A numerical technique
for stability analysis
of linear switched systems

C.A. Yfoulis and R. Shorten

A numerical technique for stability analysis of linear switched systems

C.A. Yfoulis, R. Shorten

Hamilton Institute, NUI, Maynooth, Ireland
e-mail: Christos.Yfoulis@may.ie

November 11, 2003

Summary

In this report the *ray-gridding approach*, a new numerical technique for the stability analysis of linear switched systems is presented. It is based on uniform partitions of the state-space in terms of ray directions which allow refinable families of polytopes of adjustable complexity to be examined for invariance. In this framework the existence of a *polyhedral Lyapunov function* that is common to a family of asymptotically stable subsystems can be checked efficiently via simple iterative algorithms. The technique can be used to prove the stability of switched linear systems, classes of linear time-varying systems and *Linear Differential Inclusions*.

We also present preliminary results on two other related problems; namely, the existence of *stabilising switching sequences* for a switched system constructed from a family of unstable linear subsystems and the construction of *multiple polyhedral Lyapunov functions*.

Keywords : Stability; Switched systems; Polyhedral Lyapunov functions; Absolute stability; Multiple Lyapunov functions; Stabilising switching sequences.

1 Introduction

Recent years has seen enormous interest in switching systems. While many important issues have been resolved, many issues related to the stability of such systems remain open. In this context two very important questions remain unresolved; namely, if a given switching system remains stable under arbitrary switching sequences, or if this is impossible to identify the switching domains in the state-space and/or switching frequencies which result in stable behavior.

Lyapunov theory provides a convenient starting point for the study of continuous-time and discrete-time linear and nonlinear systems. While most of the available results pertain to the existence of quadratic Lyapunov functions, recently derived converse theorems, suggest that while such a function always exists for stable switching systems, the associated Lyapunov function may not be quadratic. In this context interest has grown in the study of non-quadratic, and in particular, piecewise linear (PL) Lyapunov functions (LF). PL LFs have been considered in a number of papers for establishing the stability of nonlinear time-varying systems and numerical techniques for the calculation of such functions have been developed. Although the class of PL LFs appears powerful in theory, the computational requirements necessary to establish their existence represents a serious bottleneck in practice. The main reason is that a complex representation (with a large number of parameters) is usually required for a solution to be found rendering the techniques applicable to low-dimensional problems only.

In this work we develop a new numerical technique for the calculation of polyhedral Lyapunov functions (PLFs) for switched linear systems. The existence of a polyhedral LF is equivalent to the existence of a polytope which is invariant under the dynamical flow of a set of LTI systems.

A *ray-gridding* technique has been developed for the problem of calculating controllable and recoverable regions in [50] and is a useful framework for systematic generation of refinable families of polytopes with adjustable complexity, having their vertices on certain ray directions.

In this work the same framework is applied for the stability analysis problem of linear switched systems. It is shown that the calculation of invariant polytopes is significantly simplified resulting in a noteworthy reduction in the computational burden. Although the technique is conceived for linear

switched systems, it can be also applied to nonlinear systems represented in the form of LDIs or in an absolute stability framework. The suitability of the technique to other related problems in stability of switched systems is also investigated. In summary, the technique has been applied to the following problems:

- The calculation of *polyhedral* Lyapunov functions proving asymptotic or absolute stability of arbitrary switching sequences for stable linear subsystems,

- Checking the *existence* of stabilising switching sequences on certain switching domains for unstable subsystems,

- Calculation of *multiple* polyhedral Lyapunov functions specifying families of stabilising switching sequences for stable subsystems.

This report is organised as follows: After introducing the necessary mathematical results and preliminaries in section 2, the ray-gridding technique is introduced in section 3 for the first problem mentioned above and applied to some illustrative examples. In sections 4 and 5 the technique is extended to the second and third aforementioned problems.

Notation : In this report, $\mathbb{R}$ denotes the real numbers and $\mathbb{R}^n$ is the vector space of n-dimensional real vectors. All vectors are assumed to be column vectors. $\boldsymbol{x}^T$ denotes the transpose and x_i the i-th component of vector $\boldsymbol{x}$ respectively. If P is a set in $\mathbb{R}^n$, $ri\{P\}$, and ∂P denote the relative interior and the boundary of P respectively. For a set S, $|S|$ denotes the cardinality of S. For a polytope P, $vert\{P\}$ denotes the set of vertices of P. $conv\{V\}$ and $cconv\{V\}$ denote the convex hull and conic convex hull of a set of vectors V. The inequality symbols $>, \geq, <, \leq$ for vectors are understood componentwise.

2 Mathematical results and preliminaries

We are interested in *switched linear systems* of the form

$$\dot{\boldsymbol{x}} = \mathbf{A}(t)\boldsymbol{x} \quad , \quad \mathbf{A}(t) \in \{\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_p\}, \mathbf{A}_i \in \mathbb{R}^{n \times n} \ , \ i = 1, 2, \ldots, p \tag{1}$$

where all the individual subsystems $\mathbf{A}_i$ are linear continuous-time invariant systems and switching between them occurs according to a rule. The subsystems $\dot{\boldsymbol{x}} = \mathbf{A}_i\boldsymbol{x} \ , \ i = 1, \ldots, p$ in (1) may be stable or unstable. It might be the case that system (1) remains stable for any switching strategy, hence appropriate methods for investigating whether this is true are important. If Lyapunov theory is to be used, proving asymptotic stability is equivalent to the existence of a *common* Lyapunov function for all subsystems. We consider the following problem [30]:

Problem 1. Check the existence of a common Lyapunov function for each individual subsystem of (1), that guarantees that system (1) is stable or asymptotically stable for any switching signal.

