}) &:= D^+(V_i \circ \mathbf{x})(t)\big|_{t=0} \\ &:= \limsup_{h\to 0+} \frac{V_i(\mathbf{x}(t+h)) - V_i(\mathbf{x}(t))}{h}\bigg|_{t=0} \\ &= \limsup_{h\to 0+} \frac{V_i(\mathbf{x}+hA_i\mathbf{x}) - V_i(\mathbf{x})}{h} \end{aligned}$$

is the Dini-derivative of V_i along the solution trajectories of the system $\dot{\mathbf{x}} = A_i\mathbf{x}$. The proof can be copied almost verbatim from the proof of Theorem 3.2 in Liberzon (2003), although our V_i are only locally Lipschitz continuous and we use the Dini-derivative instead of the usual derivative, because with $W(t) := \mathrm{e}^{\alpha t}V_{\sigma(t)}(\mathbf{x}(t))$ we have for two consecutive discontinuity times t_j and t_{j+1} of σ for every $t_j \le t < t_{j+1}$, that

2405-8963 Copyright © 2025 The Authors. This is an open access article under the CC BY-NC-ND license.
Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2025.11.026

$$
\begin{aligned}
W(t) &= W(t_j) + \int_{t_j}^{t} D^+W(\tau)\,\mathrm{d}\tau \\
&= W(t_j) + \int_{t_j}^{t} \Big[\alpha\,\mathrm{e}^{\alpha\tau}\, V_{\sigma(t_j)}(\mathbf{x}(\tau)) \\
&\qquad + \mathrm{e}^{\alpha\tau}\, D^+V_{\sigma(t_j)}(\mathbf{x}(\tau), A_{\sigma(t_j)}\mathbf{x}(\tau))\Big]\,\mathrm{d}\tau \\
&\le W(t_j) \quad \text{by (2b)},
\end{aligned}
$$

see e.g. Theorem 1 in Hagood and Thomson (2006).

The following corollary to Theorem 1 was used in Hafstein and Tanwani (2023) to develop a method to compute a lower bound on the minimum average dwell-time τ for the switched system (1), such that the origin is GES.

Corollary 2. Assume that for each $i \in \mathcal{P}$, there exists a locally Lipschitz continuous functions $V_i\colon \mathbb{R}^n \to \mathbb{R}$, such that

$$
\begin{aligned}
\underline{a}\,\|\mathbf{x}\|^d \le V_i(\mathbf{x}) \le \overline{a}\,\|\mathbf{x}\|^d, &\quad \forall i \in \mathcal{P}, \\
D^+V_i(\mathbf{x}, A_i\mathbf{x}) \le -\alpha\|\mathbf{x}\|^d, &\quad \forall i \in \mathcal{P}, \\
V_i(\mathbf{x}) \le \mu V_j(\mathbf{x}), &\quad \forall i,j \in \mathcal{P},
\end{aligned} \tag{4}
$$

$\overline{a}, \underline{a}, \alpha, d > 0$, and $\mu \ge 1$. Then the origin is GES for the switched system (1) with average dwell-time τ for every

$$
\tau > \frac{\overline{a}\ln(\mu)}{\alpha}. \tag{5}
$$

Note that in the corollary the $\mathcal{K}_\infty$ functions $\underline{a}^*$ and $\overline{a}^*$, have been replaced with the $\mathcal{K}_\infty$ functions $\mathbf{x} \mapsto \underline{a}\|\mathbf{x}\|$ and $\mathbf{x} \mapsto \overline{a}\|\mathbf{x}\|$ for constants $0 < \underline{a} \le \overline{a}$. The corollary follows immediately from $-\|\mathbf{x}\|^d \le -V_i(\mathbf{x})/\overline{a}$.

It is shown in Hafstein and Tanwani (2023), that with $d = 1$ and $d = 2$, one can fix $\underline{a}, \overline{a} > 0$, and $\mu \ge 1$, and then use optimization to maximize $\alpha > 0$, such that the conditions of Corollary 2 are fulfilled. From formula (5) one then obtains a lower bound on the minimum average dwell-time needed to assert GES of the origin for the switched system (1).

For $d = 2$ this was done by searching for quadratic Lyapunov functions $V_i(\mathbf{x}) = \mathbf{x}^T P_i \mathbf{x}$, $i \in \mathcal{P}$, by solving the linear matrix inequality (LMI) opimazation problem:

$$
\begin{aligned}
&\text{maximize } \alpha \\
&\text{subject to} \\
&\underline{a}I \preceq P_i \ \text{ and } \ P_i \preceq \overline{a}I, \quad \forall i \in \mathcal{P}, && \text{(6a)} \\
&A_i^T P_i + P_i A_i \preceq -\alpha I, \quad \forall i \in \mathcal{P}, \\
&P_i \preceq \mu P_j, \quad \forall i,j \in \mathcal{P}. && \text{(6b)}
\end{aligned}
$$

In these formulas I denotes the $n \times n$ identity matrix and $A \preceq B$ means that the matrix $B - A \in \mathbb{R}^{n\times n}$ is symmetric and positive semidefinite. The variables of the LMI optimization problem are $\alpha > 0$ and the symmetric matrices $P_i \in \mathbb{R}^{n\times n}$, $i \in \mathcal{P}$.

For $d = 1$, and this was the main contribution of Hafstein and Tanwani (2023), linear programming (LP) was used to parameterize piecewise linear Lyapunov functions V_i fulfilling the conditions of Corollary 2. In the three examples presented, the LP approach with $d = 1$ outperformed the LMI approach with $d = 2$, which is not surprising because the set of piecewise linear Lyapunov functions is richer than the set of quadratic Lyapunov functions.

The advantage of using the conditions of Corollary 2 on the Lyapunov functions V_i, rather than the conditions in Theorem 1, is that the former is linear in the optimization variables while the latter is bilinear. Hence, α can be maximized using efficient and well-understood methods; LP in the case of $d = 1$ and semidefinite programming (SDP) in the case $d = 2$.

However, there is some conservatism involved in the conditions (4) in Corollary 2 with respect to the conditions (2b) in Theorem 1. Further, even after the maximum α has been found using a fixed $\mu \ge 1$, the corresponding lower bound τ in formula (3), on the needed average dwell-time to assert GES, might be suboptimal. Hence, in this paper we will derive and study methods to find low values for the average dwell-time τ using the parameters α and μ directly. We will use LP and LMI optimization problems, strongly based on those in Hafstein and Tanwani (2023), to construct Lyapunov functions, but in addition we will investigate methods to improve the values of α and μ in order to obtain lower τ.

The paper is organized as follows. In Section 2 we discuss how to parameterize compatible Lyapunov functions for the individual subsystems of the switched system (1) using LP or LMI optimization problems. In Section 3 we describe our method to compute a minimum average dwell-time τ that asserts the origin is GES for the system (1) and in Section 4 we apply our method to four examples and discuss the results, before we conclude the paper in Section 5.

2. PARAMETERIZING LYAPUNOV FUNCTIONS

The LMI optimization problem we use to construct quadratic Lyapunov functions is a straightforward adaptation of the LMI optimization problem from Hafstein and Tanwani (2023) discussed in the last section. For fixed $0 < \underline{a} < \overline{a}$, $\alpha > 0$, and $\mu \ge 1$, find symmetric matrices $P_i \in \mathbb{R}^{n\times n}$, $i \in \mathcal{P}$, fulfilling

$$
\begin{aligned}
&\underline{\text{LMI feasibility problem:}} \\
&\underline{a}I \preceq P_i \ \text{ and } \ P_i \preceq \overline{a}I, \quad \forall i \in \mathcal{P}, && \text{(7a)} \\
&A_i^T P_i + P_i A_i + \alpha P_i \preceq 0, \quad \forall i \in \mathcal{P}, && \text{(7b)} \\
&P_i \preceq \mu P_j, \quad \forall i,j \in \mathcal{P}. && \text{(7c)}
\end{aligned}
$$

If a solution exists to the LMI optimization problem (7), then the functions $V_i(\mathbf{x}) = \mathbf{x}^T P_i \mathbf{x}$, $i \in \mathcal{P}$, are Lyapunov functions to the subsystems $\dot{\mathbf{x}} = A_i\mathbf{x}$ fulfilling $V_i(\mathbf{x}) \le \mu V_j(\mathbf{x})$ for all $i,j \in \mathcal{P}$. Hence, by Theorem 1, $\tau = \ln(\mu)/\alpha$ is a lower bound on the needed average dwell-time to assert GES of the origin for the switched system (1).

The LP problem to construct piecewise linear Lyapunov functions V_i for the switched system (1) is more involved, mainly because one needs to triangulate the state-space $\mathbb{R}^n$. As we are essentially using the construction from Hafstein and Tanwani (2023), we only give a short description and refer the reader to Hafstein and Tanwani (2023) for the details. We use the triangulation $\mathcal{T}_K^{\mathbf{F}}$, where $K \in \mathbb{N}$ is a parameter determining the density or fineness of the triangulation, which consists of the n-simplices $\mathfrak{S}_\nu = \operatorname{co}\{\mathbf{0}, \mathbf{x}_1^\nu, \mathbf{x}_2^\nu, \ldots, \mathbf{x}_n^\nu\}$. Here ν is just an index to enumerate the simplices $\mathfrak{S}_\nu \in \mathcal{T}_K^{\mathbf{F}}$ and $\operatorname{co}\mathcal{A}$ denotes the convex hull of the vectors in $\mathcal{A} \subset \mathbb{R}^n$. Two different simplices $\mathfrak{S}_\nu, \mathfrak{S}_\mu \in \mathcal{T}_K^{\mathbf{F}}$ intersect in a common face, which might be as small as $\{\mathbf{0}\}$, and the set-theoretic union of the

simplices is a neighbourhood of the origin. As discussed in Andersen et al. (2024) the triangulation $\mathcal{T}_K^{\mathbf{F}}$ has $2^n K^{n-1} n!$ simplices.

To every simplex $\mathfrak{S}_\nu = \text{co}\{\mathbf{0}, \mathbf{x}_1^\nu, \mathbf{x}_2^\nu, \dots, \mathbf{x}_n^\nu\}$ we associate the variables $V^i_{\mathbf{x}_j^\nu} \in \mathbb{R}$, $i \in \mathcal{P}$ and $j = 1, 2, \dots, n$; note that if $\mathbf{y} = \mathbf{x}_j^\nu = \mathbf{x}_k^\mu$ for two different simplices $\mathfrak{S}_\nu$ and $\mathfrak{S}_\mu$, then $V^i_{\mathbf{y}} = V^i_{\mathbf{x}_j^\nu} = V^i_{\mathbf{x}_k^\mu}$. Once the values of the variables have been fixed, we can define the functions V_i on

$$\begin{aligned}\mathfrak{C}_\nu &:= \text{cone}\{\mathbf{x}_1^\nu, \mathbf{x}_2^\nu, \dots, \mathbf{x}_n^\nu\} \\ &= \left\{ \sum_{j=1}^n \lambda_j \mathbf{x}_j^\nu \,:\, \lambda_j \geq 0 \;\text{ for }\; j = 1, 2, \dots, n \right\}\end{aligned}$$

through $V_i(\mathbf{x}_j^\nu) := V^i_{\mathbf{x}_j^\nu}$ for $j = 1, 2, \dots, n$, and

$$V_i\left(\sum_{j=1}^n \lambda_j \mathbf{x}_j^\nu\right) := \sum_{j=1}^n \lambda_j V^i_{\mathbf{x}_j^\nu} = \sum_{j=1}^n \lambda_j V_i(\mathbf{x}_j^\nu). \tag{8}$$

Since $\mathfrak{S}_\nu$ is an n-simplex the vertices $\mathbf{x}_j^\nu$, $j = 1, 2, \dots, n$, are linearly independent and V_i is a well-defined linear function on $\mathfrak{C}_\nu$. Further, since two different simplices $\mathfrak{S}_\nu$ and $\mathfrak{S}_\mu$ intersect in a common face, V_i is well-defined and continuous on the set-theoretic union of the $\mathfrak{C}_\nu$, which is the whole of $\mathbb{R}^n$, because the union of the $\mathfrak{S}_\nu$ is a neighbourhood of the origin. Another useful equivalent formula for V_i on $\mathfrak{C}_\nu$ is

$$V_i(\mathbf{x}) = [\mathbf{v}_\nu^i]^T X_\nu^{-1} \mathbf{x},$$

where

$$[\mathbf{v}_\nu^i]^T := \begin{bmatrix} V^i_{\mathbf{x}_1^\nu} & V^i_{\mathbf{x}_2^\nu} & \cdots & V^i_{\mathbf{x}_n^\nu} \end{bmatrix} \in \mathbb{R}^{1\times n} \tag{9}$$

and

$$X_\nu := [\mathbf{x}_1^\nu \;\; \mathbf{x}_2^\nu \;\; \cdots \;\; \mathbf{x}_n^\nu] \in \mathbb{R}^{n\times n},$$

i.e. the $\mathbf{x}_j^\nu$ are the columns of X_ν; see e.g. Andersen et al. (2023) or Remark 9 in Giesl and Hafstein (2014) for this formula.

Given the triangulation $\mathcal{T}_K^F$, the LP feasibility problem to construct Lyapunov functions V_i for the switched system (1) is: for every simplex $\mathfrak{S}_\nu = \text{co}\{\mathbf{0}, \mathbf{x}_1^\nu, \mathbf{x}_2^\nu, \dots, \mathbf{x}_n^\nu\}$ we have the constraints for every $i, k \in \mathcal{P}$ and every $j = 1, 2, \dots, n$:

LP feasibility problem:

$$\underline{a}\,\|\mathbf{x}_j^\nu\| \leq V^i_{\mathbf{x}_j^\nu} \leq \overline{a}\,\|\mathbf{x}_j^\nu\|, \tag{10a}$$

$$[\mathbf{v}_\nu^i]^T X_\nu^{-1} A_i \mathbf{x}_j^\nu \leq -\alpha [\mathbf{v}_\nu^i]^T X_\nu^{-1} \mathbf{x}_j^\nu, \tag{10b}$$

$$V^i_{\mathbf{x}_j^\nu} \leq \mu V^k_{\mathbf{x}_j^\nu}, \tag{10c}$$

Theorem 3. Any solution to the LP feasibility problem (10) delivers Lyapunov functions $V_i \colon \mathbb{R}^n \to \mathbb{R}$, $i \in \mathcal{P}$, through the formulas (8), that fulfill (2) with appropriate $\underline{a}^*, \overline{a}^* \in \mathcal{K}_\infty$.

Proof: Follows immediately from formulas (8) and (9) and the fact that for every $\mathbf{x} \in \mathbb{R}^n$ and $i \in \mathcal{P}$, there exists a $\mathfrak{C}_\nu$, such that $\mathbf{x} + hA_i\mathbf{x} \in \mathfrak{C}_\nu$ for all small enough $0 \leq h$; hence by (10b) and with $\mathbf{x} = \sum_{j=1}^n \lambda_j \mathbf{x}_j^\nu$, $\lambda_j \geq 0$, we have

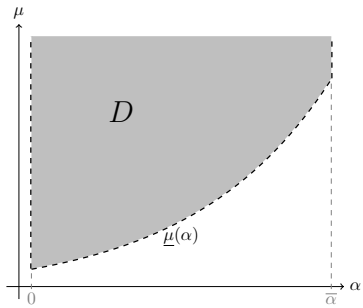


Fig. 1. An example of the area D where (2) has a feasible solution given α and μ. Note that the area extends indefinitely as $\mu \to \infty$. The function $\alpha \mapsto \underline{\mu}(\alpha)$ is discussed at the beginning of Section 4. All of our computations indicate that $\underline{\mu}(\alpha)$ is convex, although we have not been able to prove this statement.

$$\begin{aligned} D^+V_i(\mathbf{x}, A_i\mathbf{x}) &= [\mathbf{v}_\nu^i]^T X_\nu^{-1} A_i \mathbf{x} \\ &= \sum_{j=1}^n \lambda_j [\mathbf{v}_\nu^i]^T X_\nu^{-1} A_i \mathbf{x}_j^\nu \\ &\leq -\alpha \sum_{j=1}^n \lambda_j [\mathbf{v}_\nu^i]^T X_\nu^{-1} \mathbf{x}_j^\nu \\ &= -\alpha [\mathbf{v}_\nu^i]^T X_\nu^{-1} \mathbf{x} \\ &= -\alpha\, V_i(\mathbf{x}). \end{aligned}$$

□

3. MINIMIZING THE MINIMUM AVERAGE DWELL-TIME

For the switched system (1) define

$$D := \{(\alpha, \mu) \in (0, \infty) \times [1, \infty) \colon \text{(2) is feasible}\}, \tag{11}$$

see Figure 1 for a schematic picture. By Theorem 1 the switched system (1) is GES with average dwell-time $\tau > \ln(\mu)/\alpha$ for every $(\alpha, \mu) \in D$. To obtain the optimal minimum average dwell-time that asserts GES for the system (1), we attempt to

$$\text{minimize} \quad \tau := \tau(\alpha, \mu) := \frac{\ln(\mu)}{\alpha} \quad \text{on } D.$$

We will approximate the set D using two classes for the Lyapunov functions V_i. In more detail, we use the LP problem (10) to compute piecewise linear Lyapunov functions and the LMI optimization problem (7) to compute quadratic Lyapunov functions. Hence, checking whether a point (α, μ) is in D or not involves solving a feasibility problem. Now one can of course proceed with a brute force approach and check for numerous points on a dense grid if $(\alpha, \mu) \in D$, but since this is quite expensive numerically, we will suggest a more clever way to search systematically for (α, μ) that deliver low values for τ. Note that since τ is nonlinear and a function of two variables, it is not trivial figuring out which parameter or parameters to focus on in its minimization. Methods like gradient descent are a

poor option for optimizing both α and μ at the same time, because the update

$$\begin{bmatrix}\alpha_{k+1}\\ \mu_{k+1}\end{bmatrix} = \begin{bmatrix}\alpha_k\\ \mu_k\end{bmatrix} - \eta\nabla\tau(\alpha_k,\mu_k),$$

with the learning rate $\eta > 0$, does not consider the set D and τ obtains its minimum $\tau = 0$ for every $(\alpha, 1)$, $\alpha \in \mathbb{R}$.

Before we suggest our method to minimize τ in Algorithm 1, we first make the following useful observation:

3.1 Upper bound on α

Recall that a linear system $\dot{\mathbf{x}} = A\mathbf{x}$ is GES, if and only if A is Hurwitz. We can use this to our advantage and compute upper bounds on potential α, such that the LMI optimization problem (7) or the LP problem (10) are feasible.

LMI: The Lyapunov inequality $A_i^T P_i + P_i A_i \prec 0$ has a symmetric positive definite solution $P_i \succ 0$, if and only if A_i is Hurwitz, see Sylvester (1884). The inequality (7b) implies

$$\begin{aligned} 0 &\succeq A_i^T P_i + P_i A_j + \alpha P_i \\ &= A_i^T P_i + P_i A_i + \frac{\alpha}{2} P_i + \frac{\alpha}{2} P_i \\ &= \left(A_i + \frac{\alpha}{2} I\right)^T P_i + P_i \left(A_i + \frac{\alpha}{2} I\right) \end{aligned}$$

That is, for (7b) to be feasible, the real parts of the eigenvalues of $A_i + \frac{\alpha}{2} I$ must be strictly negative for all $i \in \mathcal{P}$. This gives us an upper bound $\overline{\alpha}_{\text{LMI}}$ on α for the LMI optimization problem (7).

LP: Similarly, consider that (10b) implies

$$\begin{aligned} 0 &\geq [\mathbf{v}_\nu^i]^T X_\nu^{-1} A_i \mathbf{x}_j^\nu + \alpha [\mathbf{v}_\nu^i]^T X_\nu^{-1} \mathbf{x}_j^\nu \\ &= [\mathbf{v}_\nu^i]^T X_\nu^{-1} (A_i + \alpha I) \mathbf{x}_j^\nu. \end{aligned}$$

That is, for (10b) to be feasible, the real parts of the eigenvalues of $A_i + \alpha I$ must be strictly negative for all $i \in \mathcal{P}$. This gives us an upper bound $\overline{\alpha}_{\text{LP}}$ on α for the LP problem (10).

3.2 Our method to minimize the average dwell-time

We suggest the following algorithm to search for points $(\alpha, \mu) \in D$ that minimize τ. Recall, that we either use the LP problem (10) to search for piecewise linear Lyapunov functions V_i or the LMI optimization problem (7) to search for quadratic Lyapunov functions V_i. Thus, we first fix the method to search for feasibility as either the LP problem (10) or the LMI optimization problem (7). The method is referred to as the *feasibility problem* with values (α, μ). If we are using the LP problem (10) we define $\overline{\alpha} := \overline{\alpha}_{\text{LP}}$ and if we are using the LMI optimization problem (7) we define $\overline{\alpha} := \overline{\alpha}_{\text{LMI}}$. Before the algorithm is executed, it is worthwhile to first check the feasibility for $(\alpha, \mu) = (0, 1)$. If it is feasible then one usually also gets a solution for a small $\alpha > 0$ and $\mu = 1$, which shows GES for the switched system (1) for arbitrary switchings. That is, no minimum average-dwell time is needed for stability.

Algorithm 1. We fix $N, M \in \mathbb{N}$, $N \geq 2$, and $a, b > 0$ and distribute the values $a = \tau_N < \tau_{N-1} < \ldots < \tau_2 < \tau_1 = b$ and $0 < \alpha_1 < \ldots < \alpha_M < \overline{\alpha}$ uniformly, i.e.

$$\tau_j = (a-b)\frac{j-1}{N-1} + b \quad \text{and} \quad \alpha_i = \overline{\alpha}\frac{i}{M+1}$$

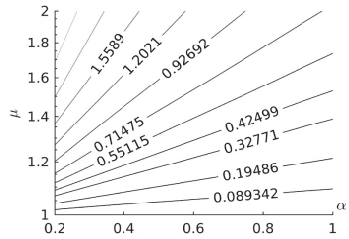


Fig. 2. Level sets for $\tau = \ln(\mu)/\alpha$ on $[0.2, 1] \times [1, 2]$. Note the μ-axis is logarithmical, therefore each level set is linear in the figure.

for $j = 1, 2, \ldots, N$ and $i = 1, 2, \ldots, M$.