A closely related problem in the literature ivolved developing conditions for the asymptotic stability of an *LDI (linear differential inclusion)*

$$\dot{\boldsymbol{x}} \in F(\boldsymbol{x}) \quad , \quad F(\boldsymbol{x}) \in conv\{\mathbf{A}\boldsymbol{x} : \mathbf{A} \in \mathcal{A}\} \quad , \quad \mathcal{A} = \left\{ \sum_{i=1}^{p} \mu_i \mathbf{A}_i \ , \ 0 \leq \mu_i \leq 1 \ , \ \sum_{i=1}^{p} \mu_i = 1 \right\} \tag{2}$$

where the matrices $\mathbf{A}_i$ are the vertices of the matrix polytope $\mathcal{A}$. Again, the problem of finding a Lyapunov function proving stability is equivalent to finding a simultaneous Lyapunov function for all subsystems $\dot{\boldsymbol{x}} = \mathbf{A}_i\boldsymbol{x} \ , \ i = 1, \ldots, p$ [36],[37]. A necessary condition for the asymptotic stability of (1) or (2) is that all individual subsystems are asymptotically stable. From now on, when focusing to Problem 1 we will assume that all linear systems $\dot{\boldsymbol{x}} = \mathbf{A}_i\boldsymbol{x} \ , \ i = 1, \ldots, p$ are asymptotically stable, i.e. that all matrices $\mathbf{A}_i$ are Hurwitz.

A related problem is concerned with stability analysis for systems which do not have a common Lyapunov function; namely, systems for which a destabilising switching sequence exists. In this case,
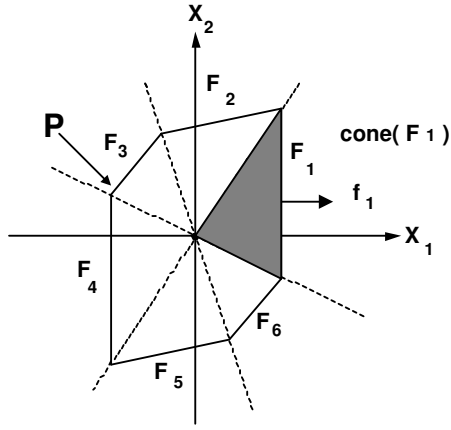
Figure 1: A centrally symmetric polytope.

a stability analysis requires finding the family of switching sequences that result in a stable switching systems. Such sequences can be specified using the *multiple* Lyapunov function idea [15],[16], [6],[29] and is discussed in Section 5.

2.1 Polyhedral Lyapunov functions and invariance

For a solution to Problem 1, we restrict our attention to the class of *polyhedral* Lyapunov functions (PLFs), a special class of piecewise linear (PL) Lyapunov functions. PLFs are *set-induced* PL functions

$$V_P(\boldsymbol{x}) = \max_{1 \leq i \leq s} \{\boldsymbol{f}_i^T \cdot \boldsymbol{x}\} \tag{3}$$

induced by a *polyhedral* set of the form

$$P = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{f}_i^T \cdot \boldsymbol{x} \leq 1 \ , \ i = 1, \ldots, s\} \tag{4}$$

which is compact and contains the origin in its interior. Such functions can be shown to be proper and locally Lipschitz (see [39]) and decompose the state-space into a number of convex cones $\mathcal{C}_i = \{\boldsymbol{x} \in \mathbb{R}^n : V(\boldsymbol{x}) = \boldsymbol{f}_i^T \cdot \boldsymbol{x}\}$ with disjoint relative interiors $ri\{\mathcal{C}_i\} \cap ri\{\mathcal{C}_j\} = \emptyset \ , \ i \neq j$. Note that now P can be expressed as $P = \{\boldsymbol{x} \in \mathbb{R}^n : V_P(\boldsymbol{x}) \leq 1\}$. The linear functions $\boldsymbol{f}_i^T \cdot \boldsymbol{x}$ are called the *generators* of the PL function $V_P(\boldsymbol{x})$.

If the polyhedron P is bounded and centrally symmetric, then it is a *polytope* and V_P can be expressed as

$$V_P(\boldsymbol{x}) = \|F \cdot \boldsymbol{x}\|_\infty = \max_{1 \leq i \leq m} \{|\boldsymbol{f}_i^T \cdot \boldsymbol{x}|\} \ , \ F \in \mathbb{R}^{m \times n}, \, m \geq n \tag{5}$$

where $\|\cdot\|_\infty$ the infinity vector norm in $\mathbb{R}^n$. Figure 1 shows a centrally symmetric polytope and the cones corresponding to its faces.

The positive invariance principle and the use of positively invariant sets is vital in the construction of set-induced PLFs. A complete survey of their properties and usage for a series of problems in control theory can be found in [8]. We have the following definitions: We have the following definitions:

Definition 1. A polyhedral set P is called *positively invariant* (P.I.) with respect to the trajectories of a dynamical system if for all $\boldsymbol{x}(0) \in P$ the solution $\boldsymbol{x}(t) \in P$ for $t > 0$.

Definition 2. A polyhedral set P is called *ε-contractive* (with a *level of contractivity ε*) with respect to the trajectories of a dynamical system if for all $\boldsymbol{x}(0) \in P$ the solution $\boldsymbol{x}(t) \in P$ for $t > 0$ and, additionally $\exists \varepsilon > 0$ s.t.

$$\mathcal{D}^+ V_P(\boldsymbol{x}) \ \leq \ -\varepsilon \, V_P(\boldsymbol{x}) \ , \ \forall \boldsymbol{x} \in \partial P \ , \ \mathcal{D}^+ V_P(\boldsymbol{x}) = \lim_{\Delta \to 0^+} \frac{V(\boldsymbol{x}(t + \Delta)) - V(\boldsymbol{x}(t))}{\Delta} \tag{6}$$

where $V_P(\boldsymbol{x})$ the set-induced function (3) of P and $\mathcal{D}^+ V_P(\boldsymbol{x})$ the upper right Dini derivative of function $V(\boldsymbol{x})$ which replaces the classical derivative, since $V_P(\boldsymbol{x})$ is continuous but not continuously differentiable [36].

A positively invariant set guarantees stability and not asymptotic stability, i.e. periodic motions are also allowed, whereas ε-*contractivity* implies not only asymptotic stability but also exponential stability with decay rate ε.

Invariance (contractivity) for a polytope P implies stability (asymptotic stability) which can be shown with its set-induced PLF (3). The invariance conditions for P become existence conditions for (3).

For a single linear system

$$\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} \ , \ \boldsymbol{x} \in \mathbb{R}^n, \mathbf{A} \in \mathbb{R}^{n \times n} \tag{7}$$

and a polytope P which is compact, closed and containing the origin in its interior, the invariance conditions can be formulated via its set-induced PLF (3) as follows [12],[13]:

Lemma 1. *For the function (3) to be a Lyapunov function for system (7) (and the polytope P to be invariant under the dynamics of A) it is necessary and sufficient that*

$$\boldsymbol{f}_i^T \cdot \dot{\boldsymbol{x}} \leq 0 \ , \ \forall \boldsymbol{x} \in \partial P \ , \ i \in J(\boldsymbol{x}) \ , \ J(\boldsymbol{x}) = \{i \in \{1, \dots, m\} : \boldsymbol{f}_i^T \cdot \boldsymbol{x} = V_P(\boldsymbol{x})\} \tag{8}$$

where $J(\boldsymbol{x})$ is a set of indexes corresponding to active constraints.

It is easy to prove (see also Figure 1) that, for linear systems, the above invariance conditions can be also formulated in terms of the vertices of P, and then the active set of constraints is determined by the faces which intersect on the vertex in question. Thus, we have the following corollary (without proof):

Corollary 1. *For the function (3) to be a Lyapunov function for system (7) (and the polytope P to be invariant under the dynamics of A) it is necessary and sufficient that*

$$\boldsymbol{f}_i^T \cdot \dot{\boldsymbol{v}} \leq 0 \ , \ \forall \boldsymbol{v} \in vert\{P\} \ , \ i \in J(\boldsymbol{v}) \ , \ J(\boldsymbol{v}) = \{i \in \{1, \dots, m\} : \boldsymbol{f}_i^T \cdot \boldsymbol{v} = 1\} \tag{9}$$

A nice property of PL LFs is that invariance conditions on vertices are also necessary and sufficient for systems (1),(2).

Corollary 2. *For the function (3) to be a Lyapunov function for system (1) (and the polytope P to be invariant under the dynamics of (1)) it is necessary and sufficient that (8) is satisfied on the vertices of P for all linear subsystems involved.*

Molchanov and Pyatnitskii [36],[37] have proposed another equivalent compact formulation [44]:

Lemma 2. *The function (3) induced by P is a Lyapunov function for system (2) (absolute stability) iff there exist p matrices $M_1, M_2, \dots, M_p$, $M_i \in \mathbb{R}^{N \times N}$, each being strictly diagonal column dominant $(m_{ii} + \sum_{j=1, j \neq i}^{p} |m_{ji}| < 0)$ and satisfying*

$$A_i \cdot S = S \cdot M_i \tag{10}$$

where $S = [\boldsymbol{v}_1, \dots, \boldsymbol{v}_N] \in \mathbb{R}^{n \times N}$ is a matrix containing the vertices $\boldsymbol{v}_i \in vert\{P\}$, $i = 1, \dots, N$ of P.

It is straightforward to see that invariance conditions (9),(10) can be stated as a linear programming problem. This is the route taken in [23], [27] for PL uncertain systems and in [42],[44] for LDIs. As remarked later, the efficiency of direct implementation of (8),(9) using linear programs is severely limited by high computational complexity.

In a number of theoretical works, several authors [34],[35],[36], [1],[2],[3],[4],[37],[45] have considered the problem of absolute stability of an LDI and proved that the class of PLFs is *universal*, i.e. absolute stability of an LDI and existence of a common PLF function are equivalent. The existence of a PLF has also been found necessary and sufficient for the robust control of linear uncertain systems [7], and such functions have also been used for estimating transient responses [9] and generating smooth classes of universal LFs [10]. More recently the authors in [21] have studied the problem of the existence of a Lyapunov function for exponentially stable switching systems, and proved that a non-quadratic common Lyapunov function always exists. Finally, the problem of robust absolute stability of an LDI is reduced in [31] to the existence of a polyhedral vector norm type Lyapunov function.

Following the justification of the power of PLFs, a number of attempts have also been made to develop numerical techniques for the construction of such Lyapunov functions. Brayton and Tong [17], [18] designed an algorithm for difference inclusions which calculates a series of balanced polytopes converging to the level set of a common PLF after a finite number of steps. Via discretization the same technique can be applied to differential inclusions. Although there are convergence proofs, the technique suffers from high computational complexity due to the use of computationally expensive convex hull computations. Barabanov [5] proposed another technique for checking asymptotic stability of a linear differential inclusion. An algorithm is constructed which calculates in a finite number of steps the Lyapunov exponent and a common PLF. This idea has been developed initially for difference inclusions and requires a sufficiently dense discretisation and progressive refinements. Again convex hull computations increase the computational load significantly, rendering the techniques applicable to planar systems, as evidenced by the examples in [17], [18], [33],[5].

Ohta et al. in [39],[40] propose a numerical procedure to construct non-balanced polytopes that satisfy some invariance properties for a piecewise linear differential inclusion. The algorithm requires again the use of convex hull computations and the solution of many linear programming problems. Although the procedure is applicable to three-dimensional systems and can also cover the case of some subsystems being unstable in local regions, there is unfortunately no guarantee of convergence and the choice of the initial conditions is crucial.