Set $i = 1$ and $j = 1$. Then execute:

(1) Solve the feasibility problem with $\alpha = \alpha_i$ and $\mu = \exp(\tau_j \alpha_i)$.
(2)
- If there is a feasible solution to the problem, then increase j by one, i.e. $j \leftarrow j + 1$, and go back to step (1). That is, we decrease τ from τ_j to τ_{j+1} and check if this problem, with a lower average dwell-time and a smaller $\mu = \exp(\tau_{j+1}\alpha_i)$, also has a feasible solution.
- If there does not exists a feasible solution to the problem, then increase i by one, i.e. $i \leftarrow i+1$, and go back to step (1). That is, we increase α from α_i to α_{i+1} and set $\mu = \exp(\tau_j \alpha_{i+1})$ and check if this problem, with the same average dwell-time $\tau = \tau_j$, but with larger parameters α and μ, has a feasible solution.

Note that the algorithm follows the level sets of τ, see Figure 2 for an exemplary picture, as long as the optimization problem is not feasible. If the optimization problem is feasible we jump to the next level-set below for the same α and try our luck.

4. EXAMPLES

To visually interpret the examples the following nomenclature is useful: We say that the set D in (11) is *approximated by the function* $\alpha \mapsto \underline{\mu}(\alpha)$ if $\text{graph}(\underline{\mu}) \subset D$. Note that if $(\alpha, \mu^*) \in D$, then $(\alpha, \mu) \in D$ for all $\mu \geq \mu^*$. Hence, the epigraph $\{(\alpha, \mu) \in (0, \infty) \times [1, \infty) \colon \mu \geq \underline{\mu}(\alpha)\}$ is a subset of D. In the following we explicitly write the argument α in $\underline{\mu}(\alpha)$ to emphasize that $\underline{\mu}$ is a function of α.

For each example we computed $\underline{\mu}(\alpha)$ using both the LP problem (10) and the LMI optimization problem (7) with a brute forced approach; checking the feasibility of the optimization problem on a dense grid in the (α, μ) plane. Starting with $(\alpha, \mu) = (0, 1)$, where the optimization problem is not feasible, we show how Algorithm 1 first approaches $\underline{\mu}(\alpha)$ and then walks along it until a minimum is found. The LP problem was implemented with C++ and solved with Gurobi (Gurobi Optimization, LLC (2024)). The LMI optimization problem was implemented in Matlab (The

MathWorks Inc. (2023)) using YALMIP Löfberg (2004) and solved with SDPT3 Toh et al. (1999).

Example 1 The switched system (1) with

$$A_1 = \begin{bmatrix} -0.1 & -1 \\ 2 & -0.1 \end{bmatrix} \quad \text{and} \quad A_2 = \begin{bmatrix} -0.1 & -2 \\ 1 & -0.1 \end{bmatrix}, \tag{12}$$

taken from (Liberzon, 2003, p. 26). Progression of Algorithm 1 using LMI and LP optimization problems are shown in Figures 3a and 4a, respectively. Using the LMI optimization we obtain the minimum average dwell-time $\tau_{\text{LMI}} = 3.5195$ and using the LP optimization we obtain $\tau_{\text{LP}} = 3.5636$, compared to $\tau_{\text{LMI}} = 4.5283$ and $\tau_{\text{LP}} = 4.6241$ in Hafstein and Tanwani (2023).

Example 2 The switched system (1) with

$$A_1 = \begin{bmatrix} -1 & -1 \\ 1 & -1 \end{bmatrix} \quad \text{and} \quad A_2 = \begin{bmatrix} -1 & -10 \\ 0.1 & -1 \end{bmatrix} \tag{13}$$

taken from Dayawansa and Martin (1999). Progression of Algorithm 1 using LMI and LP optimization problems are shown in Figures 3b and 4b, respectively. Using the LMI optimization we obtain the minimum average dwell-time $\tau_{\text{LMI}} = 1.1532$ and using the LP optimization we obtain $\tau_{\text{LP}} = 0$, compared to $\tau_{\text{LMI}} = 17.0394$ and $\tau_{\text{LP}} = 0$ in Hafstein and Tanwani (2023).

Example 3 The switched system (1) with

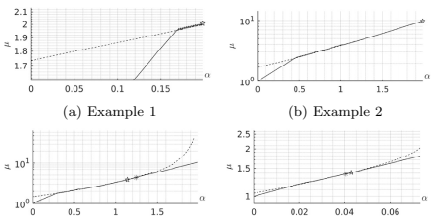
$$A_1 = \begin{bmatrix} -5 & 1 & 2 \\ 0 & -5 & 1 \\ 0 & 1 & -2 \end{bmatrix}, \qquad A_2 = \begin{bmatrix} -1 & 3 & 1 \\ 0 & -2 & 0 \\ 0 & 1 & -1 \end{bmatrix},$$
$$A_3 = \begin{bmatrix} 0 & 0 & 3 \\ -2 & -1 & -3 \\ -1 & 0 & -2 \end{bmatrix}, \qquad A_4 = \begin{bmatrix} -4 & 0 & -3 \\ 2 & -2 & 4 \\ 1 & 0 & -1 \end{bmatrix},$$
$$A_5 = \begin{bmatrix} -1 & 0 & 0 \\ -1 & -1 & -1 \\ -3 & 0 & -4 \end{bmatrix} \tag{14}$$

taken from Liu et al. (2021). Progression of Algorithm 1 using LMI and LP optimization problems are shown in Figures 3c and 4c, respectively. An interesting observation: $\underline{\mu}(\alpha)$ is identical to the worst case when only considering two subsystems at a time. Using the LMI optimization we obtain the minimum average dwell-time $\tau_{\text{LMI}} = 1.1852$ and using the LP optimization we obtain $\tau_{\text{LP}} = 0$, compared to $\tau_{\text{LMI}} = 4.6870$ and $\tau_{\text{LP}} = 0$ in Hafstein and Tanwani (2023), both of which are improvements over the approach proposed in Liu et al. (2021).

Example 4 The switched system (1) with

$$A_1 = \begin{bmatrix} -0.5302 & 0.0012 & 0.0873 \\ 0.2185 & -0.7494 & 0.5411 \\ 0.7370 & 0.1543 & -0.3606 \end{bmatrix},$$
$$A_2 = \begin{bmatrix} -0.5136 & 0.4419 & 0.3689 \\ 0.1840 & -0.3951 & 0.0080 \\ 0.3163 & 0.6099 & -1.0056 \end{bmatrix}. \tag{15}$$

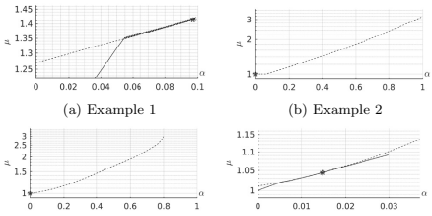
Progression of Algorithm 1 using LMI and LP optimization problems are shown in Figures 3d and 4d, respectively. Using the LMI optimization we obtain the minimum average dwell-time $\tau_{\text{LMI}} = 8.0090$ and using the LP optimization we obtain $\tau_{\text{LP}} = 3.0305$,



(a) Example 1 (b) Example 2

(c) Example 3 (d) Example 4

Fig. 3. Progression of Algorithm 1 for the switched system (1) with the matrices A_i from (12) (a), (13) (b), (14) (c), and (15) (d) using the LMI optimization problem (7). The dashed lines show $\underline{\mu}(\alpha)$ computed with a brute force approach. Marked with an asterisk is the location of the smallest τ on $\underline{\mu}(\alpha)$ and marked with a pentagram is the smallest τ found with Algorithm 1.



(a) Example 1 (b) Example 2

(c) Example 3 (d) Example 4

Fig. 4. Progression of Algorithm 1 for the switched system (1) with the matrices A_i from (12) (a), (13) (b), (14) (c) and (15) (d) using the LP optimization problem (10). The dashed lines show $\underline{\mu}(\alpha)$. Marked with an asterisk is the location of the smallest τ on $\underline{\mu}(\alpha)$ and marked with a pentagram is the smallest τ found with Algorithm 1.

4.1 Discussion of the Results

Note that for both two dimensional systems (Example 1 and Example 2) we get $\underline{\mu}(\alpha) \approx e^{C_1\alpha} + C_2$ for constants C_1 and C_2, whenever $\underline{\mu}(0) \neq 1$, posing the question; can we find a closed form solution for $\underline{\mu}(\alpha)$ in the two dimensional case? Another observation is that whenever $\underline{\mu}(0) \neq 1$, the smallest τ is obtained when $\alpha \to \overline{\alpha}$.

When examining the three dimensional systems (Example 3 and Example 4) this apparent consistency disappears; the location of the best τ is no longer predictable nor is the shape of $\underline{\mu}(\alpha)$.

Figures 3 and 4 show how Algorithm 1 follows $\underline{\mu}(\alpha)$. In Table 1 we can see that Algorithm 1 finds the same τ as the brute force method with, on average, an error of 4×10^{-2}. As expected the LP problems generally return lower minimum average-dwell times, with the exception of Example 1. Note that the minimum of τ is also a function of K, i.e. the resolution of the triangulation used, and in

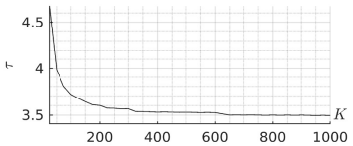


Fig. 5. The minimum average dwell-time τ obtained using the LP optimization problem (10) as a function of the resolution of the triangulation K used.

Figure 5 we depict this dependency for Example 1. For large K the minimum value of τ approaches the minimum value obtained with the LMI optimization problem, indicating that in this particular example quadratic Lyapunov functions are optimal.

Note that the brute force computations done for comparison are computationally very demanding and can only be done practically for small optimization problems. For example, in the Brute Force computations for Example 1 we solved 4,800 optimization problems, in comparison to 200 optimization problems when using Algorithm 1.

Table 1. Results of Algorithm 1 compared to the results from using a brute force approach and the method used in Hafstein and Tanwani (2023), denoted Method $(*)$.

Ex.	Method	Optimization	τ	α	μ
1	Brute Force		3.4180	0.1996	1.9960
	Algorithm 1	SDP	3.5195	0.1980	2.0074
	Method $(*)$		5.1929	N/A	2
	Brute Force	LP ($K = 300$)	3.5401	0.9790	1.4142
	Algorithm 1		3.5636	0.0860	1.3995
	Method $(*)$	LP ($K = 500$)	4.5283	N/A	1.4
2	Brute Force		1.1516	1.9990	9.9946
	Algorithm 1	SDP	1.1532	1.9800	9.8084
	Method $(*)$		17.039	N/A	3.1
	Brute Force		0	*	1
	Algorithm 1	LP ($K = 300$)	0	*	1
	Method $(*)$		0	*	1
3	Brute Force		1.1845	1.2440	4.3647
	Algorithm 1	SDP	1.1852	1.1400	3.8617
	Method $(*)$		4.6870		2.7
	Brute Force		0	*	1
	Algorithm 1	LP($K = 6$)	0	*	1
	Method $(*)$		0	*	1
4	Brute Force	SDP	7.9994	0.0405	1.3829
	Algorithm 1		8.0090	0.0428	1.4090
	Brute Force	LP($K = 25$)	2.9451	0.0146	1.0440
	Algorithm 1		3.0305	0.0059	1.0229

5. CONCLUSION

We described a new method to compute the minimum average dwell-time τ needed to assert the global exponential stability (GES) of the equilibrium at the origin for a switched linear system. For this, the method solves linear programming (LP) or linear matrix inequality (LMI) optimization problems as in Hafstein and Tanwani (2023), but searches for the optimal parameters (α, μ) directly, instead of fixing μ and maximizing α as in Hafstein and Tanwani (2023). We demonstrated our approach on four examples and made some interesting observations. In all cases we obtain lower values for τ than reported in the literature.

REFERENCES

Andersen, S., August, E., Hafstein, S., and Piccini, J. (2024). *Simulation and Modeling Methodologies, Technologies and Applications*, chapter Numerical Computations of Lyapunov Functions for Switched Linear Systems, 192–213. Springer.

Andersen, S., Giesl, P., and Hafstein, S. (2023). Common Lyapunov functions for switched linear systems: Linear programming-based approach. *IEEE Control Systems Letters*, 7, 901–906. doi:10.1109/LCSYS.2022.3228857.

Dayawansa, W. and Martin, C. (1999). A converse Lyapunov theorem for a class of dynamical systems which undergo switching. *IEEE Trans. Automat. Control*, (44), 751–760.

Giesl, P. and Hafstein, S. (2014). Revised CPA method to compute Lyapunov functions for nonlinear systems. *J. Math. Anal. Appl.*, 410, 292–306.

Gurobi Optimization, LLC (2024). Gurobi Optimizer Reference Manual. URL `https://www.gurobi.com`.

Hafstein, S. and Tanwani, A. (2023). Linear programming based lower bounds on average dwell-time via multiple Lyapunov functions. *European Journal of Control*, 74, 100838. doi:10.1016/j.ejcon.2023.100838. URL `https://laas.hal.science/hal-04161620`.

Hagood, J. and Thomson, B. (2006). Recovering a function from a Dini derivative. *The American Mathematical Monthly*, 113(1), 34–46.

Hespanha, J. and Morse, A. (1999). Stability of switched systems with average dwell-time. In *In Proc. 38th IEEE Conference on Decision and Control*, 2655–2660.

Liberzon, D. (2003). *Switching in systems and control*. Systems & Control: Foundations & Applications. Birkhäuser.

Liu, S., Martínez, S., and Cortés, J. (2021). Average dwell-time minimization of switched systems via sequential convex programming. *IEEE Contr. Syst. Lett.*, 6, 1076–1081.

Löfberg, J. (2004). YALMIP: A toolbox for modeling and optimization in MATLAB. In *In Proceedings of the CACSD Conference*. Taipei, Taiwan.

Sylvester, J. (1884). Sur l'equation en matrices px=xq. *C. R. Acad. Sci. Paris.*, 99(2), 67–71, 115–116.

The MathWorks Inc. (2023). MATLAB version: 23.2.0 (r2023b). URL `https://www.mathworks.com`.

Toh, K.C., Todd, M.J., and Tütüncü, R.H. (1999). SDPT3 - a matlab software package for semidefinite programming, version 1.3. *Optimization Methods and Software*, 11(1-4), 545–581. doi:10.1080/10556789908805762.

Paper IV

Constructing Piecewise Quadratic Lyapunov Functions with Linear Programming for Continuous-Time Switched and Conewise Linear Systems

J.J. Palacios Roman, S.F. Hafstein, P.A. Giesl, S.J.A.M. v.d. Eijnden, S. Andersen, W.P.M.H. Heemels

Automatica, Volume 186, Article 112838, April 2026.

This article is reproduced in full with permission from Elsevier.

In this paper, we present a method for constructing continuous piecewise quadratic (CPQ) Lyapunov functions for continuous-time switched and conewise linear systems using linear programming (LP). Key in our approach is the formulation of effective sufficient conditions for the copositivity of matrices via diagonal dominance. This formulation consists of linear constraints and can be expressed as an LP. It is shown that the sufficient conditions are also necessary conditions for the existence of a Lyapunov function, given a sufficiently refined CPQ function. We provide an in-depth comparison between our new method and other computational methods in the literature, and provide extensive numerical experiments on various switched and conewise linear systems. In particular, we show that the proposed method is the most accurate of the LP-based methods for constructing Lyapunov functions and is a numerically competitive alternative to LMI-based methods.

Automatica 186 (2026) 112838

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Constructing piecewise quadratic Lyapunov functions with linear programming for continuous-time switched and conewise linear systems ☆

Juan Javier Palacios Roman [a,*], Sigurdur Hafstein [b], Peter Giesl [c], Sebastiaan van den Eijnden [a], Stefania Andersen [b], W.P.M.H. Heemels [a]

[a] Department of Mechanical Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands
[b] Faculty of Physical Sciences, University of Iceland, Reykjavik, Iceland
[c] Department of Mathematics, University of Sussex, Brighton, UK

ARTICLE INFO

Article history:
Received 7 February 2025
Received in revised form 12 September 2025
Accepted 11 December 2025

Keywords:
Stability of hybrid systems
Lyapunov methods
Linear programming

ABSTRACT

In this paper, we present a method for constructing continuous piecewise quadratic (CPQ) Lyapunov functions for continuous-time switched and conewise linear systems using linear programming (LP). Key in our approach is the formulation of effective sufficient conditions for the copositivity of matrices via diagonal dominance. This formulation consists of linear constraints and can be expressed as an LP. It is shown that the sufficient conditions are also necessary conditions for the existence of a Lyapunov function, given a sufficiently refined CPQ function. We provide an in-depth comparison between our new method and other computational methods in the literature, and provide extensive numerical experiments on various switched and conewise linear systems. In particular, we show that the proposed method is the most accurate of the LP-based methods for constructing Lyapunov functions and is a numerically competitive alternative to LMI-based methods.

© 2026 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

1. Introduction

Switched linear systems (SLSs) and *conewise linear systems* (CLSs) represent a particular class of dynamical systems formed by a collection of linear subsystems that are combined with time and/or state-based switching rules (Liberzon, 2003). In recent years, SLSs and CLSs have become increasingly important within a wide range of applications, including power converters (Middlebrook & Cuk, 1976), traffic management (Lygeros et al., 1998), epidemiological modeling (Ait Rami et al., 2013) and high-performance motion control systems (Van den Eijnden et al., 2020; Van Loon et al., 2016). Assessing stability of SLSs and CLSs is a well-studied problem, and over the years many constructive tools for stability analysis have been presented, see, e.g., Lin and Antsaklis (2009) for an overview. Examples range from dwell-time approaches (Yuan & Wu, 2015; Zhao et al., 2012), to frequency-domain techniques (Kamenetskiy, 2017; Van den Eijnden et al., 2024) and Lyapunov-based methods (Branicky, 1998; Johansson & Rantzer, 1998; Jungers, 2024; Sun, 2010).

Of particular interest is the use of *continuous piecewise quadratic* (CPQ) Lyapunov functions (Johansson & Rantzer, 1998) for the stability analysis of continuous-time SLSs and CLSs. The specific interest in CPQ Lyapunov functions stems from the possibility to express stability tests in terms of convex optimization problems that can be solved efficiently (Boyd & Vandenberghe, 2004; Rantzer & Johansson, 2000), and from their success in reducing conservatism compared to earlier methods in the analysis of CLSs, see, e.g., the seminal works in Johansson and Rantzer (1998), Rantzer and Johansson (2000). The key idea underlying CPQ methods is to partition (part of) the state-space into smaller regions, to each of which a quadratic function is assigned that is continuous over the partition boundaries, and satisfies certain conditions locally. These conditions are typically expressed in terms of constrained inequalities that can be transformed into linear matrix inequalities (LMIs) by means of the so-called S-procedure (Yakubovich, 1977). For a partitioning of the state space with sufficiently small regions, these LMIs must admit a solution for an exponentially stable CLS. An alternative method that does not make explicit use of the S-procedure has been proposed in Iervolino et al. (2015a, 2015b), where stability conditions are

☆ The research leading to these results has received funding from the European Research Council under the Advanced ERC Grant Agreement PROACTHIS, no. 101055384. The material in this paper was not presented at any conference. This paper was recommended for publication in revised form by Associate Editor Philipp Braun under the direction of Editor Luca Zaccarian.

* Corresponding author.
E-mail addresses: j.j.palacios.roman@tue.nl (J.J. Palacios Roman), shafstein@hi.is (S. Hafstein), p.a.giesl@sussex.ac.uk (P. Giesl), s.j.a.m.v.d.eijnden@tue.nl (S.v.d. Eijnden), saa20@hi.is (S. Andersen), m.heemels@tue.nl (W.P.M.H. Heemels).

https://doi.org/10.1016/j.automatica.2026.112838
0005-1098/© 2026 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

expressed in terms of a copositive problem, i.e., the problem of verifying whether a certain matrix is copositive or, equivalently, has a corresponding quadratic form that is positive definite on the positive orthant (Hadeler, 1983). This problem, in turn, is translated into an LMI problem.

The aforementioned results hinge on the idea of reformulating constrained inequalities as LMIs. Although LMI conditions facilitate numerically tractable stability tests for SLSs and CLSs, they scale badly in terms of computational complexity with increasing system dimensions and the number of state-space partitions. These qualities motivate the development of alternative methods for constructing CPQ Lyapunov functions that preserve the structure and numerical tractability coming from LMI-based methods, but require less computational effort (even in case of a large number of regimes in the state-space partitioning).

Therefore, we present in this paper a new approach for constructing CPQ Lyapunov functions for SLSs with arbitrary switching and CLSs using linear programming (LP). Compared to solving large-scale LMI problems using semidefinite programming (SDP), solving large LP problems can be done more efficiently and reliably as a result of their significantly simpler structure, see, e.g., Boyd and Vandenberghe (2004), Vandenberghe and Boyd (1996) for further discussions. The use of LP for stability analysis of SLSs and CLSs has been proposed before: For example, LP can be used to construct continuous piecewise affine (CPA) Lyapunov functions (Andersen et al., 2023; Hafstein, 2002, 2023). However, it was shown in Palacios Roman et al. (2025) on a numerical example that CPA Lyapunov functions were less effective than CPQ Lyapunov functions, despite the computational benefit of LP over SDP, because CPA Lyapunov functions required a much larger number of piecewise segments. Moreover, in Bundfuss and Dür (2009), Iervolino et al. (2011), sufficient linear inequalities were used to prove the existence of a CPQ Lyapunov function by solving a copositive problem. However, these conditions are more conservative than the conditions we will present, which will be discussed in Section 6.3.