Romanchuk [46] proposed an algorithm that approximates the region of attraction of a PL system with a convex polytope, hence calculating a PLF. This algorithm is cumbersome and has been shown to work in planar systems only. For higher-dimensional systems the initial choice is crucial and the class of polytopes considered (simple polytopes) is restrictive in the search for solutions.

Blanchini [7] uses the Fourier-Motzkin elimination technique for calculating controlled invariant polytopes, i.e. polytopes which can be made P.I. with admissible controls and applies them to the problem of robust controller synthesis under state and control constraints for linear uncertain systems. Yfoulis [50] developed another algorithm, the ray-gridding technique for the same problem for linear and PL uncertain systems. Both techniques can be applied to three-dimensional systems.

Johansson [23], [25],[24] has developed a useful framework for a unifying treatment of piecewise quadratic (PQ) and piecewise linear LFs for the more general class of piecewise linear systems on uniform partitions of the state space. Solutions can be found using convex optimization (Linear Matrix Inequalities for PQ LFs and linear programming for PL LFs). The computation of PL LFs requires solving large linear programs and is expected to be computationally involved for systems with state dimension $n \geq 3$.

In a similar fashion Julian et. al. [27] proposes the use of a high level canonical piecewise linear representation of nonlinear systems, interpolated on a systematic simplicial partition of the state-space, which uses the minimum and exact number of parameters required. A parametrization of all polyhedral Lyapunov candidates on the selected partition results in linear programming problems, with a large number of constraints and variables, exponentially increasing with the state dimension and the density of the grid. It is debatable whether the technique can be applied comfortably in dimension $n = 3$.

Finally, Polanski in [42] uses the algebraic stability conditions stated for a PLF of the infinity norm type in [34],[35],[36],[37] and a scaling idea to formulate the search for PLFs as a linear program. Similar numerical difficulties with high complexity arise and the technique is applicable to planar systems. In [44] an improved formulation using polytope vertices and scaling makes the technique applicable to three-dimensional problems, although there are limitations in the number of vertices that can be dealt with, as remarked later in this report. Thus there are cases, even in three dimensions, in which instability cannot be inferred even when a solution cannot be found. A method to overcome this shortcoming and detect instability is found in [43], although implementation to dimension $n \geq 3$ faces the same numerical difficulties.

In summary, from the previous discussion it is apparent that although powerful in theory, PLFs suffer from high computational complexity in practice, when numerical techniques are applied for their computation. Computationally demanding convex hull computations and/or large-scale linear programs are highly responsible for exponential increase in computational time and computer memory requirements, restricting their applicability to low-dimensional systems.

In this report the ray-gridding technique introduced in [49],[50] is applied successfully to Problem 1. It is shown that a simple and very efficient iterative procedure, with guaranteed convergence, can offer significant savings compared to all other techniques previously mentioned, which we believe allows consideration of medium-sized problems in state dimensions $n \geq 4$, reported for the first time in the literature. Although this is a subject of future research, it is also shown that it sets up a framework that can be extended to relevant problems in the analysis and design of linear switching systems, such as finding *stabilising switching sequences*. The ray-gridding technique is introduced in the next section.

3 Main results

Efficient construction of PLFs for solving Problem 1 is based on flexible search for invariant polytopes for all linear subsystems involved. For this purpose, uniform partitions of the state space which can be easily refined to an adjustable degree of complexity in the search for a solution are vital. For PL systems [23] the partition is usually imposed by the PL system description. For nonlinear systems approximated as PL or linear uncertain a typical choice is a uniform gridding of the state space and use of more general PL functions than polyhedral [27].

As opposed to standard gridding, *ray-gridding* of the state-space has been proposed in [50] in an attempt to build up a framework where families of polytopes of adjustable complexity and flexible representation could be used as invariant polytope candidates. These families of polytopes generated by ray-gridding and the corresponding set-induced polyhedral functions are suitable for linear switched systems. The *ray-gridding* technique operates in terms of rays.

3.1 Definitions

We begin with the following definitions:

Definition 3. A *ray partition* in $\mathbb{R}^n$ is a set $\mathcal{R} = \{r_i , i = 1, 2, \ldots, N\}$ of rays, where $r_i = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{x} = \lambda_i \cdot \boldsymbol{e}_i , \lambda_i \geq 0 , \boldsymbol{e}_i \in \mathbb{R}^n, \boldsymbol{e}_i \neq \boldsymbol{0}\}$. The vectors $\boldsymbol{e}_i$ which specify the rays are termed *ray vectors* and then any point on the ray r_i is uniquely determined by the non-negative scalar λ_i, referred to herein as its *scaling factor*. The number N of the rays in $\{r_i\}$ is the *order* of the ray partition. The *scaling vector* $\boldsymbol{\lambda} = [\lambda_1, \ldots, \lambda_N]^T$ is the collection of all scaling factors.

Definition 4. The following special subclasses of ray partitions in $\mathbb{R}^n$ are also defined. A ray-partition is

- *proper* when all its rays r_i are disjoint, or $r_1 \cap r_2 \cap \ldots \cap r_N = \{\boldsymbol{0}\}$, i.e. they only intersect at the origin. Otherwise it is called *improper*.

- *unit* if and only if all its ray vectors are unit vectors, i.e. their magnitude is equal to 1.

- *constrained* when all its scaling factors are bounded, i.e. there exist upper limits $\overline{\lambda}_i$ such that $0 \leq \lambda_i \leq \overline{\lambda}_i$.

- A constrained ray partition is called *normalized* if and only if all its scaling factors are bounded by 1, i.e. $\overline{\lambda}_i = 1 \ \forall i$.

- A ray partition is termed *symmetric* if and only if for every ray r_i there exists another ray r_j such that $\forall \boldsymbol{x} \in r_i , -\boldsymbol{x} \in r_j$.

Definition 5. A *ray-polytope* is a polytope R with non-empty interior (i.e. $\lambda_i > 0 \ \forall i$) that is compatible with a proper ray partition $\mathcal{R} = \{r_i\}$, i.e. each vertex $\boldsymbol{v}_j \in vert\{R\}, j = 1, \ldots, N_v$ belongs to one and only one of the rays in $\mathcal{R}$, i.e. $N = N_v$ and $\forall i = 1, \ldots N \ \exists j = 1, \ldots, N_v$ s.t. $\boldsymbol{v}_j = \lambda_i \boldsymbol{e}_i$.

Remark 1. It is easy to see that any constrained ray-partition can be made normalized by replacing the ray vectors $\boldsymbol{e}_i$ with new vectors $\hat{\boldsymbol{e}}_i = \boldsymbol{e}_i / \overline{\lambda}_i$.

Remark 2. In the presence of state bounds $\boldsymbol{x} \in \mathcal{X}$, a ray partition becomes constrained, and it can be normalised by using as new ray vectors $\hat{\boldsymbol{e}}_i$ the intersection of the rays and the boundary of $\mathcal{X}$, i.e. $\hat{\boldsymbol{e}}_i = r_i \cap \partial\mathcal{X}$.

Let us consider an n-dimensional convex polytope $P \in \mathbb{R}^n$. Then there always exists a *ray-polytope R* with a corresponding appropriately selected ray-partition and scaling factors λ_i, $i = 1, \ldots, N$, such that P and R coincide. Moreover, the ray partition $\{r_i\}$ specifies a whole family of polytopes $R(\boldsymbol{\lambda}) = conv\{\boldsymbol{x}_i = \lambda_i \boldsymbol{e}_i, i = 1, \ldots, N\}$ with vertices on the rays and variable scaling factors $\boldsymbol{\lambda} = [\lambda_1, \ldots, \lambda_N]^T$. For different selections for the scaling vector $\boldsymbol{\lambda}$ new scaled versions of $R(\overline{\boldsymbol{\lambda}})$, $\overline{\boldsymbol{\lambda}} = [\overline{\lambda}_1, \ldots, \overline{\lambda}_N]^T$ are produced. For sufficiently large N and uniform coverage of the whole state-space any shape can be approximated with arbitrary accuracy. With candidates from the class of the ray-polytopes $R(\boldsymbol{\lambda})$, $\boldsymbol{0} < \boldsymbol{\lambda} \le \overline{\boldsymbol{\lambda}}$ the search for invariant polytopes can be done very efficiently due to simplicity in the invariance conditions. For linear switched systems, centrally symmetric polytopes P, $\boldsymbol{0} \in ri\{P\}$ is a suitable choice, since $\forall \boldsymbol{x} : \boldsymbol{x} \in vert\{P\}$, $-\boldsymbol{x} \in vert\{P\}$, hence the invariance conditions in (8) need to be checked for half of the vertices only. For simplicity in the presentation of the ideas and the main issues, we deal initially with planar systems.

3.2 Planar case

Before proceeding further let us recall some basic results from the literature on infinity norm LFs. The existence of a Lyapunov function of the form (3) for a linear system $\dot{\boldsymbol{x}} = \mathbf{A} \cdot \boldsymbol{x}$, $\boldsymbol{x} \in \mathbb{R}^n$ is equivalent to the conditions of Lemma 2 and it has been proved [28],[41] that there always exists a PLF (3) that can be constructed using the transformation matrix that transforms the system into block diagonal form. For systems with eigenvalues inside the 45°-region ($|re(\mu_i)| \le |im(\mu_i)|$) a minimal number of $m = n$ generators is required (i.e. $\mathbf{F} \in \mathbb{R}^{n \times n}$ in (3)), whereas outside the 45°-region a number of generators proportional to the ratio $\frac{|re(\mu_i)|}{|im(\mu_i)|} \le 1$ is required ($re(\mu_i)$ and $im(\mu_i)$ are the real and imaginary part of an eigenvalue μ_i, respectively). The maximal number of generators required for a PLF (3), as well as upper estimates on their number are given in [12],[13]. In other words, for a single linear system, the previously mentioned results determine the number and position of suitable ray directions that can form a ray partition for which existence of a PLF is guaranteed.

For linear switched systems similar results do not yet exist. Recently, for switching between two stable linear systems in $\mathbb{R}^2$ some results by Wulff et. al. in [47] suggest those particular ray directions for which the existence of a PLF with a minimal number of $m = n$ generators (called *unic* LF in [47]) is guaranteed. These are the directions of *common flow*, i.e. rays on which the flow of both systems is collinear.

Lemma 3. *[47] Given two stable matrices $\mathbf{A}_1, \mathbf{A}_2$ in $\mathbb{R}^{2 \times 2}$ each having real eigenvalues, the constituent systems of the switched system (1) have a common unic Lyapunov function (along the directions of common flow) if the matrix pencil $\alpha \mathbf{A}_1 + (1 - \alpha) \mathbf{A}_2$ has complex eigenvalues for some $\alpha \in [0, 1]$ and if its eigenvalues lie in the interior of the 45°-region for all $\alpha \in [0, 1]$.*

However, the extension of the previous results to general switched systems in $\mathbb{R}^n$ with eigenvalues that may lie outside the 45°-region is a challenging but also very difficult problem. Until this problem is theoretically deeply understood and properly investigated for analytic solutions, efficient and systematic numerical techniques for solving Problem 1 are essential.