In line with the above, the main contributions of this paper are as follows. We present efficient conditions for the computation of CPQ Lyapunov functions for continuous-time SLSs and CLSs based on LP. Key in our approach is the reformulation of copositive matrix problems into (sufficient) tests that require verifying diagonal dominance of the involved matrices. These tests are subject to linear constraints that can be formulated as an LP problem. Our results apply to both SLSs with arbitrary switching and CLSs. For both of these classes we will show that for a sufficiently fine partitioning of the state space our tests become necessary conditions for exponential stability. We compare our method to other stability methods in the literature, including the use of CPQ and CPA Lyapunov functions, and provide extensive numerical experiments showcasing our method in comparison to other methods in the literature. In particular, we show theoretically and numerically that the proposed method is the least conservative of the LP-based methods, among Hafstein (2002) and Iervolino et al. (2011), given a fixed partitioning of the state-space and numerically that the proposed method is computationally competitive with the LMI-based method in Johansson and Rantzer (1998).

The remainder of this paper is structured as follows. We provide the necessary preliminaries in Section 2 such as the parameterization of CPQ Lyapunov functions. Section 3 will specify the systems under consideration and the stability analysis problem to be solved. In Section 4 we present our main results in the form of a method that exploits LP for computing CPQ Lyapunov functions for SLSs with arbitrary switching. In Section 5 we present similar results for CLSs. In Section 6 we provide an in-depth comparison of our LP-based method and other stability methods found in the literature, including computations of CPA functions through LP and CPQ functions through LMIs. Section 7 provides extensive numerical case studies on various systems demonstrating the benefits of our approach. We close the paper with conclusions and directions for future work in Section 8. Finally, supporting theorems are provided in the Appendix.

2. Preliminaries

In this paper, the subset of $\mathbb{R}$ with nonnegative value is denoted by $\mathbb{R}_{\geq 0}$. By $[A]_{ij}$ we denote the i,j-th entry of the matrix $A \in \mathbb{R}^{n\times n}$, $i,j \in \{1,2,\dots,n\}$. We say that a symmetric matrix $A = A^\top \in \mathbb{R}^{n\times n}$ is positive definite (negative definite), denoted by $A \succ 0$ ($A \prec 0$), if $x^\top Ax > 0$ ($x^\top Ax < 0$) for all $x \in \mathbb{R}^n \setminus \{\underline{0}\}$, where $\underline{0}$ denotes the origin of $\mathbb{R}^n$. Moreover, we say that a symmetric matrix $A = A^\top \in \mathbb{R}^{n\times n}$ is positive definite (negative definite) on the set $\mathcal{B} \subseteq \mathbb{R}^n$, denoted by $A \succ_{\mathcal{B}} 0$ ($A \prec_{\mathcal{B}} 0$), if $x^\top Ax > 0$ ($x^\top Ax < 0$) for all $x \in \mathcal{B} \setminus \{\underline{0}\}$. If $A \succ_{\mathcal{B}} 0$ and $\mathcal{B} = \mathbb{R}^n_{\geq 0}$, then A is said to be a copositive matrix. Finally, a matrix $A \in \mathbb{R}^{n\times n}$ is said to be strictly diagonally dominant if $2\cdot|[A]_{ii}| > \sum_{j=1}^n |[A]_{ij}|$ for all $i \in \{1,2,\dots,n\}$.

Inequalities for vectors hold componentwise. Furthermore, $\|x\| := \sqrt{\sum_{i=1}^n x_i^2}$ and $\|x\|_\infty := \max_{i=1,2,\dots,n}|x_i|$ denote the Euclidean norm and infinity norm of $x \in \mathbb{R}^n$, respectively, and $\|A\| := \max_{\|x\|=1}\|Ax\|$ denotes the matrix norm of $A \in \mathbb{R}^{n\times n}$ induced by the Euclidean norm.

2.1. Triangulations

A triangulation $\mathcal{T} = \{\mathcal{S}_1, \mathcal{S}_2, \dots, \mathcal{S}_N\}$, $N \in \mathbb{N}$, is a collection of simplices $\mathcal{S}_i$, $i \in \{1,2,\dots,N\}$, defined as

$$\mathcal{S}_i = \operatorname{co}(r_0^i, r_1^i, \dots, r_n^i) := \left\{ \sum_{k=0}^n \lambda_k r_k^i \;\middle|\; \sum_{k=0}^n \lambda_k = 1, \lambda_k \geq 0, k \in \{0,1,\dots,n\} \right\},$$

where $\operatorname{co}(\cdot)$ denotes the convex hull and $r_0^i, r_1^i, \dots, r_n^i \in \mathbb{R}^n$ are affinely independent vertices. The affine independence of the vertices implies that each simplex $\mathcal{S}_i$, $i \in \{1,2,\dots,N\}$, has a non-empty interior. Furthermore, $\{v_0, v_1, \dots, v_p\} \subset \mathbb{R}^n$, $p \in \mathbb{N}$, denotes the collection of all vertices of the simplices used in the triangulation, such that $\{r_0^i, r_1^i, \dots, r_n^i\} \subset \{v_0, v_1, \dots, v_p\}$ for all $i \in \{1,2,\dots,N\}$. Clearly, for all $i \in \{1,2,\dots,N\}$, there exists then an index function $g_i : \{0,1,\dots,n\} \to \{0,1,\dots,p\}$ such that $r_k^i = v_{g_i(k)}$ for all $k \in \{0,1,\dots,n\}$.

We require that any two simplices $\mathcal{S}_i$ and $\mathcal{S}_j$, $i,j \in \{1,2,\dots,N\}$ and $i \neq j$, either have an empty intersection or the intersection is a polyhedron described by the convex hull of the common vertices of the simplices $\mathcal{S}_i$ and $\mathcal{S}_j$. We further require that $\bigcup_{i=1}^N \mathcal{S}_i$ is a neighborhood of the origin and that $r_0^i = v_0 = \underline{0}$ for all $i \in \{1,2,\dots,N\}$. Then $r_1^i, r_2^i, \dots, r_n^i$ are linearly independent for all $i \in \{1,2,\dots,N\}$. Examples of triangulations that satisfy these conditions are the triangular fans $K^{-1}\mathcal{T}_K$ and $\mathcal{T}_K$ from Hafstein (2023). Here, $K \in \mathbb{N}$ is a measure of how refined the triangulation is, namely for all simplices $\mathcal{S}_i$, $i \in \{1,2,\dots,N\}$, in $K^{-1}\mathcal{T}_K$ we have that $\|r_k^i - r_\ell^i\|_2 \leq \sqrt{n}/K$ for all $k,\ell \in \{1,2,\dots,n\}$. Furthermore, $N = 2^n \cdot K^{n-1} \cdot n!$ and $\{v_1, v_2, \dots, v_p\} = \{K^{-1}z \mid z \in \mathbb{Z}^n,\ \|z\|_\infty = K\}$. The triangulation $\mathcal{T}_K$ is a scaled version of $K^{-1}\mathcal{T}_K$ which we will use in Section 7. For further details, see Hafstein (2023).

2.2. Parameterization of CPQ functions

Given a triangulation $\mathcal{T}$ satisfying all conditions and assumptions above, we can associate a simplicial cone $\mathcal{C}_i$ to each simplex

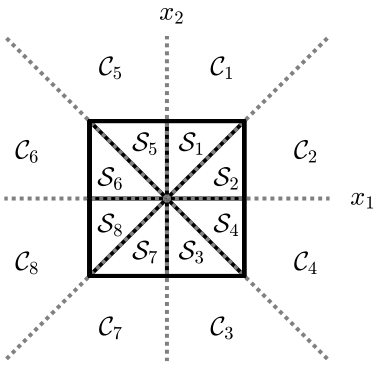


Fig. 1. Visualization of the conic subdivision $\{\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_N\}$ that results from the triangulation $\mathcal{T}_1$ of a two-dimensional space (x_1, x_2). The boundaries of the simplices $\{\mathcal{S}_1, \mathcal{S}_2, \dots, \mathcal{S}_N\}$ of the triangulation $\mathcal{T}_1$ are indicated with black lines and the boundaries of the cones $\{\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_N\}$ with gray dotted lines. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

$\mathcal{S}_i$, $i \in \{1, 2, \dots, N\}$, defined as

$$\begin{aligned}\mathcal{C}_i &= \operatorname{pos}(r_1^i, r_2^i, \dots, r_n^i)\\ &:= \left\{ \sum_{k=1}^{n} \lambda_k r_k^i \;\middle|\; \lambda_k \geq 0, k \in \{1, 2, \dots, n\} \right\},\end{aligned} \tag{1}$$

where $\operatorname{pos}(\cdot)$ denotes the positive hull. Since $\mathcal{T}$ is a triangulation of a neighborhood of the origin, the union of all $\mathcal{C}_i$ is equal to $\mathbb{R}^n$. Note that the triangulations $\mathcal{T}_K$ and $K^{-1}\mathcal{T}_K$ from Hafstein (2023) give the same conical subdivision of $\mathbb{R}^n$. An example of such a conical subdivision is given in Fig. 1.

Then, a CPQ function $V : \mathbb{R}^n \to \mathbb{R}$ can be defined as

$$V(x) = x^\top P_i x \qquad \text{if } x \in \mathcal{C}_i. \tag{2}$$

The matrices $P_i = P_i^\top \in \mathbb{R}^{n\times n}$ are parameterized by a matrix $\Phi = \Phi^\top \in \mathbb{R}^{p\times p}$ as follows:

$$\begin{aligned}[\Psi_i]_{k\ell} &= [\Phi]_{g_i(k)g_i(\ell)} && \forall k, \ell \in \{1, 2, \dots, n\}\\ R_i &= \begin{bmatrix} r_1^i & r_2^i & \dots & r_n^i \end{bmatrix} && \in \mathbb{R}^{n\times n}\\ E_i &= R_i^{-1} && \in \mathbb{R}^{n\times n}\\ P_i &= E_i^\top \Psi_i E_i && \in \mathbb{R}^{n\times n}.\end{aligned} \tag{3}$$

Note that vertices $r_1^i, r_2^i, \dots, r_n^i$ are linearly independent and thus R_i is invertible and we assume that the order of the vertices $r_1^i, r_2^i, \dots, r_n^i$ is fixed, in accordance with the definition of g_i in Section 2.1, so that R_i is uniquely defined. Note that if $x \in \mathcal{C}_i$, then $x = R_i\lambda$, where $\lambda := \begin{bmatrix}\lambda_1 & \lambda_2 & \dots & \lambda_n\end{bmatrix}^\top \in \mathbb{R}^n_{\geq 0}$ is the vector with the λ_k in (1) as its entries, and thus

$$\begin{aligned}V(x) &= x^\top P_i x = \lambda^\top R_i^\top P_i R_i \lambda\\ &= \lambda^\top R_i^\top E_i^\top \Psi_i E_i R_i \lambda = \lambda^\top \Psi_i \lambda\\ &= \sum_{k=1}^{n}\sum_{\ell=1}^{n} \lambda_k\lambda_\ell [\Psi_i]_{k\ell} = \sum_{k=1}^{n}\sum_{\ell=1}^{n} \lambda_k\lambda_\ell [\Phi]_{g_i(k)g_i(\ell)}.\end{aligned} \tag{4}$$

Remark 1. This method of parameterizing V was suggested in Rantzer and Johansson (2000). It can be interpreted as follows: The value of $V(v_m) = [\Phi]_{mm}$ for all $m \in \{1, 2, \dots, p\}$. Moreover, $V(r_k^i) = [\Phi]_{g_i(k)g_i(k)}$ and

$$\begin{aligned}V\left(\frac{1}{2}r_k^i + \frac{1}{2}r_\ell^i\right) =& \frac{1}{4}[\Phi]_{g_i(k)g_i(k)} + \frac{1}{4}[\Phi]_{g_i(k)g_i(\ell)}\\ &+ \frac{1}{4}[\Phi]_{g_i(\ell)g_i(k)} + \frac{1}{4}[\Phi]_{g_i(\ell)g_i(\ell)}\end{aligned} \tag{5}$$

for all $i \in \{1, 2, \dots, N\}$ and $k, \ell \in \{1, 2, \dots, n\}$.

By construction, V will be well-defined and continuous. This is obvious for the case when x is in the interior of $\mathcal{C}_i$. In case $x \in \mathcal{C}_i \cap \mathcal{C}_j$, where $i, j \in \{1, 2, \dots, N\}$ and $i \neq j$, then x is a linear combination of the common (extreme) rays of the cones $\mathcal{C}_i$ and $\mathcal{C}_j$, denoted by $\{y_1, y_2, \dots, y_m\} := \{r_1^i, r_2^i, \dots, r_n^i\} \cap \{r_1^j, r_2^j, \dots, r_n^j\} \subset \{v_1, v_2, \dots, v_p\}$. In other words,

$$x = \sum_{k=1}^{m} \lambda_k y_k = \sum_{k=1}^{m} \lambda_k r_{s_k}^i = \sum_{k=1}^{m} \lambda_k r_{t_k}^i$$

where $0 < m < n$ and $y_k = r_{s_k}^i = r_{t_k}^j$ for some $s_k, t_k \in \{1, 2, \dots, n\}$ for all $k \in \{1, 2, \dots, m\}$. Thus, $g_i(s_k) = g_j(t_k)$ and by using (4) we can show that for all $x \in \mathcal{C}_i \cap \mathcal{C}_j$

$$\begin{aligned}V(x) = x^\top P_i x &= \sum_{k=1}^{m}\sum_{\ell=1}^{m} \lambda_k\lambda_\ell [\Phi]_{g_i(s_k)g_i(s_\ell)}\\ &= \sum_{k=1}^{m}\sum_{\ell=1}^{m} \lambda_k\lambda_\ell [\Phi]_{g_j(t_k)g_j(t_\ell)} = x^\top P_j x,\end{aligned}$$

which shows that V is continuous and well-defined.

3. System classes and problem statement

In this section we will introduce the classes of systems under consideration and relevant stability theorems. Finally, the main problem statement will be specified.

3.1. Switched linear systems with arbitrary switching

Consider the continuous-time SLS

$$\dot{x}(t) = A_{\sigma(t)}x(t), \tag{6}$$

where $x(t) \in \mathbb{R}^n$ is the state at time $t \in \mathbb{R}_{\geq 0}$, $\sigma : \mathbb{R}_{\geq 0} \to \{1, 2, \dots, M\}$ is a piecewise constant function that is right-continuous and only has a finite number of discontinuity points on every finite time-interval, called the switching signal and $A_j \in \mathbb{R}^{n\times n}$ for all $j \in \{1, 2, \dots, M\}$. A solution $x : \mathbb{R}_{\geq 0} \to \mathbb{R}^n$ to (6) is understood in the sense of Carathéodory (Walter, 1998).

We use the following definition and theorem adapted from Liberzon (2003) and Heemels and Weiland (2008).

Definition 2. The SLS (6) is said to be globally uniformly exponentially stable (GUES), if there exist constants $c, \lambda > 0$ such that for all switching signals σ, i.e., under arbitrary switching, all solutions x of (6) satisfy

$$\|x(t)\| \leq c\|x(0)\|e^{-\lambda t}$$

for all $t \in \mathbb{R}_{\geq 0}$.

Theorem 3. *The SLS (6) is GUES, if there exists a common CPQ Lyapunov function $V : \mathbb{R}^n \to \mathbb{R}$ as in (2) that satisfies for all $i \in \{1, 2, \dots, N\}$ and $j \in \{1, 2, \dots, M\}$*

$$c_1\|x\|^2 \leq x^\top P_i x \leq c_2\|x\|^2 \qquad \forall x \in \mathcal{C}_i \tag{7a}$$

$$x^\top(A_j^\top P_i + P_i A_j)x \leq -c_3\|x\|^2 \qquad \forall x \in \mathcal{C}_i \tag{7b}$$

for some $c_1, c_2, c_3 > 0$.

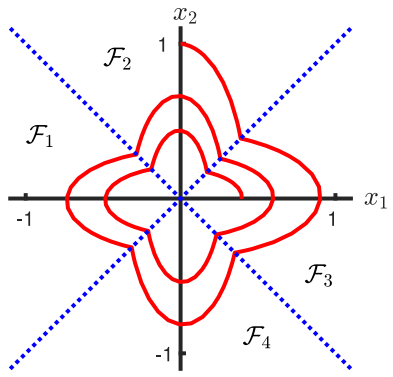


Fig. 2. Example of a CLS taken from Johansson (1999), where $M = 4$ and $n = 2$. The definition of the system is given in (47)–(49). The boundaries of the sets $\mathcal{F}_1$, $\mathcal{F}_2$, $\mathcal{F}_3$ and $\mathcal{F}_4$ are indicated with blue dotted lines and a sample trajectory starting at $(x_1, x_2) = (0, 1)$ with red. When the state enters a new set, the differential equations governing the dynamics change. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Note that the cones $\mathcal{C}_i$ are not related to the system modes A_j. Instead, the cones $\mathcal{C}_i$ are a construct for building the CPQ Lyapunov function V only, which must hold for any active system mode across the entire state-space.

3.2. Conewise linear systems

Consider the continuous-time CLS Camlibel et al. (2006)

$$\dot{x}(t) = A_j x(t) \qquad \text{if } x(t) \in \mathcal{F}_j, \tag{8}$$

where $x(t) \in \mathbb{R}^n$ is the state at time $t \in \mathbb{R}_{\geq 0}$ and $A_j \in \mathbb{R}^{n\times n}$ and $\mathcal{F}_j \subset \mathbb{R}^n$ for all $j \in \{1, 2, \dots, M\}$. We assume that the union of the subsets $\mathcal{F}_j$ is equal to $\mathbb{R}^n$, the intersection of the interiors of $\mathcal{F}_i$ and $\mathcal{F}_j$ is empty for all $i, j \in \{1, 2, \dots, M\}$, $i \neq j$, and the closure of $\mathcal{F}_j$ is a simplicial cone for all $j \in \{1, 2, \dots, M\}$. These systems are also referred to as switched linear systems with state-dependent switching and are a subclass of piecewise linear systems. Solutions to (8) are understood in the sense of Filippov, i.e., are solutions to the differential inclusion

$$\dot{x}(t) \in F(x(t)) := \operatorname{co}\left\{\bigcup_{j=1}^{M} \{A_j x(t) \mid x(t) \in \bar{\mathcal{F}}_j\}\right\}, \tag{9}$$

where $\bar{\mathcal{F}}_j$ is the closure of the set $\mathcal{F}_j$. Thus, potential sliding modes are taken into account, see, e.g., Aubin and Cellina (1984), Clarke et al. (1998), Filippov (1988). An example of a CLS is given in Fig. 2.

We use the following definition and theorem adapted from Heemels and Weiland (2008) and Kellett (2015).

Definition 4. The CLS (8) is said to be globally exponentially stable (GES), if there exist constants $c, \lambda > 0$ such that for all Filippov solutions x to (8), i.e., for all solutions to (9), it holds that

$$\|x(t)\| \leq c\|x(0)\|e^{-\lambda t}$$

for all $t \in \mathbb{R}_{\geq 0}$.

Theorem 5. *The CLS (8) is GES, if there exists a CPQ Lyapunov function $V : \mathbb{R}^n \to \mathbb{R}$ as in (2) that satisfies for all $i \in \{1, 2, \dots, N\}$ and $j \in \{1, 2, \dots, M\}$*

$$c_1\|x\|^2 \leq x^\top P_i x \leq c_2\|x\|^2 \qquad \forall x \in \mathcal{C}_i \tag{10a}$$

$$x^\top(A_j^\top P_i + P_i A_j)x \leq -c_3\|x\|^2 \qquad \forall x \in \mathcal{C}_i \cap \bar{\mathcal{F}}_j, \tag{10b}$$

for some $c_1, c_2, c_3 > 0$.

3.3. Problem definition

Our goal is to find a CPQ function V that satisfies condition (7b) for an SLS or conditions (10b) for a CLS, i.e., a CPQ function V such that $V(x(t))$ is decreasing over time along all solutions x of the system, except for the zero-solution. Then:

(i) if the resulting function V additionally satisfies (7a) or (10a), i.e., if V is positive definite, then the SLS or CLS is GUES or GES, respectively;
(ii) if the resulting function V does not satisfy (7a) or (10a), i.e., if V is not positive definite, then the SLS or CLS is unstable.

The first statement follows from Theorem 3 or 5. The second statement follows from Lemma 4.1 in Johansson (1999) and a sketch of proof is provided here for completeness: Suppose there exists a CPQ function V that, given a certain SLS, satisfies (7b), but does not satisfy (7a). Then, there exists a state $\hat{x}(0) \neq \underline{0}$ such that $V(\hat{x}(0)) \leq 0$. Consider the solution $\hat{x}(t)$ that starts at $\hat{x}(0)$. Because V satisfies (7b), V is decreasing over time along all solutions of the system, except for the zero-solution. Thus, after a time step δ, it will hold that $V(\hat{x}(\delta)) < 0$. In fact, as $t \to \infty$, $V(\hat{x}(t)) \to -\infty$, which means $\|\hat{x}(t)\| \to \infty$ as $t \to \infty$. This implies instability of the SLS. A similar argument can be made for a CLS.

This method is known as the two-step procedure (Johansson, 1999, Proposition 4.4), where the first step is finding a function V that is decreasing over time along all solutions of the system except for the zero solution, and the second step is checking whether V is positive definite. This reduces the size of an optimization problem to find a suitable V, since less constraints have to be taken into account.

4. CPQ Lyapunov functions for SLSs

In this section we will show sufficient conditions in the form of linear inequalities under which a CPQ function is a Lyapunov function for an SLS (6). Then, a converse theorem will be presented that shows that for a sufficiently refined CPQ function the conditions become necessary, when the system is GUES.