Intuition suggests that in state-space regions in which there is oscillatory motion, a large number of linear segments may be required for forming an invariant set –the ratio of imaginary to real part being a relevant measure– , while in other regions few segments may be sufficient. Points of common flow can be a good initial guess only. Ways for exploring the whole state space with sufficiently dense local and global ray populations are essential, since apart from special cases mentioned above, there is no way to know which subsets of the state-space require complex descriptions. Let us consider symmetric and unit ray partitions in $\mathbb{R}^2$

$$\mathcal{R} = \{r_i, i = 1, 2, \ldots, N, N = 2 \cdot q, q \ge 2\} \tag{11}$$

which specify families of closed and bounded ray-polytopes $R(\boldsymbol{\lambda})$, $\boldsymbol{0} < \boldsymbol{\lambda} \le \boldsymbol{1}$, $\boldsymbol{\lambda} \in \mathbb{R}^{q \times 1}$ containing the origin in their interior and being centrally symmetric w.r.t. the origin.

In the planar case, *uniform* ray partitions can be simply constructed in a number of ways. E.g. the family of *regular* polygons (N-gons) [51]

$$P(N) := conv \left\{ (\cos(\frac{2\pi k}{N}), \sin(\frac{2\pi k}{N})) \ , \ 0 \leq k \leq N \right\} \tag{12}$$

can cover uniformly the phase plane (and approximate the circumscribed unit circle) to any prescribed degree of accuracy, for sufficiently large N. Moreover, regular polygons induce ray partitions where the angles between consecutive rays are equal and there is a regular simplicial decomposition in equilateral equal triangles (Figure 2). These features make the families of regular polygons $P(N)$ good candidates in the search for invariant polytopes.
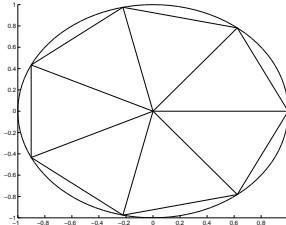
Figure 2: A regular polygon with 7 sides and its circumscribed unit circle.

3.2.1 Linear planar systems

For a solution to **Problem 1** for a single linear system $\dot{\boldsymbol{x}} = \mathbf{A} \cdot \boldsymbol{x}$ in $\mathbb{R}^2$ let us consider a regular polygon $P(N)$ and the ordered ray-partition $\mathcal{R}$ induced by it, where $\boldsymbol{e}_k = \left[\cos(\frac{2\pi k}{N}), \sin(\frac{2\pi k}{N})\right]^T$. $\mathcal{R}$ specifies a family of ray-polytopes $R_N(\boldsymbol{\lambda})$, $\mathbf{0} < \boldsymbol{\lambda} \leq \mathbf{1}$, $\boldsymbol{\lambda} \in \mathbb{R}^{N \times 1}$, which are scaled versions of $R_N(\mathbf{1}) = P(N)$. If a solution cannot be found with a polytope from $R_N(\boldsymbol{\lambda})$ progressive refinements with larger values for n which offer added flexibility can be considered. For the family $R_N(\boldsymbol{\lambda})$ we have the following:

Proposition 1. *A necessary and sufficient condition for the existence of an invariant ray-polytope $R_N(\boldsymbol{\lambda})$ under the dynamical flow of $\dot{\boldsymbol{x}} = \mathbf{A} \cdot \boldsymbol{x}$ in $\mathbb{R}^2$ is the existence of positive scaling factors $\lambda_k > 0, k = 1, \ldots, N$ s.t. the following set of conditions is satisfied*

$$\lambda_k \leq \Delta_k^{q(k)} \cdot \lambda_{q(k)} \ , \ k = 1, \ldots, N \tag{13}$$

where

$$\Delta_k^{q(k)} = \frac{det[\mathbf{A} \cdot \boldsymbol{e}_k, \boldsymbol{e}_{q(k)}]}{det[\mathbf{A} \cdot \boldsymbol{e}_k, \boldsymbol{e}_k]} \tag{14}$$

and $q(k)$ determines the ray $r_{q(k)}$ which together with the current ray r_k specify the conic sector $S(k, q(k)) = cconv\{\boldsymbol{e}_k, \boldsymbol{e}_{q(k)}\}$ which contains the velocity vector at $\boldsymbol{e}_k$ (for all points on ray r_k).

    Proof: The proof is based on Lemma 1 and Corollary 1 and is given in the Appendix.   ■

Remark 3. Note that only one condition of the form (13), parameterized by the two scaling factors λ_k , $\lambda_{q(k)}$ is imposed at each vertex of the ray-polytope $R_n(\boldsymbol{\lambda})$, although the active constraints in Lemma 1 are two. The derivative for points on the current ray r_k belongs to the single *active* sector $S(k, q(k))$, and the condition of this single sector is solely considered. If a ray direction coincides with an eigenvector line (such a ray is called *trivial*), there is no condition to be checked for invariance, since the velocity vector points inwards along the ray direction.

For planar systems, only few possibilities for the direction of the dynamical flow exist. For systems with complex eigenvalues the flow is either clockwise or counterclockwise in the whole phase plane, whereas for systems with real eigenvalues the flow changes direction across the eigenvector lines. Note that on the eigenvector lines the flow remains on them since the derivative points inwards along them. The notion of the dependency graph of a ray-partition is next introduced:

Definition 6. The *dependency graph* $\mathcal{G}_\mathcal{R}$ of a ray-partition $\mathcal{R}$ for a single linear system $\dot{\boldsymbol{x}} = \mathbf{A} \cdot \boldsymbol{x}$ in $\mathbb{R}^2$ is a planar directed graph (digraph) $\mathcal{G}_\mathcal{R} = (V, E)$ of order (number of vertices) n_v with n_v labeled nodes $v_i \in V, |V| = n_v$ and size (number of edges) n_e with arcs (edges) $e_j \in E$ directed from k to $q(k)$. The vertices (nodes) of the graph are labeled after the rays k and $q(k)$ as defined in Proposition 1. Hence, the dependency graph depicts the dependency of ray k to ray $q(k)$, as expressed algebraically in (13). Each arc e_j , $j = 1, \ldots, n_e$ carries a weight $w_j = \Delta_k^{q(k)}$.

Definition 7. If v and w are vertices of a directed graph (digraph), a *directed walk* from v to w is a finite sequence $v = v_0, e_1, v_1, e_2, v_2, e_3, \ldots, e_n, v_n = w$, of vertices and arcs of the digraph such that each arc in the sequence is an arc from v_{i-1} to v_i. If the arcs are distinct and no vertices repeat the same sequence is called a *directed path* v to w. A *closed walk* in a graph is a walk between a vertex and itself. A closed walk in which no edges repeat is a *circuit*. A *cycle* C is a circuit with no repeated vertices.

Definition 8. If v is a vertex of a dependency graph, the *cycle* $C = \{v = v_0, e_1, v_1, e_2, v_2, e_3, \ldots, e_n, v_n = v\}$ between v and itself is called *feasible* iff

$$\prod_{i=1}^{N} w_i \geq 1 \tag{15}$$

where w_i are the weights of the edges e_i , $i = 1, \ldots, N$

Proposition 2. *The dependency graph of a ray-partition of a linear system $\dot{\boldsymbol{x}} = \mathbf{A} \cdot \boldsymbol{x}$ in $\mathbb{R}^2$ contains at most two cycles, no loops, no isolated vertices for non-trivial rays and all vertices are of degree k , $k = 0, 1, 2$. If the system has complex eigenvalues the graph is 2-regular.*

    Proof: See the Appendix.   ■

    We have the following results for linear planar systems :

Corollary 3. *A necessary and sufficient condition for the existence of an invariant ray-polytope $R_N(\boldsymbol{\lambda})$ with* complex *eigenvalues under the dynamical flow of $\dot{\boldsymbol{x}} = \mathbf{A} \cdot \boldsymbol{x}$ in $\mathbb{R}^2$ is*

$$\prod_{k=1}^{N} \Delta_k^{q(k)} \geq 1 \tag{16}$$

Then there exist positive scaling factors $\lambda_k > 0, k = 1, \ldots, N$ s.t. (13) is satisfied.

    Proof: See the Appendix.   ■

Corollary 4. *A necessary and sufficient condition for the existence of an invariant ray-polytope $R_N(\boldsymbol{\lambda})$ with* real *eigenvalues under the dynamical flow of $\dot{\boldsymbol{x}} = \mathbf{A} \cdot \boldsymbol{x}$ in $\mathbb{R}^2$ is*

$$\Delta_{k_1}^{k_2} \Delta_{k_2}^{k_1} \geq 1 \tag{17}$$

for the indices $0 \leq k_1, k_2 \leq N$ which specify the cycles in the dependency graph of the ray partition. If no cycle exists, then no conditions are imposed for the existence of positive scaling factors $\lambda_k > 0$, $k = 1, \ldots, N$ s.t. (13) is satisfied.

    Proof: See the Appendix.   ■

    It is obvious from the previous results for a single linear system that the existence of a PLF from a family of ray-polytopes $R_N(\boldsymbol{\lambda})$ is reduced to checking simple conditions of the form (16) or (17). If these easily verifiable conditions are not satisfied, a larger family of ray-polytopes $R_{\hat{N}}(\boldsymbol{\lambda}) \supset R_N(\boldsymbol{\lambda})$, $\hat{N} > N$ can be considered until a solution is found.

    Note that satisfaction of (16) or (17) guarantees the existence of $\boldsymbol{\lambda}^* > 0$ s.t. $R_N(\boldsymbol{\lambda}^*)$ is invariant. This value can be easily calculated from conditions (13) with the help of the following iterative algorithm :

Algorithm 1. The algorithm consists of the following steps :

Step 1 : Consider a unit normalized ray-partition $\mathcal{R}$ induced by a regular polygon $P(N)$ and the corresponding family of ray-polytopes $R_N(\boldsymbol{\lambda})$, $\mathbf{0} < \boldsymbol{\lambda} \leq \mathbf{1}$, $\boldsymbol{\lambda} \in \mathbb{R}^{N \times 1}$, which are scaled versions of $R_N(\mathbf{1}) = P(N)$, $\mathbf{1} = [1, 1, \ldots, 1]^T \in \mathbb{R}^{N \times 1}$. Start from the initial polytope $R_N(\mathbf{1})$ in which the scaling factors $\lambda_i = 1$ of all rays have their maximum value.

Step 2 : Consider all rays one after another in their ordered sequence and update their scaling factors. When ray r_k , $k = 1, \ldots, N$ is visited, its corresponding condition (13) is checked and its scaling factor is left unchanged if (13) is satisfied or is reduced to a new value $\hat{\lambda}_k$ with $\hat{\lambda}_k = \Delta_k^{q(k)} \cdot \hat{\lambda}_{q(k)}$ if not.

Step 3 : The algorithm concludes when all scaling factors have converged to a constant value. Therefore, the stopping criteria is a check of convergence.

After visiting all rays in Step 2, the values of some scaling factors may have been reduced, thus (13) is not necessarily satisfied for all rays. The algorithm checks whether all conditions (13) are satisfied for the new updated values $\hat{\lambda}_k$. If they are satisfied, the algorithm stops and a solution is found. If not, the algorithm continues in a number of iterations, i.e. it returns to Step 2 and updates all scaling factors again until they have converged.

Step 4 : Since there is a lower limit ($=0$) for the scaling factors, when the algorithm stops the scaling factors have converged either to fixed positive values or to zero. In the first case a solution is found, while in the second it is proved that no invariant set from the family $R_N(\lambda)$ exists .