4.1. Sufficient conditions for CPQ Lyapunov functions of SLSs

The CPQ function V is a Lyapunov function to the SLS (6), i.e., satisfies (7a) and (7b), if

$$P_i \succ_{\mathcal{C}_i} 0 \tag{11a}$$

$$Q_{ij} \prec_{\mathcal{C}_i} 0 \tag{11b}$$

for all $i \in \{1, 2, \dots, N\}$ and $j \in \{1, 2, \dots, M\}$, where

$$Q_{ij} := A_j^\top P_i + P_i A_j. \tag{12}$$

For the first step of the two-step procedure in Section 3.3, a CPQ function V that satisfies (11b) needs to be constructed. We wish to do this with an LP problem. Thereto, consider the following theorem.

Theorem 6. *Consider a CPQ function V, parameterized as in (2), and the SLS (6). Define, for all $i \in \{1, 2, \dots, N\}$ and $j \in \{1, 2, \dots, M\}$, the matrices*

$$B_{ij} = R_i^\top Q_{ij} R_i, \tag{13}$$

with Q_{ij} as in (12), and $C_{ij} \in \mathbb{R}^{n\times n}$ such that

$$[C_{ij}]_{kk} = [B_{ij}]_{kk} \qquad \forall k \in \{1, 2, \dots, n\} \tag{14}$$

$$[C_{ij}]_{k\ell} = \max([B_{ij}]_{k\ell}, 0) \qquad \forall k, \ell \in \{1, 2, \dots, n\}, k \neq \ell. \tag{15}$$

If for all $i \in \{1, 2, \dots, N\}$ and $j \in \{1, 2, \dots, M\}$, the sum of each row of C_{ij} is negative, i.e.,

$$\sum_{\ell=1}^{n} [C_{ij}]_{k\ell} < 0 \quad \forall k \in \{1, 2, \dots, n\}, \tag{16}$$

then V satisfies condition (11b).

Proof. First, we prove that (15) and (16) imply that C_{ij} is a negative definite matrix, i.e., a matrix with all eigenvalues strictly negative. Note that C_{ij} is a real symmetric matrix, and therefore its eigenvalues are real. Then, Gerschgorin's circle theorem (Gerschgorin, 1931) states that all eigenvalues of C_{ij} must lie in the set

$$\bigcup_{k\in\{1,2,\dots,n\}} \{a \in \mathbb{R} \mid [C_{ij}]_{kk} - \sum_{\substack{\ell=1\\ \ell\neq k}}^{n} |[C_{ij}]_{k\ell}| \leq a \leq [C_{ij}]_{kk} + \sum_{\substack{\ell=1\\ \ell\neq k}}^{n} |[C_{ij}]_{k\ell}|\}. \tag{17}$$

Due to (15), the off-diagonal entries of C_{ij} are all non-negative. Therefore,

$$[C_{ij}]_{kk} + \sum_{\substack{\ell=1\\ \ell\neq k}}^{n} |[C_{ij}]_{k\ell}| = \sum_{\ell=1}^{n} [C_{ij}]_{k\ell}.$$

Thus, (16) implies that the set (17) only contains negative numbers, which means C_{ij} must be a negative definite matrix.

Secondly, we show that $\lambda^\top B_{ij}\lambda \leq \lambda^\top C_{ij}\lambda < 0$ for all $\lambda \in \mathbb{R}^n_{\geq 0} \setminus \{\underline{0}\}$. The second inequality holds because C_{ij} is a negative definite matrix. The first inequality follows from

$$\begin{aligned}
\lambda^\top C_{ij}\lambda &= \sum_{k=1}^{n} \lambda_k^2 [C_{ij}]_{kk} + \sum_{k=1}^{n}\sum_{\substack{\ell=1\\ \ell\neq k}}^{n} \lambda_k\lambda_\ell [C_{ij}]_{k\ell} \\
&= \sum_{k=1}^{n} \lambda_k^2 [B_{ij}]_{kk} + \sum_{k=1}^{n}\sum_{\substack{\ell=1\\ \ell\neq k}}^{n} \lambda_k\lambda_\ell \max([B_{ij}]_{k\ell}, 0) \\
&\geq \sum_{k=1}^{n} \lambda_k^2 [B_{ij}]_{kk} + \sum_{k=1}^{n}\sum_{\substack{\ell=1\\ \ell\neq k}}^{n} \lambda_k\lambda_\ell [B_{ij}]_{k\ell} \\
&= \lambda^\top B_{ij}\lambda.
\end{aligned}$$

Finally, we show that $\lambda^\top B_{ij}\lambda < 0$ for all $\lambda \in \mathbb{R}^n_{\geq 0} \setminus \{\underline{0}\}$ implies that $Q_{ij} \prec_{\mathcal{C}_i} 0$. Note that (1) can equivalently be expressed as

$$\mathcal{C}_i = \left\{ x \in \mathbb{R}^n \mid E_i x = \lambda \in \mathbb{R}^n_{\geq 0} \right\},$$

where E_i is defined in (3). Then, if we substitute λ with $E_i x$, B_{ij} with (13) and R_i with E_i^{-1}, we obtain

$$\begin{aligned}
& \lambda^\top B_{ij}\lambda < 0 && \forall \lambda \in \mathbb{R}^n_{\geq 0} \setminus \{\underline{0}\} \\
\implies \quad & x^\top E_i^\top (E_i^{-1})^\top Q_{ij} E_i^{-1} E_i x < 0 && \forall E_i x \in \mathbb{R}^n_{\geq 0} \setminus \{\underline{0}\} \\
\implies \quad & x^\top Q_{ij} x < 0 && \forall x \in \mathcal{C}_i \setminus \{\underline{0}\},
\end{aligned}$$

which completes the proof. □

The theorem states conditions under which Φ parameterizes a common CPQ Lyapunov function candidate V that satisfies (11b). Such a matrix Φ can be found by solving the following feasibility problem:

$$\begin{aligned}
&\text{find } \Phi, C_{ij} && && (18)\\
&\text{s.t. } P_i = E_i^\top [\Phi]_{g_i(k)g_i(\ell)} E_i && \forall i, k, \ell \\
&\quad B_{ij} = R_i^\top (A_j^\top P_i + P_i A_j) R_i && \forall i, j \\
&\quad [C_{ij}]_{kk} = [B_{ij}]_{kk} && \forall i, j, k \\
&\quad [C_{ij}]_{k\ell} \geq [B_{ij}]_{k\ell} && \forall i, j, k, \ell, k \neq \ell && (19)\\
&\quad [C_{ij}]_{k\ell} \geq 0 && \forall i, j, k, \ell, k \neq \ell && (20)\\
&\quad \sum_{\ell=1}^{n} [C_{ij}]_{k\ell} < 0 && \forall i, j, k,
\end{aligned}$$

where E_i, g_i and R_i depend on the triangulation $\mathcal{T}$. Note that the nonlinear constraints (15) were replaced by the linear constraints (19) and (20) and that the matrices C_{ij} are being treated as slack variables in (18). This is so that (18) becomes an LP feasibility problem.

Remark 7. The second step of the two-step procedure in Section 3.3, i.e., the verification that V satisfies (11a), can be done with some conservatism by checking whether there exist $U_i = U_i^\top \in \mathbb{R}^{n\times n}_{\succeq 0}$ such that

$$P_i - E_i^\top U_i E_i \succ 0 \qquad \forall i \in \{1, 2, \dots, N\}, \tag{21}$$

where the S-procedure (Yakubovich, 1977) is used to validate (11a). This condition can be checked with an LMI solver. The second step of the two-step procedure can also be completed by verifying that

$$[R_i^\top P_i R_i]_{k\ell} > 0 \qquad \forall i, k, \ell, \tag{22}$$

since then

$$\begin{aligned}
V(x) &= x^\top P_i x && \forall x \in \mathcal{C}_i \\
&= \lambda^\top R_i^\top P_i R_i \lambda && \forall \lambda \in \mathbb{R}^n_{\geq 0} \\
&= \sum_{k=1}^{n}\sum_{\ell=1}^{n} \lambda_k\lambda_\ell [R_i^\top P_i R_i]_{k\ell} \geq 0 && \forall \lambda \in \mathbb{R}^n_{\geq 0}.
\end{aligned}$$

Conditions (22) are more conservative than conditions (21). However, conditions (22) can be verified with an LP solver and could even be included in the LP problem (18), if we wish to skip the two-step procedure and construct a Lyapunov function in one step.

Remark 8. To ensure that V satisfies (11b), the matrices $-B_{ij}$ are forced to be strictly copositive, i.e., $-\lambda^\top B_{ij}\lambda > 0$ for all $\lambda \in \mathbb{R}^n_{\geq 0} \setminus \{\underline{0}\}$. The construction of and the conditions on the matrices C_{ij} are all for the purpose of asserting this. The use of strictly copositive matrices for the analysis of SLSs and CLSs has been suggested before, see, e.g., Bundfuss and Dür (2009), Camlibel et al. (2007), Iervolino et al. (2011, 2015a, 2015b), Pogromsky et al. (1998).

4.2. Converse theorem

To prove a converse theorem, i.e., that our method always succeeds in finding a Lyapunov function to a GUES system (6), we largely follow the proof of Hafstein (2023, Thm. 1), where a similar converse theorem is proved for CPA Lyapunov functions for SLSs with arbitrary switching.

Theorem 9. *Assume that the system (6) is GUES. Then, when using the triangulation $K^{-1}\mathcal{T}_K$, the LP problem (18) is feasible for all large enough $K \in \mathbb{N}$.*

Proof. To prove the feasibility of the LP problem (18), we assign values to all the variables and then show that the constraints of the problem are fulfilled when the variables have these values.

In Hafstein (2023, Lemma 2) it is shown that, if (6) is GUES, there exists a Lyapunov function V that:

- is infinitely differentiable, except at $x = \underline{0}$;
- is positively homogeneous of degree one;
- and satisfies for all x and some $c_1, c_2, c_3 > 0$

$$c_1\|x\| \le V(x) \le c_2\|x\|$$
$$\nabla V(x)A_j x \le -c_3\|x\| \quad \forall j \in \{1, 2, \dots, M\}.$$

If we define $W(x) = V(x)^2$, then W is a Lyapunov function for (6) as well that:

- is infinitely differentiable, except at $x = \underline{0}$;
- is positively homogeneous of degree two, i.e.,

$$W(cx) = c^2 W(x) \quad \text{and} \quad W(x) = \frac{1}{2}\nabla W(x)x. \tag{23}$$

  for all $x \neq \underline{0}$ and $c > 0$. The second statement follows from Euler's homogeneous function theorem (Apostol, 1969);
- and satisfies for all x and some $c_1, c_2, c_3 > 0$

$$c_1\|x\|^2 \le W(x) \le c_2\|x\|^2$$
$$\nabla W(x)A_j x \le -c_3\|x\|^2 \quad \forall j \in \{1, 2, \dots, M\}.$$

We assign values to the variables of the LP problem by defining for every simplex $\mathcal{S}_i$ the symmetric matrix P_i through

$$(r_k^i + r_\ell^i)^\top P_i (r_k^i + r_\ell^i) = W(r_k^i + r_\ell^i) \tag{24}$$

for all $k, \ell \in \{1, 2, \dots, n\}$. That the resulting CPQ function (2) is still continuous and well-defined follows from Giesl et al. (2025, Thm. 2.8). Define the functions $V_i : \mathbb{R}^n \to \mathbb{R}$ through $V_i(x) = x^\top P_i x$ and set, as before, $V(x) = V_i(x)$, if $x \in \mathcal{C}_i$. Then $V(r_k^i + r_\ell^i) = W(r_k^i + r_\ell^i)$ for all i, k, ℓ and we have from (24) that

$$2(r_k^i)^\top P_i r_\ell^i = W(r_k^i + r_\ell^i) - W(r_k^i) - W(r_\ell^i). \tag{25}$$

The values of the elements of Φ, i.e., the variables of the LP problem, are now obtained from the values of $V(r_k^i/2 + r_\ell^i/2)$ by using (5).

Define the constants

$$\bar{a} := \max_{j=1,2,\dots,N} \|A_j\|, \tag{26}$$

$$\bar{g} := \max_{\|x\|_\infty = 1} \|\nabla W(x)\|, \text{ and} \tag{27}$$

$$\bar{h} := \max_{\|x\|_\infty = 1} \|H_W(x)\|, \tag{28}$$

where H_W is the Hessian matrix of W. We will show that all the entries of all the matrices B_{ij} in the LP problem (18) are negative, if

$$K > \frac{\sqrt{n}\bar{a}\left(\bar{g} + 9\sqrt{3}n^2\bar{h}\right)}{c_3}, \tag{29}$$

where $c_3 > 0$ is the constant from (7b). Then, a feasible solution is given by Φ and by setting $[C_{ij}]_{kk} = [B_{ij}]_{kk} < 0$ and $[C_{ij}]_{k\ell} = 0$, $k \neq \ell$, for all i, j, k, ℓ.

Let us first derive an upper bound on the difference between ∇W and ∇V_i at the vertices r_k^i. Let $\mathcal{S}_i \in K^{-1}\mathcal{T}_K$ be an arbitrary, but fixed simplex. Following the proof of Hafstein (2023, Thm. 1), we have

$$\|r_k^i\|_\infty = 1 \quad \text{and} \quad \|r_k^i - r_\ell^i\| \le \frac{\sqrt{n}}{K} \tag{30}$$

for all $k, \ell \in \{1, 2, \dots, n\}$ and

$$\|E_i\| \le K\sqrt{3(n+1)}. \tag{31}$$

Now, since $\nabla V_i(r_k^i) = 2(r_k^i)^\top P_i$ and $E_i = R_i^{-1}$, we have

$$\begin{aligned}\|\nabla V(r_k^i) - \nabla W(r_k^i)\| &= \|2(r_k^i)^\top P_i - \nabla W(r_k^i)\| \\ &\le \|2(r_k^i)^\top P_i R_i - \nabla W(r_k^i)R_i\|\|E_i\|.\end{aligned} \tag{32}$$

Consider the ℓ-th component

$$y_\ell = 2(r_k^i)^\top P_i r_\ell^i - \nabla W(r_k^i) r_\ell^i \tag{33}$$

of the vector $y := 2(r_k^i)^\top P_i R_i - \nabla W(r_k^i)R_i$.

The case $\ell = k$ is simple: By (25) and (23) we have

$$y_\ell = 2W(r_\ell^i) - 2W(r_\ell^i) = 0 \quad \text{when } \ell = k.$$

To tackle the case $\ell \neq k$, consider first that by Taylor's theorem (Apostol, 1969) we have

$$\begin{aligned}W(r_\ell^i) &= W(r_k^i) + \nabla W(r_k^i)(r_\ell^i - r_k^i) \\ &\quad + \frac{1}{2}(r_\ell^i - r_k^i)^\top H_W(x_{k,\ell})(r_\ell^i - r_k^i),\end{aligned}$$

where $x_{k,\ell}$ is a point on the line segment between r_k^i and r_ℓ^i. Hence, there is a constant $\bar{h}_1$, $|\bar{h}_1| \le \bar{h}$, such that

$$\begin{aligned}\nabla W(r_k^i)r_\ell^i &= W(r_\ell^i) - W(r_k^i) + \nabla W(r_k^i)r_k^i - \frac{n\bar{h}_1}{2K^2} \\ &= W(r_\ell^i) + W(r_k^i) - \frac{n\bar{h}_1}{2K^2}\end{aligned}$$

by (30) and (23). Hence, formula (33) can be written as

$$\begin{aligned}y_\ell &= W(r_k^i + r_\ell^i) - 2W(r_k^i) - 2W(r_\ell^i) + \frac{n\bar{h}_1}{2K^2} \\ &= 4\left[W\left(\frac{1}{2}r_k^i + \frac{1}{2}r_\ell^i\right) - \frac{1}{2}W(r_k^i) - \frac{1}{2}W(r_\ell^i)\right] + \frac{n\bar{h}_1}{2K^2}\end{aligned}$$

using (25) and (23). By Baier et al. (2012, Prop. 4.1), (28) and (30) we get

$$\left|W\left(\frac{1}{2}r_k^i + \frac{1}{2}r_\ell^i\right) - \frac{1}{2}W(r_k^i) - \frac{1}{2}W(r_\ell^i)\right| \le \frac{n\bar{h}}{K^2},$$

hence

$$|y_\ell| \le \frac{4n\bar{h}}{K^2} + \frac{n|\bar{h}_1|}{2K^2} \le \frac{9n\bar{h}}{2K^2} \quad \text{when } \ell \neq k.$$

Combining cases $\ell = k$ and $\ell \neq k$ delivers

$$\|y\| \le \frac{9n\bar{h}\sqrt{n-1}}{2K^2}$$

and, by (32) and (31),

$$\|\nabla V(r_k^i) - \nabla W(r_k^i)\| \le \|y\|\|E_i\| \le \frac{9\sqrt{3}n^2\bar{h}}{2K}, \tag{34}$$

from which also

$$\|\nabla V(r_k^i)\| \leq \bar{g} + \frac{9\sqrt{3}n^2\bar{h}}{2K} \tag{35}$$

follows, see (27).

Consider the matrix $B_{ij} = R_i^\top(A_j^\top P_i + P_i A_j)R_i$ from the LP problem (40). Its k,ℓ-th element is given by

$$[B_{ij}]_{k\ell} = 2(r_k^i)^\top P_i A_j r_\ell^i = \nabla V_i(r_k^i)A_j r_\ell^i$$

and by (7b), (34), $\|x\|_\infty \leq \|x\| \leq \sqrt{n}\|x\|_\infty$, (35), and (30) we have

$$\begin{aligned}
&\nabla V_i(r_k^i)A_j r_\ell^i \\
&= \nabla W(r_k^i)A_j r_k^i + [\nabla V_i(r_k^i) - \nabla W(r_k^i)]A_j r_k^i \\
&\qquad + \nabla V_i(r_k^i)A_j(r_\ell^i - r_k^i) \\
&\leq -c_3\|r_k^i\|^2 \\
&\qquad + \|\nabla V_i(r_k^i) - \nabla W(r_k^i)\|\|A_j\|\|r_k^i\| \\
&\qquad\quad + \|\nabla V_i(r_k^i)\|\|A_j\|\|r_\ell^i - r_k^i\| \\
&\leq -c_3 + \frac{9\sqrt{3}n^2\bar{h}}{2K}\cdot\bar{a}\cdot\sqrt{n} \\
&\qquad + \left(\bar{g} + \frac{9\sqrt{3}n^2\bar{h}}{2K}\right)\cdot\bar{a}\cdot\frac{\sqrt{n}}{K} \\
&\leq -c_3 + \frac{\sqrt{n}\bar{a}}{K}\left(\bar{g} + 9\sqrt{3}n^2\bar{h}\right),
\end{aligned}$$

which is negative for every K fulfilling (29). Hence, the LP problem is feasible and the proof is finished. □

Note that Theorem 9 follows a similar line of reasoning as the converse theorem in Hafstein (2023), which shows the existence of continuous piecewise affine (CPA) Lyapunov functions. It is important to note that the converse theorem presented here is however not a corollary to the converse theorem in Hafstein (2023). Indeed, since the set of CPQ functions is disjoint from the set of CPA functions a separate converse theorem was needed. Moreover, since Theorem 6 uses linear constraints to enforce copositivity of matrices $-B_{ij}$, it needed to be verified whether these linear constraints can be satisfied for all GUES SLSs given a sufficiently refined triangulation $K^{-1}\mathcal{T}_K$, which is proven with Theorem 9.

5. CPQ Lyapunov functions for CLSs

Similar to the previous section, we will show sufficient conditions, that are linear in their arguments, under which a CPQ function is a Lyapunov function for a CLS (8). Then, a converse theorem will be presented that shows that for a sufficiently refined CPQ function the conditions become necessary for GES.

5.1. Sufficient conditions for CPQ Lyapunov functions of CLSs

To apply the method described in the previous section to the CLS (8), we must make the following assumption on the subsets $\mathcal{F}_j$ in (8) and $\mathcal{C}_i$ in (2). This assumption can always be satisfied by proper selection of the triangulation $\mathcal{T}$.

Assumption 10. The intersection of $\mathcal{C}_i$ and the closure of $\mathcal{F}_j$ is given by

$$\mathcal{C}_i \cap \bar{\mathcal{F}_j} = pos(s_1^{ij}, s_2^{ij}, \ldots, s_{m_{ij}}^{ij}),$$

where $m_{ij} = 0$, $0 < m_{ij} < n$ or $m_{ij} = n$ if $\mathcal{C}_i \cap \bar{\mathcal{F}_j}$ is $\{\underline{0}\}$, equal to a lower dimensional face of $\mathcal{C}_i$ or equal to $\mathcal{C}_i$, respectively, and $\{s_1^{ij}, s_2^{ij}, \ldots, s_{m_{ij}}^{ij}\} \subseteq \{r_1^i, r_2^i, \ldots, r_n^i\}$, for all $i \in \{1, 2, \ldots, N\}$ and $j \in \{1, 2, \ldots, M\}$.

Assumption 10 can be interpreted as that the triangulation $\mathcal{T}$ must be chosen such that the closure of every set $\mathcal{F}_j$, $j \in \{1, 2, \ldots, M\}$, of the CLS (8) is equal to the union of some convex combinations of some vertices $v_0, v_1, \ldots, v_p$ of the triangulation $\mathcal{T}$.

For brevity, let us define $\hat{\mathcal{C}}_{ij} := \mathcal{C}_i \cap \bar{\mathcal{F}_j}$ and

$$\hat{R}_{ij} := \begin{bmatrix} s_1^{ij} & s_2^{ij} & \ldots & s_{m_{ij}}^{ij} \end{bmatrix} \in \mathbb{R}^{n\times m_{ij}}. \tag{36}$$

The CPQ function V is a Lyapunov function to the CLS (8), i.e., satisfies (10a) and (10b), if

$$P_i \succ_{\mathcal{C}_i} 0 \tag{37a}$$

$$Q_{ij} \prec_{\hat{\mathcal{C}}_{ij}} 0 \tag{37b}$$

for all $i \in \{1, 2, \ldots, N\}$ and $j \in \{1, 2, \ldots, M\}$, where $\hat{\mathcal{C}}_{ij} \neq \{\underline{0}\}$, and with Q_{ij} as in (12). For the first step of the two-step procedure in Section 3.3, a CPQ function V that satisfies (37b) needs to be constructed. Hereto we can use a slightly modified version of Theorem 6.

Theorem 11. *Consider a CPQ function V, parameterized as in (2), the CLS (8) and a triangulation $\mathcal{T}$ that satisfies Assumption 10. Define, for all $i \in \{1, 2, \ldots, N\}$ and $j \in \{1, 2, \ldots, M\}$ where $\hat{\mathcal{C}}_{ij} \neq \{\underline{0}\}$, the matrices*

$$B_{ij} = \hat{R}_{ij}^\top Q_{ij}\hat{R}_{ij} \quad \in \mathbb{R}^{m_{ij}\times m_{ij}} \tag{38}$$

and $C_{ij} \in \mathbb{R}^{m_{ij}\times m_{ij}}$ such that

$$\begin{aligned}
[C_{ij}]_{kk} &= [B_{ij}]_{kk} && \forall k \in \{1, 2, \ldots, m_{ij}\} \\
[C_{ij}]_{k\ell} &= \max([B_{ij}]_{k\ell}, 0) && \forall k, \ell \in \{1, 2, \ldots, m_{ij}\}, k \neq \ell.
\end{aligned}$$

If for all $i \in \{1, 2, \ldots, N\}$ and $j \in \{1, 2, \ldots, M\}$ where $\hat{\mathcal{C}}_{ij} \neq \{\underline{0}\}$, the sum of each row of C_{ij} is negative, i.e.,

$$\sum_{\ell=1}^{m_{ij}} [C_{ij}]_{k\ell} < 0 \quad \forall k \in \{1, 2, \ldots, m_{ij}\},$$

then V satisfies condition (37b).

Proof. The proof follows the same steps as the proof of Theorem 6, except in the final step, where it was shown that $\lambda^\top B_{ij}\lambda < 0$ for all $\lambda \in \mathbb{R}^n_{\geq 0} \setminus \{\underline{0}\}$ implies that $Q_{ij} \prec_{\mathcal{C}_i} 0$. Instead, it needs to be shown that

$$\lambda^\top B_{ij}\lambda < 0 \quad \forall \lambda \in \mathbb{R}^{m_{ij}}_{\geq 0} \setminus \{\underline{0}\} \implies Q_{ij} \prec_{\hat{\mathcal{C}}_{ij}} 0 \tag{39}$$

Note that $x \in \hat{\mathcal{C}}_{ij}$ is equivalent to $x = \hat{R}_{ij}\lambda$ for some $\lambda \in \mathbb{R}^{m_{ij}}_{\geq 0}$. Thus

$$x^\top Q_{ij}x < 0 \qquad \forall x \in \hat{\mathcal{C}}_{ij} \setminus \{\underline{0}\}$$

is equivalent to

$$\lambda^\top \hat{R}_{ij}^\top Q_{ij}\hat{R}_{ij}\lambda < 0 \qquad \forall \lambda \in \mathbb{R}^{m_{ij}}_{\geq 0} \setminus \{\underline{0}\}.$$

Since $B_{ij} = \hat{R}_{ij}^\top Q_{ij}\hat{R}_{ij}$, (39) has been proven and the proof is complete. □

Similar to the case of the SLS, an LP problem can be constructed such that a solution to the LP problem parameterizes a CPQ Lyapunov function candidate V that satisfies (37b):

$$\begin{aligned}
&\text{find } \Phi, C_{ij} && && (40)\\
&\text{s.t. } P_i = E_i^\top[\Phi]_{g_i(k)g_i(\ell)}E_i && \forall i, k, \ell \\
&\quad B_{ij} = R_i^\top(A_j^\top P_i + P_i A_j)R_i && \forall i, j, \hat{\mathcal{C}}_{ij} \neq \{\underline{0}\} \\
&\quad [C_{ij}]_{kk} = [B_{ij}]_{kk} && \forall i, j, k, \hat{\mathcal{C}}_{ij} \neq \{\underline{0}\} \\
&\quad [C_{ij}]_{k\ell} \geq [B_{ij}]_{k\ell} && \forall i, j, k, \ell, \hat{\mathcal{C}}_{ij} \neq \{\underline{0}\}, k \neq \ell \\
&\quad [C_{ij}]_{k\ell} \geq 0 && \forall i, j, k, \ell, \hat{\mathcal{C}}_{ij} \neq \{\underline{0}\}, k \neq \ell
\end{aligned}$$

$$\sum_{\ell=1}^{m_{ij}} [C_{ij}]_{k\ell} < 0 \qquad \forall i, j, k, \hat{C}_{ij} \neq \{\underline{0}\}.$$

5.2. Converse theorem

Similar to the converse theorem in the previous section, we want to show that our method always succeeds in finding a Lyapunov function for a GES CLS (8).

Theorem 12. *Assume that system (8) is GES and that the triangulation $K^{-1}\mathcal{T}_K$ satisfies Assumption 10 for some $K^\star \in \mathbb{N}$. Then, the LP problem (40) is feasible for sufficiently large $K \in \mathbb{N}$ that satisfy $K = 2^p K^\star$, where $p \in \mathbb{N}$.*

Proof. First, note that the condition $K = 2^p K^\star$ ensures that Assumption 10 is satisfied, which is a necessary condition to Theorem 11. Second, to prove the feasibility of the LP problem (40), we assign values to all the variables and then show that the constraints of the problem are fulfilled when the variables have these values by following the same steps as in the proof of Theorem 9.

Note that the systems in (8) and (9) are positively homogeneous of degree one, i.e., $F(cx(t)) = cF(x(t))$ for all $c > 0$, where $F(x(t))$ is the right-hand side of (9). If (8), or equivalently (9), is GES, it is shown in Nakamura et al. (2002) that there exists a Lyapunov function W that:

- is infinitely differentiable;
- is positively homogeneous of degree two, i.e., satisfies (23);
- and satisfies for all x and some $c_1, c_2, c_3 > 0$

$$c_1\|x\|^2 \leq W(x) \leq c_2\|x\|^2$$
$$\nabla W(x) A_j x \leq -c_3\|x\|^2 \quad \forall x \in \mathcal{F}_j, \quad j \in \{1, 2, \ldots, M\}.$$

Then, by following the same steps as in Theorem 9, we find that

$$\nabla V_i(r_k^i) A_j r_\ell^i \tag{41}$$

is negative for all $i \in \{1, 2, \ldots, N\}$, $j \in \{1, 2, \ldots, M\}$, $k, \ell \in \{1, 2, \ldots, n\}$ for which $r_k^i, r_\ell^i \in \bar{\mathcal{F}}_j$, if

$$K > \frac{\sqrt{n}\bar{a}\left(\bar{g} + 9\sqrt{3}n^2\bar{h}\right)}{c_3},$$

where $\bar{a}$, $\bar{g}$ and $\bar{h}$ are defined in (26)–(28). If (41) is negative for all i, j, k, ℓ, then all entries of the matrices B_{ij} in (38) are negative and thus the LP problem (40) is feasible. □

Remark 13. In the case that system (8) and the triangulation $K^{-1}\mathcal{T}_K$ do not satisfy Assumption 10 for any $K^\star \in \mathbb{N}$, a different triangulation $\mathcal{T}$ that does satisfy Assumption 10 could be used. The converse theorem only requires that the triangulation $\mathcal{T}$ satisfies the bounds in (30) and (31). This hints at a possible generalization of our converse theorems to triangulations other than the triangulation $K^{-1}\mathcal{T}_K$, which could potentially have numerical benefits.

6. Comparison to similar methods

Alternative methods to assess the stability of SLSs and CLSs are:

- Johansson and Rantzer (1998), which constructs CPQ Lyapunov functions using LMIs;
- Hafstein (2002), which constructs CPA Lyapunov functions using LP and;
- Iervolino et al. (2011), which uses the results in Bundfuss and Dür (2009) to construct CPQ Lyapunov functions using LP similarly to our method, but with different linear inequalities.

In this section we will highlight some differences between these methods and the proposed method.

6.1. Comparison to constructing CPQ Lyapunov functions with LMIs

If the method of Johansson and Rantzer (1998) is applied to the SLS (6), then (11b) would be enforced by the linear matrix inequality

$$Q_{ij} + E_i^\top W_{ij} E_i \prec 0,$$

where $W_{ij} = W_{ij}^\top \in \mathbb{R}^{n\times n}_{\geq 0}$ would be a new optimization variable. This is equivalent to enforcing that $B_{ij} + W_{ij}$ is a negative definite matrix.

In our method, we enforce that C_{ij}, defined by (14) and (15), satisfies (16). It can be shown that this is equivalent to the existence of a matrix $W_{ij} = W_{ij}^\top \in \mathbb{R}^{n\times n}_{\geq 0}$ such that $C_{ij} := B_{ij} + W_{ij}$ is a strictly diagonally dominant and negative definite matrix, i.e.,

$$|[C_{ij}]_{kk}| > \sum_{\ell=1, \ell\neq k}^{n} |[C_{ij}]_{k\ell}| \tag{42}$$

and $[C_{ij}]_{kk} < 0$ for all $k \in \{1, 2, \ldots, n\}$. See Theorem 15 in the Appendix for the proof. The reason we do not use this equivalent formulation in the conditions of LP problems (18) and (40), is because the absolute operators in (42) would require a large number of slack variables, which would increase the number of optimization variables and linear constraints of the LP problems and therefore it is not computationally attractive.

Nevertheless, the alternative formulation shows that our method requires $B_{ij} + W_{ij}$ to be a negative definite and strictly diagonally dominant matrix, whereas the method in Johansson and Rantzer (1998) solely requires it to be negative definite. Therefore, given the same triangulation $\mathcal{T}$, our method will always produce more conservative results. For example, the matrix

$$C_{ij} = B_{ij} + W_{ij} = \begin{bmatrix} -1 & 2 \\ 2 & -5 \end{bmatrix}$$

is negative definite, but not strictly diagonally dominant.

Our method, however, can benefit from *smaller* optimization problems and faster and more robust LP solvers (Ahmadi & Majumdar, 2019). Hence, there is a trade-off between accuracy and complexity. Note that both methods provide necessary and sufficient conditions for stability if the CPQ function uses a sufficiently fine partitioning of the state space (Rantzer & Johansson, 2000). Thus, the conservatism of our method compared to the LMI method goes away for a sufficiently refined partitioning, although the level of sufficient refinement for the LMI-based method will always be lower or equal to the level of sufficient refinement of our LP-based method.

6.2. Comparison to constructing CPA Lyapunov functions with LP

Linear constraints on $v_i^\top \in \mathbb{R}^n_{\geq 0}$ and $c_i \in \mathbb{R}$, $i \in \{1, 2, \ldots, N\}$, such that the CPA function

$$V(x) = v_i^\top E_i x + c_i \qquad \text{if } x \in C_i,$$

is a Lyapunov function to a generic nonlinear system, were introduced in Hafstein (2002). In Andersen et al. (2023) the method was applied to SLSs (6) and in Palacios Roman et al. (2025) to CLSs.

A challenge with this method is that the number of piecewise segments N often needs to be much larger than when using CPQ functions. To illustrate this, consider the planar LTI system

$$\dot{x} = \begin{bmatrix} -\epsilon & -1 \\ 1 & -\epsilon \end{bmatrix} x, \tag{43}$$

where $\epsilon > 0$, as a special instance of the SLS (6), where $n = 2$ and $M = 1$. It was shown in Giesl and Hafstein (2010, Lemma 3.2) that to construct a CPA Lyapunov function for (43), the Lyapunov function requires at least N piecewise segments such that

$$1 < \epsilon \sin\left(\frac{2\pi}{N}\right) + \cos\left(\frac{2\pi}{N}\right). \tag{44}$$

This means that for this specific instance of (6), the CPA method may require a computationally large LP problem to construct a CPA Lyapunov function. This behavior can also become a computational limitation to the method when applying this method to other instances of SLSs (6) and CLSs (8), the latter of which was shown in Palacios Roman et al. (2025).

By contrast, for any stable instance of (6) with $M = 1$, our method can always find a quadratic Lyapunov function, that satisfies the conditions of Theorem 6, provided that the triangulation $\mathcal{T}$ used to parameterize V satisfies the next assumption.

Assumption 14. The vertices of a simplex $\mathcal{S}_i$ of $\mathcal{T}$ all lie in the same orthant, for all $i \in \{1, 2, \dots, N\}$.

The proof of this fact is given in Theorem 16 in the Appendix. This shows that for these specific instances of (6), our proposed method will require $N = 2^n$ simplices to construct a CPQ Lyapunov function with an LP problem, whereas the CPA method may require arbitrarily many simplices N, which is a computational advantage of the proposed method.

6.3. Comparison to constructing CPQ Lyapunov functions with different linear inequalities

In Iervolino et al. (2011), linear inequalities are used to verify the existence of a CPQ Lyapunov function for stable piecewise linear differential inclusions. The linear inequalities are based on Bundfuss and Dür (2009) and are similar to enforcing

$$\begin{aligned} -[B_{ij}]_{k\ell} &> 0 \quad \forall i, j, k, \ell \\ [\hat{R}_{ij}^\top P_i \hat{R}_{ij}]_{k\ell} &> 0 \quad \forall i, j, k, \ell, \end{aligned} \tag{45}$$

i.e., that $-B_{ij}$ and $\hat{R}_{ij}^\top P_i \hat{R}_{ij}$, where B_{ij}, $\hat{R}_{ij}$ and P_i are defined respectively in (38), (36) and (3), only have positive entries for all $i \in \{1, 2, \dots, N\}$ and $j \in \{1, 2, \dots, M\}$. This enforces that $-B_{ij}$ and $\hat{R}_{ij}^\top P_i \hat{R}_{ij}$ are copositive matrices, which are sufficient and necessary conditions for a CPQ function to be a Lyapunov function.

Note however that a copositive matrix does not need to only have positive entries. For example, the matrix

$$-B_{ij} = \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}$$

is copositive, but does not contain only positive entries. Whereas the linear inequalities in Bundfuss and Dür (2009) and Iervolino et al. (2011) would not be able to verify that $-B_{ij}$ is copositive, the proposed linear inequalities in (18) and (40) would, as the rows of B_{ij} sum to negative values, i.e., our inequalities are inherently less conservative. From numerical tests we found that for low dimensional systems ($n = 2$), the difference in conservativeness between the linear inequalities in Bundfuss and Dür (2009) and Iervolino et al. (2011) and our proposed linear inequalities is sufficiently small such that neither method is significantly computationally more beneficial than the other. However, for larger dimensional systems ($n > 2$), the difference in conservativeness increases, which can be explained by the growing number of off-diagonal elements in the matrices B_{ij} on which our linear inequalities pose less restrictive conditions. This causes our method to become computationally more efficient in comparison.

It is worth mentioning that in Bundfuss and Dür (2009) and Iervolino et al. (2011) their method was shown to always be able to verify whether or not a CPQ Lyapunov function could be constructed for a GUES SLS or GES CLS given a triangulation $\mathcal{T}$ within a finite number of steps. If no CPQ Lyapunov function could be constructed using the given triangulation $\mathcal{T}$, the cones of the triangulation $\mathcal{T}$ would have to be partitioned into smaller cones. However, no converse result was given that for a GUES SLS or GES CLS this procedure would eventually find a Lyapunov function. On the contrary, our converse Theorems 9 and 12 show that for a GUES SLS or GES CLS and given the triangulation $\mathcal{T}_K$, we will find a CPQ Lyapunov function that satisfies the conditions of Theorems 6 or 11, respectively, by stepwise increasing K.

7. Numerical case study

In this section, we will demonstrate the method presented in this paper for constructing CPQ Lyapunov functions with LP on elementary ($n = 2$) examples from the literature and on randomly generated SLSs of larger dimension ($n = 4$). We will compare the proposed CPQ (LP) method with the CPQ (LMI) method from Johansson and Rantzer (1998), the CPA (LP) method from Hafstein (2002) and the CPQ (LP) method from Iervolino et al. (2011).

To parameterize the Lyapunov function V, the triangular fan triangulation $\mathcal{T}_K$ mentioned in Section 2 is used. To solve the LP and LMI problems, the solver Mosek (MOSEK ApS, 2021) implemented in MATLAB with Yalmip (Löfberg, 2004), is used. A description of an efficient implementation of our method in C++ is given in Andersen et al. (2024).

7.1. LTI system

Consider the normalized, planar LTI system (43) with $\epsilon = 0.01$. Using (44), a CPA Lyapunov function for the system would require a triangulation $\mathcal{T}$ with $N \geq 315$ piecewise segments. The triangulation $\mathcal{T}_{41}$ is used, since then $N = 2^n \cdot K^{n-1} \cdot n! = 328$. The CPQ (LMI) method comes down to solving the LMIs (11a) and (11b) where $N = M = 1$ and $\mathcal{C}_1 = \mathbb{R}^{n \times n}$. The CPQ (LP) method from Iervolino et al. (2011) comes down to solving (45) with triangulation $\mathcal{T}_1$.

For the proposed CPQ (LP) method, we use the triangulation $\mathcal{T}_1$ ($N = 8$), which satisfies Assumption 14. By solving the LP problem (18) with additional linear constraints (22), a solution Φ, which parameterizes a CPQ Lyapunov function V, is found. See Fig. 3 for a visualization of V.

A comparison between the different methods is made in Table 1. A distinction is made between the total time required to construct, solve and verify the solution of the LP/LMI problem, τ_{tot}, and the computation time required to solve the LP/LMI problem, τ_{sol}. While τ_{tot} can be reduced by using a more efficient implementation, such as the one presented in Andersen et al. (2024), τ_{sol} is more difficult to reduce and is therefore an important scalar when comparing the different methods.

It is clear that the CPA (LP) method uses the most computation time, which is due to the high number of simplices N required due to the low damping ϵ as discussed in Section 6.2. The CPQ (LMI) method is the fastest, which was expected as it only needs to solve two LMIs. The proposed CPQ (LP) method and the CPQ (LP) method from Iervolino et al. (2011) are slightly slower than the CPQ (LMI) method, but the performance is comparable. For

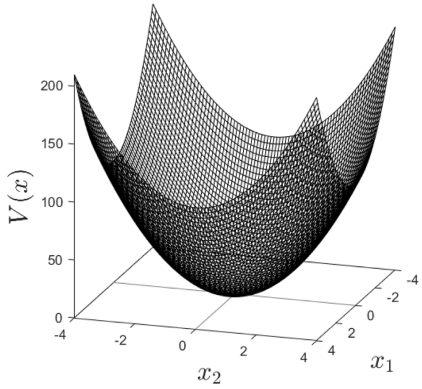


Fig. 3. CPQ Lyapunov function constructed with the proposed method for the LTI system (43) with $\epsilon = 0.01$ and $\mathcal{T}_1$.

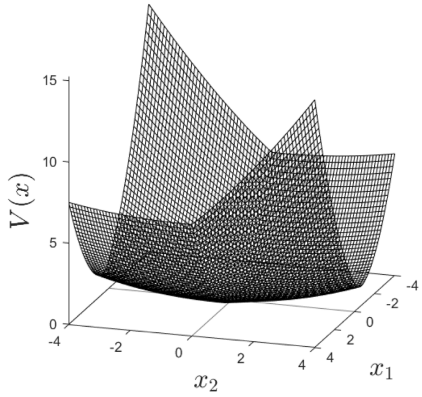


Fig. 4. CPQ Lyapunov function constructed with the proposed method for the SLS (6) with system matrices (46) and $\mathcal{T}_4$.

Table 1
Comparison between methods for elementary examples.

	$N[-]$	τ_{tot}[ms]	τ_{sol}[ms]
LTI system (43)			
CPA (LP) (Hafstein, 2002)	328	467.7	4.893
CPQ (LMI) (Johansson & Rantzer, 1998)	1	55.25	1.306
CPQ (LP) (Iervolino et al., 2011)	8	64.77	1.384
proposed CPQ (LP)	8	73.23	1.537
SLS (46)			
CPA (LP) (Hafstein, 2002)	400	1109	9.261
CPQ (LMI) (Johansson & Rantzer, 1998)	32	226.2	10.18
CPQ (LP) (Iervolino et al., 2011)	56	161.9	3.707
proposed CPQ (LP)	32	186.1	3.711
CLS (47)-(49)			
CPA (LP) (Hafstein, 2002)	192	1181	3.603
CPQ (LMI) (Johansson & Rantzer, 1998)	32	405.8	24.42
CPQ (LP) (Iervolino et al., 2011)	40	134.6	2.174
proposed CPQ (LP)	32	169.2	2.431

the two-dimensional examples, the CPQ (LP) methods will perform similarly: The method from Iervolino et al. (2011) will be slightly faster due to less constraints and optimization variables in the LP problem, but may require a more refined triangulation, as mentioned in Section 6.3. In Section 7.4 the difference between the two method for larger dimensional systems will be highlighted.

7.2. Switched linear system with arbitrary switching

Consider the SLS (6), where $M = 2$ and

$$A_1 = \begin{bmatrix} 0 & 1 \\ -0.01 & -2 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0 & 1 \\ -11.72 & -2 \end{bmatrix}. \tag{46}$$

This system was studied in Andersen et al. (2023), where CPA Lyapunov functions were constructed using linear programming. It was shown that a CPA Lyapunov function could be constructed if the triangulation $\mathcal{T}_{50}$ was used.

To construct a CPQ Lyapunov function with the proposed method, the triangulation $\mathcal{T}_4$ is used. By solving the LP problem (18) with additional linear constraints (22), a solution Φ, which parameterizes a CPQ Lyapunov function V, is found. See Fig. 4 for a visualization of V. The CPQ (LMI) method also requires a triangulation $\mathcal{T}_4$. The CPQ (LP) method from Iervolino et al. (2011) requires a triangulation $\mathcal{T}_7$.

A comparison between the four methods is made in Table 1. Note the difference between the triangulations: the triangulation $\mathcal{T}_{50}$ of a two-dimensional vector space consists of 400 simplices, whereas $\mathcal{T}_4$ consists of only 32 simplices. This shows for this example that in order to construct a CPA Lyapunov function more simplices are required than to construct a CPQ Lyapunov function. The LP problem of the CPA (LP) method is solved in a similar amount of time τ_{sol} as the CPQ (LMI) method. However, due to the large number of simplices N, constructing the LP problem of the CPA (LP) method causes τ_{tot} to be much larger. The CPQ (LP) methods perform best in this example, having both the lowest τ_{sol} and τ_{tot}. This can be attributed to the low number of simplices N, which is thanks to the increased flexibility of CPQ functions over CPA functions, and the computational advantages of LP over LMI solvers.

7.3. Conewise linear systems

Consider the CLS (8), where $M = 4$,

$$A_1 = A_3 = \begin{bmatrix} -0.1 & 1 \\ -5 & -0.1 \end{bmatrix}, \tag{47}$$

$$A_2 = A_4 = \begin{bmatrix} -0.1 & 5 \\ -1 & -0.1 \end{bmatrix} \tag{48}$$

and

$$\begin{aligned} \mathcal{F}_1 \cup \mathcal{F}_3 &= \{x \in \mathbb{R}^n \mid x_1^2 \geq x_2^2\} \\ \mathcal{F}_2 \cup \mathcal{F}_4 &= \{x \in \mathbb{R}^n \mid x_1^2 < x_2^2\}. \end{aligned} \tag{49}$$

This system was studied in Johansson (1999), where CPQ Lyapunov functions were constructed using LMIs. In Johansson (1999), a triangulation similar to $\mathcal{T}_1$ ($N = 8$) was used. However, in the example sliding mode behavior was not taken into account, since for this particular example there are no sliding modes. If we explicitly check for the stability of sliding modes, a triangulation $\mathcal{T}_4$ ($N = 32$) is required.

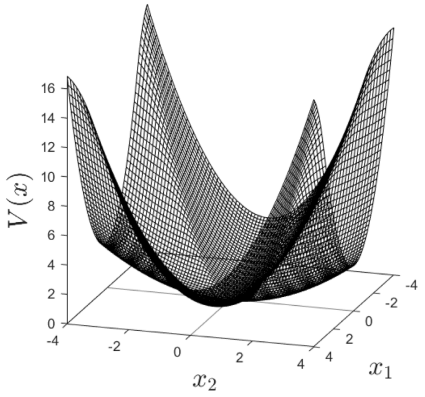


Fig. 5. CPQ Lyapunov function constructed with the proposed method for the CLS (8) with system matrices (47) and (48), subsets (49) and $\mathcal{T}_4$.

To construct a CPQ Lyapunov function with LP with the proposed method, the triangulation $\mathcal{T}_4$ is used. By solving the LP problem (40) with additional linear constraints (22), a solution Φ which parameterizes a CPQ Lyapunov function V is found. See Fig. 5 for a visualization of V. The CPA (LP) method is adapted for the stability analysis of SLSs, see Palacios Roman et al. (2025), and is applied to this example as well. The CPQ (LP) method from Iervolino et al. (2011) requires the triangulation $\mathcal{T}_5$.

See Table 1 for the comparison between the three methods. Similar to the previous example, the CPA (LP) method requires the most simplices N, causing the method to have the highest τ_{tot}. The CPQ (LMI) method has the highest τ_{sol} due to requiring an LMI solver over an LP solver. The CPQ (LP) methods achieve the lowest τ_{sol} and τ_{tot} as they benefit from the best of both other methods: They require a low number of simplices N due to the flexibility of CPQ functions and they can benefit from fast LP solvers.

7.4. Randomly generated SLSs of dimension $n = 4$

Finally, we consider a set of 1000 GUES switched linear systems that were randomly generated. The SLSs are of the form (6) with $n = 4$ and $M = 2$. All four previously mentioned methods for constructing Lyapunov functions are applied to these systems with the triangulation $\mathcal{T}_1$. We define the accuracy α of a method as the percentage of systems for which the method can find a function V that is decreasing along all trajectories of the system (i.e., step one of the two-step procedure in Section 3.3). All four methods would succeed in finding a Lyapunov function given a sufficiently refined triangulation $\mathcal{T}_k$ (a converse result for the method in Iervolino et al. (2011) is implied by our converse theorems), but we are interested in the accuracy and computation time given a set triangulation. The used code is precompiled, which reduces τ_{tot} compared to Table 1.

See Table 2 for the comparison between the four methods, where the mean and standard deviation of τ_{tot} and τ_{sol} are provided. The methods are ordered by their achieved accuracy and it can be seen that τ_{tot} and τ_{sol} increase with accuracy. The proposed method sits in between the CPQ (LP) method from Iervolino et al. (2011) and the CPQ (LMI) method from Johansson and Rantzer (1998) in both accuracy (which follows from the analysis in Sections 6.1 and 6.3) and computation time. This shows how the proposed method achieves the highest accuracy of the LP-based methods and thus comes closer to the accuracy of the CPQ (LMI) method from Johansson and Rantzer (1998), while still having lower computation times than the LMI method.

Table 2
Comparison between methods for randomly generated SLSs.

	α [%]	τ_{tot}[ms]	τ_{sol}[ms]
CPA (LP) (Hafstein, 2002)	5.10	12.8 ± 3.91	7.81 ± 2.75
CPQ (LP) (Iervolino et al., 2011)	15.9	96.9 ± 5.57	65.4 ± 5.16
proposed CPQ (LP)	21.4	398 ± 26.4	148 ± 19.3
CPQ (LMI) (Johansson & Rantzer, 1998)	36.4	839 ± 76.6	232 ± 59.2

8. Conclusion

We have presented a method for constructing CPQ Lyapunov functions for switched linear systems and conewise linear systems by making use of LP. In comparison to classical LMI methods, LP may offer computational advantages especially for problems involving a large number of optimization parameters and constraints, which is the case when accurate analysis with CPQ functions requires a partition of the state space with a large number of regions. In order to arrive at an LP problem, we formulated sufficient conditions for the copositivity of matrices in terms of testing diagonal dominance. This involves defining linear constraints that can be solved using an LP. Interestingly, despite that there is some conservatism in the formulation of copositivity of a matrix in terms of linear inequalities, we provided converse theorems guaranteeing the feasibility of our LP conditions in case of a G(U)ES SLS or CLS and a sufficiently refined partitioning. Hence, there is no conservatism in our LP conditions provided that the partitioning is sufficiently refined. The method is shown to be the least conservative of the LP-based methods Hafstein (2002) and Iervolino et al. (2011) given a fixed partitioning and numerically competitive with the LMI-based method in Johansson and Rantzer (1998) on various CLSs and SLSs. For future work, the method could be generalized to piecewise linear and affine systems, as well as discrete-time versions of those systems and the systems considered in this paper.

Appendix. Auxiliary theorems

Theorem 15. *Given a matrix $B = B^\top \in \mathbb{R}^{n\times n}$, define $C = C^\top \in \mathbb{R}^{n\times n}$ as*

$$[C]_{kk} = [B]_{kk} \qquad \forall k \in \{1, 2, \dots, n\} \tag{A.1}$$

$$[C]_{k\ell} = \max([B]_{k\ell}, 0) \qquad \forall k, \ell \in \{1, 2, \dots, n\}, k \neq \ell. \tag{A.2}$$

Then, C satisfies

$$\sum_{\ell=1}^{n} [C]_{k\ell} < 0 \quad \forall k \in \{1, 2, \dots, n\}, \tag{A.3}$$

if and only if there exists $W = W^\top \in \mathbb{K}_{\geq 0}^{n\times n}$ such that $B + W$ is a negative definite and strictly diagonally dominant matrix.

Proof. First, we will prove the "only if" part of the theorem. If C as defined in (A.1) and (A.2) satisfies (A.3), then C is negative definite (see first step in Theorem 6) and strictly diagonally dominant, because

$$0 > \sum_{\ell=1}^{n} [C]_{k\ell} \qquad \Longrightarrow$$

$$-[C]_{kk} > \sum_{\ell=1,\ell\neq k}^{n} [C]_{k\ell} \qquad \Longrightarrow$$

$$|[C]_{kk}| > \sum_{\ell=1,\ell\neq k}^{n} |[C]_{k\ell}|.$$

Here we make use of the fact that the off-diagonal entries of C are nonnegative. Thus, $W = C - B$ is an admissible solution to ensure that $B + W$ is negative definite and strictly diagonally dominant.

Next, we will prove the "if" part of the theorem. Note that if $D := B + W$ is negative definite, its diagonal entries fulfill $[D]_{kk} = e_k^\top D e_k < 0$, where e_k is the standard kth unit vector. Since W only contains nonnegative entries,

$$0 > [D]_{kk} \geq [C]_{kk} \qquad \forall k \in \{1, 2, \ldots, n\}$$
$$|[D]_{k\ell}| \geq |[C]_{k\ell}| \qquad \forall k, \ell \in \{1, 2, \ldots, n\}, k \neq \ell.$$

Therefore, C must be strictly diagonally dominant if D is strictly diagonally dominant and the diagonal of C must only contain strictly negative entries. Thus, C will satisfy (A.3). □

Theorem 16. *Given an LTI system*

$$\dot{x} = Ax, \tag{A.4}$$

where $x \in \mathbb{R}^n$ is the state and $A \in \mathbb{R}^{n\times n}$ is Hurwitz, there always exists a CPQ Lyapunov function V that satisfies the conditions of Theorem 6, provided that V is parameterized with a triangulation $\mathcal{T}$ that satisfies Assumption 14.

Proof. First, note that (A.4) is a special case of (6) where $M = 1$ and thus Theorem 6 can be applied.

Second, suppose that the CPQ Lyapunov function V is a quadratic Lyapunov function $V = x^\top P x$, i.e., $P_i = P$ for all $i \in \{1, \ldots, N\}$, where P is the solution of the Lyapunov equation

$$A^\top P + PA = -I. \tag{A.5}$$

It is a well-known result that V is then a Lyapunov function of (A.4). Now, we prove that P also satisfies the conditions of Theorem 6 to complete the proof.

The matrices Q_{ij} in (11b) will be equal to $-I$ for all $i \in \{1, \ldots, N\}$ and $j = 1$. Therefore,

$$B_{ij} = R_i^\top Q_{ij} R_i = -R_i^\top R_i$$

Since the columns of R_i are vertices of a simplex $\mathcal{S}_i$ of a triangulation $\mathcal{T}$, Assumption 14 implies that B_{ij} has only negative entries on its diagonal and nonpositive entries otherwise. Then, if we define C_{ij} according to (14) and (15), the diagonal entries of C_{ij} will also be negative and the off-diagonal entries will be zero. Such matrices C_{ij} will satisfy conditions (16). □

References

Ahmadi, A. A., & Majumdar, A. (2019). DSOS and SDSOS optimization: More tractable alternatives to sum of squares and semidefinite optimization. *SIAM Journal on Applied Algebra and Geometry*, *3*(2), 193–230.

Ait Rami, M., Bokharaie, V., Mason, O., & Wirth, F. (2013). Stability criteria for SIS epidemiological models under switching policies. *Discrete and Continuous Dynamical Systems - Series B*, *19*(9), 2865–2887.

Andersen, S., Giesl, P. A., & Hafstein, S. (2023). Common Lyapunov functions for switched linear systems: Linear programming based approach. *IEEE Control Systems Letters*, *7*, 901–906.

Andersen, S., Hafstein, S., Palacios Roman, J. J., & Van den Eijnden, S. J. A. M. (2024). Efficient implementation of piecewise quadratic Lyapunov function computations for switched linear systems. In *21st international conference of informatics in control, automation and robotics* (pp. 277–284).

Apostol, T. M. (1969). *Calculus*: *Vol. 2*, (2nd ed.). John Wiley & Sons.

Aubin, J.-P., & Cellina, A. (1984). *Differential inclusions*. Springer.

Baier, R., Grüne, L., & Hafstein, S. (2012). Linear programming based Lyapunov function computation for differential inclusions. *Discrete and Continuous Dynamical Systems - Series B*, *17*, 33–56.

Boyd, S., & Vandenberghe, L. (2004). *Convex optimization*. Cambridge University Press.

Branicky, M. (1998). Multiple Lyapunov functions and other analysis tools for switched and hybrid systems. *IEEE Transactions on Automatic Control*, *43*(4), 475–482.

Bundfuss, S., & Dür, M. (2009). Copositive Lyapunov functions for switched systems over cones. *Systems & Control Letters*, *58*(5), 342–345.

Camlibel, M. K., Pang, J.-S., & Shen, J. (2006). Conewise linear systems: Non-zenoness and observability. *SIAM Journal on Control and Optimization*, *45*(5), 1769–1800.

Camlibel, M. K., Pang, J.-S., & Shen, J. (2007). Lyapunov stability of complementarity and extended systems. *SIAM Journal on Optimization*, *17*(4), 1056–1101.

Clarke, F., Ledyaev, Y., & Stern, R. (1998). Asymptotic stability and smooth Lyapunov functions. *Journal of Differential Equations*, *149*, 69–114.

Filippov, A. (1988). *Differential equations with discontinuous right-hand side*. Kluwer.

Gerschgorin, S. (1931). Über die abgrenzung der eigenwerte einer matrix. *Izvestija Akademii Nauk SSSR, Serija Matematika*, *7*(3), 749–754.

Giesl, P. A., & Hafstein, S. (2010). Existence of piecewise affine Lyapunov functions in two dimensions. *Journal of Mathematical Analysis and Applications*, *371*, 233–248.

Giesl, P. A., Hafstein, S., & Pokkakkillath, S. (2025). Piecewise quadratic Lyapunov functions for stochastic differential equations with linear programming. *Discrete and Continuous Dynamical Systems - Series B*, *30*(6), 2027–2050.

Hadeler, K. (1983). On copositive matrices. *Linear Algebra and its Applications*, *49*, 79–89.

Hafstein, S. (2002). Lyapunov function construction for ordinary differential equations with linear programming. *Dynamical Systems*, *17*(2), 137–150.

Hafstein, S. (2023). Lyapunov functions for switched linear systems: Proof of convergence for an LP computational approach. *IEEE Control Systems Letters*, *7*, 3283–3288.

Heemels, W. P. M. H., & Weiland, S. (2008). Input-to-state stability and interconnections of discontinuous dynamical systems. *Automatica*, *44*(12), 3079–3086.

Iervolino, R., Vasca, F., & Iannelli, L. (2011). A cone-copositive approach for the stability of piecewise linear differential inclusions. In *50th IEEE conference on decision and control and European control conference* (pp. 1062–1067).

Iervolino, R., Vasca, F., & Iannelli, L. (2015a). Cone-copositive piecewise quadratic Lyapunov functions for conewise linear systems. *IEEE Transactions on Automatic Control*, *60*(11), 3077–3082.

Iervolino, R., Vasca, F., & Iannelli, L. (2015b). Stability analysis of conewise linear systems with sliding modes. In *54th IEEE conference on decision and control* (pp. 1174–1179).

Johansson, M. (1999). *Piecewise linear control systems* (Ph.D. thesis), Lund University.

Johansson, M., & Rantzer, A. (1998). Computation of piecewise quadratic Lyapunov functions for hybrid systems. *IEEE Transactions on Automatic Control*, *43*(4), 555–559.

Jungers, R. M. (2024). Statistical comparison of path-complete Lyapunov functions: a discrete-event systems perspective. *IFAC-PapersOnLine*, *58*, 258–263.

Kamenetskiy, V. (2017). Frequency-domain stability conditions for hybrid systems. *Automation and Remote Control*, *78*(12), 2101–2119.

Kellett, C. (2015). Classical converse theorems in Lyapunov's second method. *Discrete and Continuous Dynamical Systems - Series B*, *20*(8), 2333–2360.

Liberzon, D. (2003). Switching in systems and control. In *Systems & control: foundations & applications*, Birkhäuser.

Lin, H., & Antsaklis, P. J. (2009). Stability and stabilizability of switched linear systems: A survey of recent results. *IEEE Transactions on Automatic Control*, *54*(2), 308–322.

Löfberg, J. (2004). YALMIP: A toolbox for modeling and optimization in MATLAB. In *Proceedings of the CACSD conference* (p. 284–289).

Lygeros, J., Godbole, D., & Sastry, S. (1998). Verified hybrid controllers for automated vehicles. *IEEE Transactions on Automatic Control*, *43*(4), 522–539.

Middlebrook, R. D., & Cuk, S. (1976). A general unified approach to modelling switching-converter power stages. In *IEEE power electronics specialists conference* (pp. 18–34).

MOSEK ApS (2021). MOSEK optimization toolbox for MATLAB 10.1.21.

Nakamura, H., Yamashita, Y., & Nishitani, H. (2002). Smooth Lyapunov functions for homogeneous differential inclusions. In *Proceedings of the 41st SICE annual conference*: *Vol. 3*, (pp. 1974–1979).

Palacios Roman, J. J., Hafstein, S., Van den Eijnden, S. J. A. M., Heertjes, M. F., & Heemels, W. P. M. H. (2025). Stability analysis of hybrid integrator-gain systems using linear programming [manuscript submitted for publication]. Eindhoven University of Technology.

Pogromsky, A., Jirstrand, M., & Spangeus, P. (1998). On stability and passivity of a class of hybrid systems. In *37th IEEE conference on decision and control: Vol. 4*, (pp. 3705–3710).

Rantzer, A., & Johansson, M. (2000). Piecewise linear quadratic optimal control. *IEEE Transactions on Automatic Control*, *45*, 629–637.

Sun, Z. (2010). Stability of piecewise linear systems revisited. *Annual Reviews in Control*, *34*(2), 221–231.

Van den Eijnden, S. J. A. M., Heertjes, M. F., Heemels, W. P. M. H., & Nijmeijer, H. (2020). Hybrid integrator-gain systems: A remedy for overshoot limitations in linear control? *IEEE Control Systems Letters*, *4*(4), 1042–1047.

Van den Eijnden, S. J. A. M., Heertjes, M. F., Nijmeijer, H., & Heemels, W. P. M. H. (2024). Stability analysis of hybrid integrator-gain systems: A frequency-domain approach. *Automatica*, *164*, Article 111641.

Van Loon, S., Hunnekens, B., Heemels, W. P. M. H., Van de Wouw, N., & Nijmeijer, H. (2016). Split-path nonlinear integral control for transient performance improvement. *Automatica*, *66*, 262–270.

Vandenberghe, L., & Boyd, S. (1996). Semidefinite programming. *SIAM Review*, *38*, 49–95.

Walter, W. (1998). *Ordinary differential equations*. Springer.

Yakubovich, V. (1977). The S-procedure in nonlinear control theory. *Vestnik Leningrad University*, *4*, 73–93.

Yuan, C., & Wu, F. (2015). Hybrid control for switched linear systems with average dwell time. *IEEE Transactions on Automatic Control*, *60*, 240–245.

Zhao, X., Zhang, L., Shi, P., & Liu, M. (2012). Stability and stabilization of switched linear systems with mode-dependent average dwell time. *IEEE Transactions on Automatic Control*, *57*(7), 1809–1815.



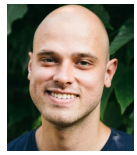
Juan Javier Palacios Roman received his M.Sc. degree (cum laude) in Mechanical Engineering from the Eindhoven University of Technology (TU/e) in 2023. From 2023 to 2024, he was working with the Department of Mathematics at the University of Iceland as a research assistant. Since 2024, he has been with Department of Mechanical Engineering, TU/e, where he is currently pursuing a Ph.D. degree. His current research includes the analysis of hybrid systems and the analysis and design of extremum seeking control schemes with applications in nuclear fusion.



Sigurdur Hafstein received his M.S. degree in mathematics from the University Duisburg–Essen, Germany, in 1998 and his Ph.D. degree in mathematics from the same university in 2002. From 2006 to 2016 he was a professor for mathematics in the School of Engineering at Reykjavik University and is since 2017 a professor with the Department of Mathematics at the University of Iceland. He has lead several research projects, has coauthored 120 scholarly articles, and serves as associate editor for Discrete and Continuous Dynamical Systems – Series B. His research interests include numerical algorithms, differential equations, and dynamical systems.



Peter Giesl received his Diplom in Mathematics from the University of Hamburg, Germany and his Maitrise in Mathematics from the University Paris-Sud, France. He obtained a Ph.D. and his Habilitation in Mathematics from the TU Munich, Germany in 2000 and 2005, respectively. He is currently a Professor of Mathematics at the University of Sussex, UK. His research interests are analytical and numerical methods in Dynamical Systems. In particular, he is interested in the stability and basin of attraction of equilibria and periodic orbits, using Lyapunov functions and contraction metrics.



Sebastiaan van den Eijnden received the M.Sc. degree and Ph.D. degree (Cum Laude) from the Eindhoven University of Technology (TU/e) in 2017 and 2022, respectively. From 2022–2023, he has been working as a Mechatronics and Control Engineer at ASML, the Netherlands. Currently, Sebastiaan is an Assistant Professor at the Mechanical Engineering department, TU/e. He was a recipient of the 2023 Hybrid Systems CSS TC Outstanding Student Paper Prize and the 2023 Automatica Paper Prize. His current research includes scaled relative graphs and input–output methods for stability analysis, positive systems, and fundamental performance limitations in control.



Stefania Andersen received her M.S. degree in electrical engineering from the University of Iceland and is currently a Ph.D. student in mathematics at the same university. She has coauthored several scholarly articles on her work with numerical computations of Lyapunov functions. Her research interests include numerical algorithms, differential equations, and dynamical systems.



W.P.M.H. Heemels received M.Sc. (mathematics) and Ph.D. (EE, control theory) degrees (summa cum laude) from the Eindhoven University of Technology (TU/e) in 1995 and 1999, respectively. From 2000 to 2004, he was with the Electrical Engineering Department, TU/e, as an assistant professor, and from 2004 to 2006 with TNO-Embedded Systems Institute as a Research Fellow. Since 2006, he has been with the Department of Mechanical Engineering, TU/e, where he is currently a Full Professor and Vice-Dean. He held visiting professor positions at ETH, Switzerland (2001), UCSB, USA (2008) and University of Lorraine, France (2020). He is a Fellow of IEEE and IFAC, and was the chair of the IFAC Technical Committee on Networked Systems (2017–2023). He served/s on the editorial boards of Automatica, Nonlinear Analysis: Hybrid Systems (NAHS), Annual Reviews in Control, and IEEE Transactions on Automatic Control, and is the Editor-in-Chief of NAHS as of 2023. He was the recipient of the 2019 IEEE L-CSS Outstanding Paper Award and the Automatica Paper Prize 2020–2022. His current research includes hybrid and cyber–physical systems, networked and event-triggered control systems and model predictive control and their applications.

Paper V

Minimum Average Dwell-Time Computations via Continuous Piecewise Quadratic Lyapunov Functions

S. Andersen, S.F. Hafstein

2026 30th International Conference on Methods and Models in Automation and Robotics (MMAR), pp. 351–356, 2026.

© 2026 IEEE. Reprinted, with permission, from original authors.

Recently, continuous, piecewise linear Lyapunov functions were used to compute the minimum average-dwell time needed to assert global exponential stability for switched linear systems. Here, we use continuous, piecewise quadratic Lyapunov functions to do the same and compare the efficiency of the two methods. As in the case for piecewise linear Lyapunov functions, we use linear programming to parametrize our Lyapunov functions on a conical subdivision of the state-space.

Minimum Average Dwell-Time Computations via Continuous Piecewise Quadratic Lyapunov Functions

Stefania Andersen[1] and Sigurdur Hafstein[1]

***Abstract*— Recently, continuous, piecewise linear Lyapunov functions were used to compute the minimum average-dwell time needed to assert global exponential stability for switched linear systems. Here, we use continuous, piecewise quadratic Lyapunov functions to do the same and compare the efficiency of the two methods. As in the case for piecewise linear Lyapunov functions, we use linear programming to parameterize our Lyapunov functions on a conical subdivision of the state-space.**

I. Introduction

We consider switched linear systems of the form

$$\dot{\mathbf{x}} = A_\sigma \mathbf{x}, \quad \sigma(t) \in \{1, 2, \dots, N_{\text{subs}}\} =: \mathcal{P}, \tag{1}$$

where the $A_i \in \mathbb{R}^{n\times n}$ are Hurwitz matrices, i.e. the real-parts of the eigenvalues are negative and the origin is globally exponentially stable (GES) for the individual subsystems $\dot{\mathbf{x}} = A_i\mathbf{x}$. Such systems have been intensively studied in the literature, see e.g. [9], [21], [19], [6], [8], [7]. GES of the origin for the arbitrary switched system (1) means that there exist constants $M, \beta > 0$, such that for any switching signal $\sigma\colon \mathbb{R}_+ \to \mathcal{P}$, $\mathbb{R}_+ := [0, \infty)$, i.e. right-continuous and only with finitely many discontinuity points on any finite interval, and any initial value $\boldsymbol{\xi} \in \mathbb{R}^n$, we have

$$\|\mathbf{x}_\sigma(t;\boldsymbol{\xi})\|_2 \le M \exp(-\beta t)\|\boldsymbol{\xi}\|_2$$

for all $t \in \mathbb{R}_+$. Here $t \mapsto \mathbf{x}_\sigma(t;\boldsymbol{\xi})$, $\mathbf{x}_\sigma(0;\boldsymbol{\xi}) = \boldsymbol{\xi}$, is the solution to (1) with switching σ. This means that $t \mapsto \mathbf{x}_\sigma(t;\boldsymbol{\xi})$ is continuous and with $0 = t_0 < t_1 < t_2 < \dots$, where the $t_1, t_2, \dots$ are the discontinuity points of σ, also called switching times, we have $\dot{\mathbf{x}}(t;\boldsymbol{\xi}) = A_{\sigma(t_i)}\mathbf{x}(t;\boldsymbol{\xi})$ for all $t_i < t < t_{i+1}$, $i = 0, 1, \dots$.

Although the origin is GES for each subsystem $\dot{\mathbf{x}} = A_i\mathbf{x}$, $i \in \mathcal{P}$, this does not necessarily hold true for the arbitrary switched system, see e.g. [18] for a counterexample. However, if one demands a large enough minimum average-dwell time (MADT) on the switching, then the origin is always GES for the switched system; see e.g. [15], [18]. Denote by $N_\sigma(t, s)$ the number of switching times of the switching signal $\sigma\colon \mathbb{R}_+ \to \mathcal{P}$ on the open time-interval (s, t). The switching signal σ is said to have an MADT $\tau_a > 0$, if there exists a constant $N_0 > 0$ such that

$$N_\sigma(t, s) \le N_0 + \frac{t - s}{\tau_a}$$

for every finite interval $(s, t) \subset \mathbb{R}_+$. If there are no restrictions on $N_\sigma(t, s)$, except that it is finite for any finite interval (s, t), we say the MADT is equal to zero. We denote the set of all switching signals with MADT $\tau_a \ge 0$ with Σ_{τ_a}.

An interesting question for the system (1) with concrete matrices A_i, $i \in \mathcal{P}$, is now: what is the MADT $\tau_a \ge 0$, for which we can assert the GES of the origin for all switchings in Σ_{τ_a}? To tackle this question, the following theorem from [4], which is adapted from Theorem 3.2 in [18], is useful. It gives a sufficient condition in terms of Lyapunov functions for the individual subsystems. Recall that $\mathcal{K}_\infty$ is the set of strictly increasing, unbounded, continuous functions $\beta\colon \mathbb{R}_+ \to \mathbb{R}_+$ with $\beta(0) = 0$.

Theorem 1: Assume that for each $i \in \mathcal{P}$, there exists a locally Lipschitz continuous function $V_i\colon \mathbb{R}^n \to \mathbb{R}$, such that

$$\underline{a}^*(\|\mathbf{x}\|_2) \le V_i(\mathbf{x}) \le \overline{a}^*(\|\mathbf{x}\|_2), \quad \forall i \in \mathcal{P}, \tag{2a}$$
$$D^+V_i(\mathbf{x}, A_i\mathbf{x}) \le -\alpha V_i(\mathbf{x}), \quad \forall i \in \mathcal{P}, \tag{2b}$$
$$V_i(\mathbf{x}) \le \mu V_j(\mathbf{x}), \quad \forall i, j \in \mathcal{P}, \tag{2c}$$

where $\overline{a}^*, \underline{a}^* \in \mathcal{K}_\infty$, $\alpha > 0$, and $\mu \ge 1$. Then the origin is GES for the switched system (1) for every switching σ with average dwell-time τ fulfilling

$$\tau > \frac{\ln(\mu)}{\alpha}, \tag{3}$$

i.e. for every $\sigma \in \Sigma_\tau$ if $\mu > 1$ and for an arbitrary switching σ if $\mu = 1$.

□

Here,

$$D^+V_i(\mathbf{x}, A_i\mathbf{x}) := \limsup_{h\to 0+} \frac{V_i(\mathbf{x} + hA_i\mathbf{x}) - V_i(\mathbf{x})}{h}$$

is the Dini-derivative of V_i along the solution trajectory of the system $\dot{\mathbf{x}} = A_i\mathbf{x}$ at the point $\mathbf{x} \in \mathbb{R}^n$.

In [14], [2], [4] linear programming (LP) was used to parameterize continuous and piecewise affine (CPA) Lyapunov functions, that are then extended to continuous, cone-wise linear Lyapunov functions, for the system (1) to compute sufficiently large MADT to assert GES of the origin; of course one is interested in as low MADT as possible needed to assert stability. In more detail, in Corollary 2 in [14] the conditions of Theorem 1 are enforced through the stricter conditions

$$\underline{a}\|\mathbf{x}\|_2 \le V_i(\mathbf{x}) \le \overline{a}\|\mathbf{x}\|_2, \quad \forall i \in \mathcal{P}, \tag{4a}$$
$$D^+V_i(\mathbf{x}, A_i\mathbf{x}) \le -\alpha\|\mathbf{x}\|_2, \quad \forall i \in \mathcal{P}, \tag{4b}$$
$$V_i(\mathbf{x}) \le \mu V_j(\mathbf{x}), \quad \forall i, j \in \mathcal{P}, \tag{4c}$$

[1]The Science Institute, University of Iceland, Dunhagi 5, 107 Reykjavik, Iceland {saa20,shafstein}@hi.is

This research was partially supported by the Icelandic Research Fund, grant number 228725-051.

979-8-3195-1921-4/26/$31.00 ©2026 IEEE

for fixed constants $\underline{a}, \overline{a} > 0$ and $\mu \geq 1$, and then $\alpha > 0$ was maximized. From the existence of these Lyapunov functions one can then assert GES of the origin for every switching signal with MADT $\tau > \overline{a} \ln(\mu)/\alpha$. In [4] CPA Lyapunov functions fulfilling (4a) and (4c), but (2b) instead of (4b), were computed for constants $\underline{a}, \overline{a}, \alpha > 0$ and $\mu \geq 1$. Note that these are just the conditions in Theorem 1 for CPA Lyapunov functions. From the existence of these Lyapunov functions one can then assert GES of the origin for every switching signal with MADT $\tau > 0$ fulfilling (3).

Note that while one can maximize $\alpha > 0$ in the LP associated to (4) to obtain as low $\tau > 0$ as possible for the given constants $\underline{a}, \overline{a} > 0$ and $\mu \geq 1$, this is not possible in an LP for (2) because of the term $\alpha V_i(\mathbf{x})$, which is quadratic in the variables. Hence, in [4] an algorithm was presented for efficiently searching the (α, μ)-plane for values delivering the lowest MADT needed to assert GES and we will use this algorithm in our numerical experiments.

Note that the LP method in [14] was able to deliver lower estimates on the MADT needed to assert GES of the origin than a standard linear matrix inequality (LMI) approach to compute quadratic Lyapunov functions, see [2]. However, a triangulation of the state-space is needed and for some systems this can be limiting. Indeed, as discussed in [10], a large number of simplices in the triangulation is needed to compute CPA Lyapunov functions for some systems, in particular if the negative real parts of some eigenvalues are small but the imaginary parts are not. To tackle this problem continuous and piecewise quadratic (CPQ) Lyapunov functions have been suggested, e.g. in [17], [16], but until recently one needed semidefinite programming (SDP) to parameterize such Lyapunov functions using LMIs. However, recently in [20] an LP approach was suggested to parameterize CPQ Lyapunov functions for switched linear systems; its efficient implementation was described in [5]. In this paper we present an LP problem to parameterize CPQ Lyapunov functions for the individual subsystems $\dot{\mathbf{x}} = A_i \mathbf{x}$, $i \in \mathcal{P}$, that additionally fulfil the compatibility conditions (2c), and we compare this method with the method in [4], which uses LP to parameterize CPA Lyapunov functions, to obtain a (low) MADT needed to assert GES of the origin for the switched system (1).

The rest of this paper is organized as follows: In Section II we describe the LP problem to compute CPQ Lyapunov functions that deliver a MADT $\tau \geq 0$, such that GES of the origin is asserted for the switched system (1) for every switching $\sigma \in \Sigma_\tau$. In Section III we compare our novel approach using CPQ Lyapunov functions to the CPA Lyapunov functions approach from [4]. Since $\sigma \in \Sigma_0$ corresponds to arbitrary switching σ, this analysis includes a study of the computation of CPQ common Lyapunov functions (CLFs) and CPA CLFs. Finally, in Section IV we give some conclusions.

II. CPQ Lyapunov functions for MADT analysis

To compute CPQ Lyapunov functions fulfilling the conditions of Theorem 1 for the system (1) in order to obtain a MADT that guaranties exponential stability of the origin, we adapt the method to compute CPQ Lyapunov functions in [5], [20]. Let us recapitulate the method: first a subdivision of the state-space into cones with the apex as the origin is needed. This is done by using the triangulation $\mathcal{T}_K^{\mathbf{F}}$ described in detail in e.g. [14], and then extending the n-simplices to cones; the parameter $K \in \mathbb{N}$ fixes the number of the simplices in the triangulation $\mathcal{T}_K^{\mathbf{F}}$, which is approximately spherically symmetric [1]. See Fig. 1 for a schematic figure of the triangulation $\mathcal{T}_1^{\mathbf{F}}$ in two-dimensions.

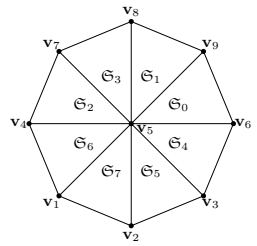


Fig. 1: Triangulation $\mathcal{T}_1^{\mathbf{F}}$ with simplices $\mathfrak{S}_\nu$ and vertices $\mathbf{v}_j$ indexed

In a little more detail, we have n-simplices

$$\mathfrak{S}_\nu := \operatorname{co}\{\mathbf{x}_0^\nu, \mathbf{x}_1^\nu, \ldots, \mathbf{x}_n^\nu\}, \quad \nu \in I,$$

where I is an index set for the simplices and $\mathbf{x}_0^\nu = \mathbf{0}$ for all $\nu \in I$. These simplices have the property that two different simplices intersect in a common face and that $\bigcup_{\nu \in I} \mathfrak{S}_\nu$ is a neighbourhood of the origin. Each point $\mathbf{x} \in \mathfrak{S}_\nu$ can be written uniquely as a convex combination of the vertices of $\mathfrak{S}_\nu$, i.e. $\mathbf{x} = \sum_{i=0}^n \lambda_i \mathbf{x}_i^\nu$, $\sum_{i=0}^n \lambda_i = 1$, $\lambda_i \in [0,1]$. For each simplex $\mathfrak{S}_\nu$ we define the cone (recall $\mathbf{x}_0^\nu = \mathbf{0}$)

$$\mathfrak{C}_\nu := \operatorname{cone}\{\mathbf{x}_1^\nu, \mathbf{x}_2^\nu, \ldots, \mathbf{x}_n^\nu\} = \left\{ \sum_{i=1}^n \lambda_i \mathbf{x}_i^\nu \colon \lambda_i \geq 0 \right\}$$

and these cones subdivide the state-space $\mathbb{R}^n$.

Our LP problem will attempt to parameterize Lyapunov functions $V_i \colon \mathbb{R}^n \to \mathbb{R}$ as in Theorem 1, that additionally are quadratic on each cone $\mathfrak{S}_\nu$. For this we need a parametrized representation of such CPQ functions and we need to express the conditions (2) as linear constraints in these parameters. Let us first discuss the parametrized representation.

For each simplex $\mathfrak{S}_\nu$, $\nu \in I$, define the matrix $X_\nu := [\mathbf{x}_1^\nu, \mathbf{x}_2^\nu, \cdots, \mathbf{x}_n^\nu] \in \mathbb{R}^{n \times n}$, i.e. the $\mathbf{x}_j^\nu$ are the columns of X_ν; see e.g. [3] or Remark 9 in [11] for a discussion of this definition. Clearly, every point $\mathbf{x} \in \mathfrak{C}_\nu$ can be written in a unique way as $\mathbf{x} = X_\nu \boldsymbol{\lambda}$, $\boldsymbol{\lambda} \in \mathbb{R}_+^n$. On each cone $\mathfrak{C}_\nu$ the function V_i, $i \in \mathcal{P}$, will have the formula $V_i(\mathbf{x}) = \mathbf{x}^T P_i^\nu \mathbf{x}$, $P_i^\nu \in \mathbb{R}^{n \times n}$.

Now one could proceed by using the entries of the matrices P_i^ν, $\nu \in I$ and $i \in \mathcal{P}$ as variables, to express the conditions (6) below in these variables. However, we prefer to use the parametrization from [5], [20], [16]. For this we collect all

the variables for the matrices P_i^ν, $\nu \in I$, in a symmetric matrix of variables $\Phi_i \in \mathbb{R}^{p\times p}$, such that for each cone $\mathfrak{C}_\nu$, $\nu \in I$, we have $\Psi_i^\nu := X_\nu^T P_i^\nu X_\nu$, where the (k,ℓ)-th entry $\psi_{k,\ell}^{i,\nu}$ of the matrix $\Psi_i^\nu \in \mathbb{R}^{n\times n}$ is the $(g_\nu(k), g_\nu(\ell))$-th entry $\varphi_{g_\nu(k),g_\nu(\ell)}^{(i)}$ of the matrix of variables Φ_i. The functions

$$g_\nu \colon \{1,2,\dots,n\} \to \{1,2,\dots,p\}, \quad \nu \in I,$$

are defined such that $\mathbf{x}_i^\nu = \mathbf{v}_{g_\nu(i)}$, where $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_p$ are the different vertices of the triangulation $\mathcal{T}_K^{\mathbf{F}}$ and take care of the correct assignment to the variables in Φ_i for P_i^ν. Note that when two vertices of different simplices $\mathfrak{S}_\nu$ and $\mathfrak{S}_\mu$ are equal, i.e. $\mathbf{x}_i^\nu = \mathbf{x}_j^\mu$, then $g_\nu(i) = g_\mu(j)$. This can be done in an identical way for all $i \in \mathcal{P}$. Now, for an $\mathbf{x} = \sum_{i=1}^n \lambda_i \mathbf{x}_i^\nu = X^\nu \boldsymbol{\lambda} \in \mathfrak{C}_\nu$ we have

$$\begin{aligned} V_i(\mathbf{x}) &= \mathbf{x}^T P_i^\nu \mathbf{x} = \boldsymbol{\lambda}^T X_\nu^T (X_\nu^{-1})^T \Psi_i^\nu X_\nu^{-1} X_\nu \boldsymbol{\lambda} \\ &= \boldsymbol{\lambda}^T \Psi_i^\nu \boldsymbol{\lambda} = \sum_{k,\ell=1}^n \lambda_k \lambda_\ell \varphi_{g_\nu(k),g_\nu(\ell)}^{(i)}. \end{aligned} \tag{5}$$

Hence, the $n\times n$-matrix $\Psi_i^\nu = \left(\psi_{k,\ell}^{i,\nu}\right)_{k,\ell} = \left(\varphi_{g_\nu(k),g_\nu(\ell)}^{(i)}\right)_{k,\ell}$ is essentially the matrix P_i^ν in the cone-coordinates $\boldsymbol{\lambda}$ on $\mathfrak{C}_\nu$. From (5) it follows that $\mathbf{x}^T P_i^\nu \mathbf{x} > 0$ is equivalent to Ψ_i^ν being strictly copositive; recall that a matrix $M \in \mathbb{R}^{n\times n}$ is said to be copositive if $\mathbf{x}^T M \mathbf{x} \geq 0$ for all $\mathbf{x} \in \mathbb{R}_+^n$ and strictly copositive if $\mathbf{x}^T M \mathbf{x} > 0$ for all $\mathbf{x} \in \mathbb{R}_+^n \setminus \{\mathbf{0}\}$.

In [5] it is shown that the V_i are well-defined and continuous when they are defined in this way, i.e. if $\mathbf{x} \in \mathfrak{C}_\nu \cap \mathfrak{C}_\mu$, $\nu, \mu \in I$, the value $V_i(\mathbf{x})$ in (5) does not depend on whether one uses $\mathfrak{C}_\nu$ or $\mathfrak{C}_\mu$.

Now let us turn to how to express the constraints (2) in Theorem 1 for our piecewise quadratic Lyapunov functions. It is not difficult to see that if $V_i(\mathbf{x}) = \mathbf{x}^T P_i^\nu \mathbf{x}$ on $\mathfrak{C}_\nu$, then the conditions in (2) are equivalent to: for every $\nu \in I$ we have for every $\mathbf{x} \in \mathfrak{C}_\nu \setminus \{\mathbf{0}\}$ and every $i, j \in \mathcal{P}$ that

$$\mathbf{x}^T P_i^\nu \mathbf{x} > 0, \tag{6a}$$

$$\mathbf{x}^T \left(A_i^T P_i^\nu + P_i^\nu A_i + \alpha P_i^\nu\right) \mathbf{x} \leq 0, \tag{6b}$$

$$\mathbf{x}^T \left(P_i^\nu - \mu P_j^\nu\right) \mathbf{x} \leq 0. \tag{6c}$$

For (6b) just note that for every $\mathbf{x} \in \mathbb{R}^n$ there must be at least one $\mathfrak{C}_\nu$ such that $\mathbf{x} + [0,h] A_i \mathbf{x} \in \mathfrak{C}_\nu$ for a small enough $h > 0$ and then

$$\begin{aligned} D^+ V_i(\mathbf{x}, A_i\mathbf{x}) &= \limsup_{h\to 0+} \frac{V_i(\mathbf{x} + hA_i\mathbf{x}) - V_i(\mathbf{x})}{h} \\ &= \limsup_{h\to 0+} \frac{(\mathbf{x} + hA_i\mathbf{x})^T P_i^\nu (\mathbf{x} + hA_i\mathbf{x}) - \mathbf{x}^T P_i^\nu \mathbf{x}}{h} \\ &= \mathbf{x}^T \left(A_i^T P_i^\nu + P_i^\nu A_i\right) \mathbf{x}. \end{aligned}$$

Note that if $\mu > 1$ we do not have to explicitly enforce the constraints (6a) if, for every $\nu \in I$, we take care that for at least one $i \in \mathcal{P}$ we have $\mathbf{x}^T P_i \mathbf{x} \neq 0$ for all $\mathbf{x} \in \mathfrak{C}_\nu \setminus \{\mathbf{0}\}$. That is, $\mathbf{x}^T P_j^\nu \mathbf{x} > 0$ for all $j \in \mathcal{P}$ and all $\mathbf{x} \in \mathfrak{C}_\nu \setminus \{\mathbf{0}\}$ is then implicitly enforced by the conditions (6c). To see this, assume for a contradiction that $\mathbf{x}^T P_i \mathbf{x} < 0$ for some $\mathbf{x} \in \mathfrak{C}_\nu \setminus \{\mathbf{0}\}$. Then, by (6c) we have

$$\mathbf{x}^T(\mu P_j)\mathbf{x} \geq \mathbf{x}^T P_i \mathbf{x} > \mu \cdot \mathbf{x}^T P_i \mathbf{x} = \mathbf{x}^T(\mu P_i)\mathbf{x} \geq \mathbf{x}^T P_j \mathbf{x},$$

which is impossible because $0 > \mu \cdot \mathbf{x}^T P_i \mathbf{x} \geq \mathbf{x}^T P_j \mathbf{x}$ and $\mu > 1$. Hence, $\mathbf{x}^T P_i \mathbf{x} > 0$ and $\mathbf{x}^T P_j \mathbf{x} \geq \mu^{-1} \mathbf{x}^T P_i \mathbf{x} > 0$ for all $\mathbf{x} \in \mathfrak{C}_\nu \setminus \{\mathbf{0}\}$.

To express the constraints (6b) and (6c) as linear constraints in our variables for our LP problem it is advantageous to define the matrices

$$\begin{aligned} Q_i^\nu &:= X_\nu^T \left(A_i^T P_i^\nu + P_i^\nu A_i + \alpha P_i^\nu\right) X_\nu \\ \text{and} \qquad R_{i,j}^\nu &:= X_\nu^T \left(P_i^\nu - \mu P_j^\nu\right) X_\nu \end{aligned}$$

for all $\nu \in I$ and all $i, j \in \mathcal{P}$. Note that in the cone coordinates $\boldsymbol{\lambda} = X_\nu^{-1}\mathbf{x}$ we have

$$\begin{aligned} \boldsymbol{\lambda}^T Q_i^\nu \boldsymbol{\lambda} &= \mathbf{x}^T \left(A_i^T P_i^\nu + P_i^\nu A_i + \alpha P_i^\nu\right) \mathbf{x} \\ \text{and} \quad \boldsymbol{\lambda}^T R_{i,j}^\nu \boldsymbol{\lambda} &= \mathbf{x}^T \left(P_i^\nu - \mu P_j^\nu\right) \mathbf{x} \end{aligned}$$

so (6b) is equivalent to $-Q_i^\nu$ being copositive and (6c) is equivalent to $-R_{i,j}^\nu$ begin copositive.

For asserting the copositivity of the matrices through linear constraints we use the following theorem proved in [20].

Theorem 2: Let $B \in \mathbb{R}^{n\times n}$ be a symmetric matrix and let $C \in \mathbb{R}^{n\times n}$ be a symmetric matrix such that $c_{k,k} = b_{k,k}$, $c_{k,\ell} \geq b_{k,\ell}$, $c_{k,\ell} \geq 0$, for all $k, \ell = 1, 2, \dots, n$, $k \neq \ell$. Then $-B$ is copositive if $\sum_{\ell=1}^n c_{k,\ell} \leq 0$ for all $k = 1, 2, \dots, n$. If the inequality is strict for all k, then $-B$ is strictly copositive. □

We close this section by summing up how we compute CPQ Lyapunov functions for the system (1) as in Theorem 1 and writing out the components of the relevant matrices Ψ_i^ν, Q_i^ν, and $R_{i,j}^\nu$ in the variables $\varphi_{k,\ell}^{(i)}$.

For the system (1) we use LP and Theorem 2 to obtain copositive $n \times n$-matrices Q_i^ν and $R_{i,j}^\nu$, $\nu = 1, 2, \dots, N$ and $i, j \in \mathcal{P}$, where

$$Q_i^\nu = \left(q_{k,\ell}^{i,\nu}\right)_{k,\ell} \quad \text{and} \quad R_{i,j}^\nu = \left(r_{k,\ell}^{i,j,\nu}\right)_{k,\ell},$$

have, with $\hat{A}_i^\nu = \left(\hat{a}_{k,\ell}^{i,\nu}\right)_{k,\ell} := X_\nu^{-1} A_i X_\nu \in \mathbb{R}^{n\times n}$, the components

$$\begin{aligned} q_{k,\ell}^{i,\nu} = \Big(\sum_{r=1}^n \hat{a}_{r,k}^{\nu,i} \varphi_{g_\nu(r),g_\nu(\ell)}^{(i)} + \hat{a}_{r,\ell}^{\nu,i} \varphi_{g_\nu(k),g_\nu(r)}^{(i)}\Big) \\ + \alpha \varphi_{g_\nu(k),g_\nu(\ell)}^{i} \end{aligned}$$

and

$$r_{k,\ell}^{i,j,\nu} = \varphi_{g_\nu(k),g_\nu(\ell)}^{(i)} - \mu \varphi_{g_\nu(k),g_\nu(\ell)}^{(j)}.$$

Additionally, we enforce $\Psi_i^\nu = \left(\varphi_{g_\nu(k),g_\nu(\ell)}^{(i)}\right)_{k,\ell}$ to be strictly copositive for $i = 1$, again using LP and Theorem 2. If we obtain a feasible solution to our LP problem for some given parameters $\alpha > 0$ and $\mu \geq 1$, i.e. are successful in generating the copositive matrices Q_i^ν and $R_{i,j}^\nu$, then the Lyapunov functions V_i for the subsystems $\dot{\mathbf{x}} = A_i\mathbf{x}$ are defined using the matrices $P_i^\nu = (X_\nu^{-1})^T \Psi_i^\nu X_\nu^{-1}$, and the origin is GES for every switching signal with MADT $\tau > \ln(\mu)/\alpha$. In particular, if $\mu = 1$, we have $V_1 = V_2 = \cdots = V_{N_{subs}} =: V$, the function V is a CLF for all the subsystems $\dot{\mathbf{x}} = A_i\mathbf{x}$, $i \in \mathcal{P}$, and the origin is GES under arbitrary switching.

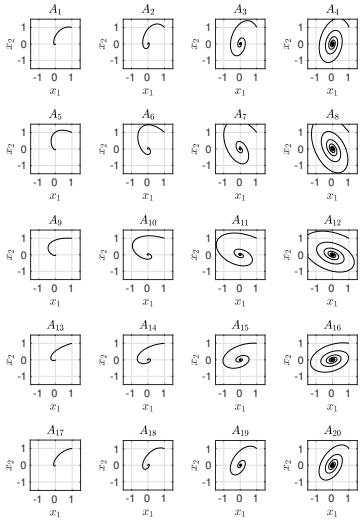


Fig. 2: Trajectories of the subsystems considered starting at $(1, 1)$. The system matrices are defined in [2].

III. Numerical Experiments

In this section we compare our method to compute CPQ Lyapunov functions as in Theorem 1 using LP to the method in [4], where LP is used to compute CPA Lyapunov functions. The solver we use in all cases is Gurobi [12]. We consider the switched linear system (1) with $N_{\text{subs}} \in \{2, 3, 4\}$ subsystems in our comparison. The subsystems are chosen from the 20 planar systems defined in [2], where the formulas for the matrices A_i are given; their trajectories can be seen in Fig. 2. We use all possible ${}_{20}\mathrm{C}_2 = 190$ combinations of two subsystems, all ${}_{20}\mathrm{C}_3 = 1,140$ combinations of three subsystems, and all ${}_{20}\mathrm{C}_4 = 4,845$ combinations of four subsystems; altogether $6,175$ switched systems.

We want to get a picture of what is happening for both high and low resolutions of the triangulation. We performed the computations using the triangulations $\mathcal{T}_K^{\mathbf{F}}$ with the scaling factors $K \in \{1, 2, \ldots, 10\}$; in the plane $\mathcal{T}_K^{\mathbf{F}}$ consists of $8K$ triangles. For each switched system we first try to find a CLF for the subsystems, i.e. we use $\mu = 1$ and a small $\alpha > 0$, see also [3], [20] for such methods for CPA- and CPQ CLFs, respectively. If a CLF can be computed for the subsystems, then the origin is GES for the switched system under arbitrary switching, i.e. with a MADT equal to zero.

If we are not successful in computing a CLF for the subsystems, then we use the algorithm from [4] to find a MADT $\tau > 0$ that asserts GES of the origin for the switched system. We are interested in computing as small MADT $\tau > 0$ as possible, and the algorithm searches the (α, μ)-plane for values such that we obtain a feasible solution to the LP problem with $\tau = \ln(\mu)/\alpha$ as small as possible. The feasibility problem can either be the LP problem described in the last section to obtain CPQ Lyapunov functions or the LP problem from [4] to compute CPA Lyapunov functions. The algorithm walks along and jumps between level sets of $\tau(\alpha, \mu) := \ln(\mu)/\alpha$, see (3), in order to efficiently search for the MADT needed for stability. It is defined as follows:

Algorithm 3: We fix $N, M \in \mathbb{N}$, $N \geq 2$, and $a, b, \overline{\alpha} > 0$ and distribute the values $a = \tau_N < \tau_{N-1} < \ldots < \tau_2 < \tau_1 = b$ and $0 < \alpha_1 < \ldots < \alpha_M < \overline{\alpha}$ uniformly, i.e.

$$\tau_j = (a - b)\frac{j-1}{N-1} + b \quad \text{and} \quad \alpha_i = \overline{\alpha}\frac{i}{M+1}$$

for $j = 1, 2, \ldots, N$ and $i = 1, 2, \ldots, M$.

Set $i = 1$ and $j = 1$. Then execute:

1) Solve the feasibility problem with $\alpha = \alpha_i$ and $\mu = \exp(\tau_j \alpha_i)$.
2)
   - If there is a feasible solution to the problem, then increase j by one, i.e. $j \leftarrow j + 1$, and go back to step (1). That is, we decrease τ from τ_j to τ_{j+1} and check if this problem, with a lower MADT and a smaller $\mu = \exp(\tau_{j+1}\alpha_i)$, also has a feasible solution.
   - If there does not exists a feasible solution to the problem, then increase i by one, i.e. $i \leftarrow i+1$, and go back to step (1). That is, we increase α from α_i to α_{i+1} and set $\mu = \exp(\tau_j \alpha_{i+1})$ and check if this problem, with the same MADT $\tau = \tau_j$, but with larger parameters α and μ, has a feasible solution.

This algorithm is discussed in detail in [4]. A strict upper bound $\overline{\alpha}$, i.e. the optimal interval $[0, \overline{\alpha}]$ on the α-axis, can be computed a priori depending on the subsystems and the type of Lyapunov functions and the search for optimal pairs (α, μ), such that $\tau = \ln(\mu)/\alpha$ is as low as possible, is much more efficient than an exhaustive search and with almost as good results.

In our experiments, the total number of variables in the LP problem to compute CPQ Lyapunov functions, when the switched system has two, three or four subsystems, is roughly 3.8, 4.8 and 5.8 times the number of variables in the LP problem to compute CPA Lyapunov functions, respectively. The number of constraints in the LP problem to compute CPQ Lyapunov functions is in all cases roughly 2.6 times the number of constraints in the LP problem to compute CPA Lyapunov functions.

A. Results

Since our method first attempts to compute a CLF, we get a comparison of the efficiency in computing CPQ- and CPA CLFs for a given resolution K of the triangulation. The results are summarized in Table I. The LP problem for a CPQ CLF always has a feasible solution if the LP problem for a CPA CLF has a feasible solution and often, for a given resolution parameter K, one can generate a CPQ CLF but not a CPA CLF. Note that for large enough K the LP for a CPA CLF will have a feasible solution and

TABLE I: The number of switched systems, where the LP for CPQ CLF ($\mu = 1$) had a feasible solution for the same, or lower, triangulation scaling factor K than the LP for CPA CLF. The results are split by the number of subsystems N_{subs} along with the overall count on the right. Note that in some cases (summarized in the bottom line) we found no CPA CLF with $K \leq 10$ although we could compute a CPQ CLF.

N_{subs}	2	3	4	SUM
Same K	18	13	2	33
Lower K	109	389	741	1239
no CPA CLF	15	120	397	672

also parameterize a CLF, see [13] for a proof. In Fig. 3 we plot the computed mean MADT, needed to assert GES of the origin for the switched system, as a function of the scaling parameter K. We do this using all the switched systems, for which we were able to compute a CPQ CLF with $K \leq 10$, and we compare the CPQ Lyapunov functions approach and the CPA Lyapunov functions approach. For Algorithm 3 we computed the optimal $\overline{\alpha}$ as described in [4] and set $[a, b] = [\tau_{\min}, \tau_{\max}] = [0, 25]$, except for the CPA Lyapunov functions approach with low K, where we set $b = \tau_{\max} = 100$ and used $\tau = \tau_{\max} = 100$ for the statistics when we did not get a feasible solution. Additionally, we plot the maximum values for the needed MADT using error bars; the minimum value is always zero. The results are clear: the CPQ Lyapunov function approach clearly outperforms the CPA Lyapunov function approach in computing a CLF, when one exist, and even if a CLF cannot be computed for a given K, then the CPQ Lyapunov function approach usually delivers a much lower MADT needed to assert GES.

Now, let us turn to the switched systems, for which we were not able to compute a CLF. In most or all of these cases we may assume, that a MADT restriction $\tau > 0$ is needed on the switching for GES of the origin for the switched system. In Fig. 4 we plot the computed mean MADT needed to assert GES of the origin as a function of the scaling parameter K. We do this using all the switched systems, for which we were *not* able to compute a CPQ CLF with $K \leq 10$, and we compare the CPQ Lyapunov functions approach and the CPA Lyapunov functions approach. The parameters for Algorithm 3 were set as above. For low K the CPQ Lyapunov functions approach again clearly outperforms the CPA Lyapunov functions approach. Notably, the computed MADT needed to assert GES does improve very little for larger K in the CPQ Lyapunov functions approach, which means that it is sufficient to use small K to get good results. For small K the CPA Lyapunov functions approach delivers much higher MADT needed to assert GES of the origin for the switched system than the CPQ Lyapunov functions approach. The results improve with larger K and the message is: the CPA Lyapunov functions approach needs more refined triangulations to deliver acceptable results.

To analyse this further, we compare in Fig. 5 how often the CPQ Lyapunov function approach delivered a lower MADT needed to assert GES than the CPA Lyapunov

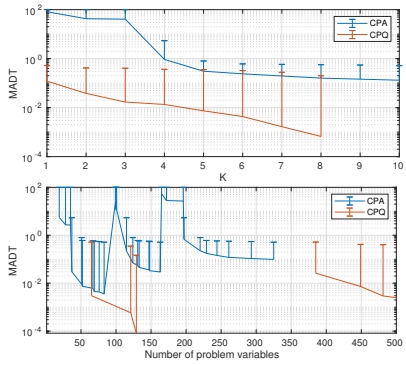


Fig. 3: Mean value of the MADT needed to assert GES of the origin as a function of the triangulation scaling factor K (top) and the number of problem variables in the LP problem (bottom), but only for the systems where we were able to generate a CPQ CLF. Comparison between the CPQ Lyapunov functions and CPA Lyapunov functions approaches. The error bars show the maximum values in each case. Note the logarithmic scale of the MADT-axis. The minimum values were all equal to zero, i.e. there was always a switched system where we could assert the GES of the origin under arbitrary switching.

function approach. We performed this analysis separately for switched system with two-, three- and four subsystems and we considered all the switched system, both those for which we were able to compute a CLF and those where we were not able to compute a CLF. The results are again very interesting. For low K, the advantage of the CPQ Lyapunov approach is larger for more complicated switched systems, i.e. with more subsystems.

IV. Conclusions

We presented an algorithm to parameterize continuous and piecewise quadratic (CPQ) Lyapunov functions for linear switched systems on a triangulation of the state-space. The algorithm generates a linear programming (LP) problem, of which every feasible solution parameterizes CPQ Lyapunov functions for the subsystems of the switched system that fulfil a compatibility condition such that one obtains a minimum average-dwell time (MADT) $\tau \geq 0$ sufficient to assert that the origin is globally exponentially stable (GES) for every switching fulfilling the MADT condition. We compared our novel algorithm in numerical experiments to an algorithm in the literature that uses LP to parameterize continuous and piecewise affine (CPA) Lyapunov functions. Our new algorithm compares very favourably, in particular for triangulations with few triangles and for switched systems with several subsystems. Further, our numerical experiments showed that CPQ Lyapunov functions are substantially su-

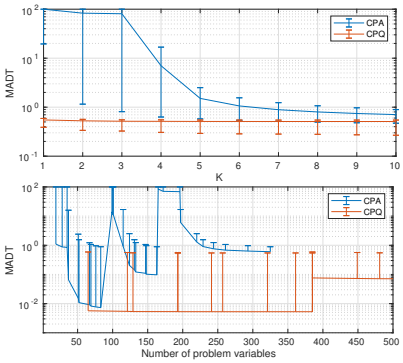


Fig. 4: Mean value of the MADT needed to assert GES of the origin as a function of the triangulation scaling factor K (top) and the number of variables in the LP problem (bottom), but now for the systems where we were *not* able to generate a CPQ CLF. Comparison between the CPQ Lyapunov functions approach and CPA Lyapunov functions approach. The error bars show the maximum- and minimum values in each case. Note the logarithmic scale of the MADT-axis.

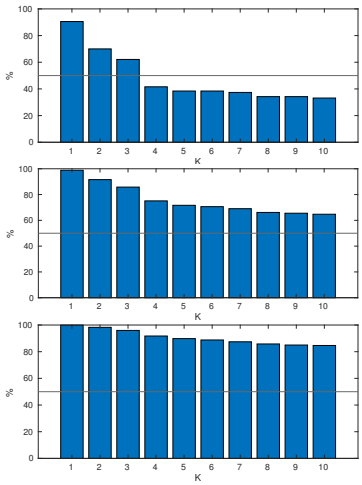


Fig. 5: Bar plot that shows the portion of systems where the CPQ Lyapunov functions approach delivered lower MADT needed to assert GES of the origin than the CPA Lyapunov functions approach for each of the triangulation scaling factors $K = 1, 2, \ldots, 10$. The switched systems have two (top), three (middle), and four (bottom) subsystems.

perior for computing common Lyapunov functions (CLFs) for the subsystems, asserting the GES of the origin under arbitrary switching, when they exist.

References

[1] S. Albertsson, P. Giesl, S. Gudmundsson, and S. Hafstein. Simplicial complex with approximate rotational symmetry: A general class of simplicial complexes. *J. Comput. Appl. Math.*, 363:413–425, 2020.

[2] S. Andersen, E. August, S. Hafstein, and J. Piccini. *Simulation and Modeling Methodologies, Technologies and Applications*, chapter Numerical Computations of Lyapunov Functions for Switched Linear Systems, pages 192–213. Springer, 2024.

[3] S. Andersen, P. Giesl, and S. Hafstein. Common Lyapunov functions for switched linear systems: Linear programming-based approach. *IEEE Control Systems Letters*, 7:901–906, 2023.

[4] S. Andersen and S. Hafstein. Investigating average dwell-time constraints on multiple Lyapunov functions for switched linear systems. In *IFAC Proceedings Volumes: Proceedings of the 13th IFAC Symposium on Nonlinear Control Systems (NOLCOS)*, volume to appear, 2025.

[5] S. Andersen, S. Hafstein, J. Palacios Roman, and S. van den Eijnden. Efficient implementation of piecewise quadratic Lyapunov function computations for switched linear systems. In *Proceedings of the 21st International Conference on Informatics in Control, Automation and Robotics - Volume 1: ICINCO*, pages 277–284. INSTICC, SciTePress, 2024.

[6] F. Blanchini and P. Colaneri. Vertex/plane characterization of the dwell-time property for switching linear systems. In *49th IEEE Conference on Decision and Control (CDC)*, pages 3258–3263, 2010.

[7] C. Briat. Convex conditions for robust stabilization of uncertain switched systems with guaranteed minimum and mode-dependent dwell-time. *Systems and Control Letters*, 78:63 – 72, 2015.

[8] G. Chesi, P. Colaneri, J. C. Geromel, R. Middleton, and R. Shorten. A nonconservative LMI condition for stability of switched systems with guaranteed dwell time. *IEEE Transactions on Automatic Control*, 57(5):1297–1302, 2012.

[9] W. Dayawansa and C. Martin. A converse Lyapunov theorem for a class of dynamical systems which undergo switching. *IEEE Trans. Automat. Control*, (44):751–760, 1999.

[10] P. Giesl and S. Hafstein. Existence of piecewise linear Lyapunov functions in arbitrary dimensions. *Discrete Contin. Dyn. Syst. Ser. A*, 32(10):3539–3565, 2012.

[11] P. Giesl and S. Hafstein. Revised CPA method to compute Lyapunov functions for nonlinear systems. *J. Math. Anal. Appl.*, 410:292–306, 2014.

[12] Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2024.

[13] S. Hafstein. Lyapunov functions for switched linear systems: Proof of convergence for an LP computational approach. *IEEE Control Systems Letters*, 7:3283–3288, 2023.

[14] S. Hafstein and A. Tanwani. Linear programming based lower bounds on average dwell-time via multiple Lyapunov functions. *European Journal of Control*, 74:100838, November 2023.

[15] J. Hespanha and A. Morse. Stability of switched systems with average dwell-time. In *In Proc. 38th IEEE Conference on Decision and Control*, pages 2655–2660, 1999.

[16] M. Johansson. *Piecewise Linear Control Systems*. PhD thesis: Lund University, Sweden, 1999.

[17] M. Johansson and A. Rantzer. Computation of piecewise quadratic Lyapunov functions for hybrid systems. *IEEE Trans. Automat. Control*, 43(4):555–559, 1998.

[18] D. Liberzon. *Switching in systems and control*. Systems & Control: Foundations & Applications. Birkhäuser, 2003.

[19] Hai Lin and Panos J. Antsaklis. Stability and stabilizability of switched linear systems: A survey of recent results. *IEEE Transactions on Automatic Control*, 54(2):308–322, 2009.

[20] J. Palacios Roman, S. Hafstein, P. Giesl, S. van den Eijnden, S. Andersen, and M. Heemels. Constructing continuous piecewise quadratic Lyapunov functions with linear programming. *Automatica*, 186(112838), 2026.

[21] R. Shorten, F. Wirth, O. Mason, K. Wulff, and C. King. Stability criteria for switched and hybrid systems. *SIAM Review*, 49(4):545–592, 2007.