Proposition 3. *Algorithm 1 converges to an optimal solution $R_N(\boldsymbol{\lambda}^*)$ iff an invariant polytope from family $R_N(\boldsymbol{\lambda})$, $\mathbf{0} < \boldsymbol{\lambda} \leq \mathbf{1}$, $\boldsymbol{\lambda} \in \mathbb{R}^{N \times 1}$ exists.*

Proof. Necessity : Suppose there exists an invariant polytope from the family $R_N(\boldsymbol{\lambda})$, $\mathbf{0} < \boldsymbol{\lambda} \leq \mathbf{1}$, $\boldsymbol{\lambda} \in \mathbb{R}^{N \times 1}$ for some $\boldsymbol{\lambda}^* > 0$. Algorithm 1 starts from the upper bound $\boldsymbol{\lambda} = \overline{\boldsymbol{\lambda}} = \mathbf{1}$ and progressively reduces some of the scaling factors when this is found necessary in order to satisfy conditions (13). Since these are not only sufficient but also necessary for invariance, the steps followed by Algorithm 1 are also necessary for a solution. Algorithm 1 is a means for "scanning" the interior of the unit circle with scaled versions of ray-polytopes exhaustively, and therefore, it cannot miss to find $R_N(\boldsymbol{\lambda}^*)$ or a polytope arbitrarily close to $R_N(\boldsymbol{\lambda}^*)$.

Sufficiency : When Algorithm 1 finds a solution this obviously proves the existence of an invariant ray-polytope. The solution is also *optimal* in size, since a ray-polytope with maximal scaling factors for all rays is obviously found. This is true because the algorithm starts from the upper bound and reduces the scaling factors when this is necessary. Note that restriction of the polytope families considered in the interior of the unit circle (which is selected as our working domain) does not restrict the applicability of the technique. For linear systems, if an invariant polytope not included in the unit circle exists, there always exists a scaled version of it in our working domain. $\square$

The following Theorem summarizes the previous results for stable linear systems in $\mathbb{R}^2$:

Theorem 1. *A necessary and sufficient condition for the existence of a PLF for a single linear system $\dot{\boldsymbol{x}} = \mathbf{A} \cdot \boldsymbol{x}$ in $\mathbb{R}^2$ is the existence of a sufficiently large n s.t. conditions (13) are satisfied, or equivalently that Algorithm 1 converges to a positive solution $\boldsymbol{\lambda}^* > 0$.*

Proof. The existence of a PLF from the family $R_N(\boldsymbol{\lambda})$, $\mathbf{0} < \boldsymbol{\lambda} \leq \mathbf{1}$, $\boldsymbol{\lambda} \in \mathbb{R}^{N \times 1}$ for a single stable linear system $\dot{\boldsymbol{x}} = \mathbf{A} \cdot \boldsymbol{x}$ in $\mathbb{R}^2$ is guaranteed for sufficiently large n, since any invariant set shape can be approximated with arbitrary accuracy with a regular polygon. Algorithm 1 is operating on the basis of the necessary and sufficient conditions (13) therefore it can locate the invariant ray-polytope. $\square$

Remark 4. Algorithm 1, as described above, operates on the equality in conditions (13) and therefore checks stability (and not asymptotic stability), i.e. it specifies invariant (Definition 1) and not contractive (Definitions 2,3) polytopes. However, Algorithm 1 can be easily modified to search for contractive polytopes if conditions (13) are modified accordingly to reflect (6) for some contractivity factor $\varepsilon > 0$. All basic results can be also extended to this case.

Remark 5. The basic advantage of Algorithm 1 is that, although it implements an exhaustive "scanning", the simplicity of the conditions (13) checked guarantees efficiency and low computational complexity, even when large families of rays are considered or a large number of iterations are required for a solution to be found. Fortunately, the same idea can be extended to linear switched systems of any dimension.

The previous results are illustrated with two simple examples :

Example 1. [real eigenvalues] The following LTI systems have real eigenvalues, $\sigma(\mathbf{A}_1) = \{-2, -3\}$ and $\sigma(\mathbf{A}_2) = \{-1.17, -0.53\}$, where $\sigma(A)$ the spectrum of matrix $\mathbf{A}$.

$$\mathbf{A}_1 = \begin{bmatrix} -2 & 0 \\ 0 & -3 \end{bmatrix} \quad , \quad \mathbf{A}_2 = \begin{bmatrix} -1.8 & 1.0 \\ -0.8 & 0.1 \end{bmatrix} \tag{18}$$

For system $\mathbf{A}_1$ and the 4 trivial rays (eigenvector lines) in Figure 3 no conditions are imposed and an invariant polytope is found for maximal scaling factors. This is the case where a minimal number of generators $m = 2$ is sufficient for a PLF.

A solution with 8 non-trivial and uniformly distributed rays for system $\mathbf{A}_2$ in (18) is shown in Figure 4.

Example 2. [complex eigenvalues] The following LTI system has complex eigenvalues, $\sigma(\mathbf{A}) = \{-0.5 \pm j \cdot \sqrt{\frac{4k+7}{4}}\}$.

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -(2+k) & -1 \end{bmatrix} \tag{19}$$

For $k = 0$ a solution with 16 uniformly distributed rays is shown in Figure 5. Condition (16) is satisfied with $\prod_{k=1}^{N} \Delta_k^{q(k)} = 1.57 \geq 1$.

For $k = 2$ a solution with 32 uniformly distributed rays is shown in Figure 6. Condition (16) is satisfied with $\prod_{k=1}^{N} \Delta_k^{q(k)} = 1.50 \geq 1$. An increase in the number of rays is naturally expected for larger values of k, since they specify a smaller ratio $\frac{|re(\mu_i)|}{|im(\mu_i)|} \leq 1$ of real to imaginary part of the eigenvalues μ_i. The solutions shown are no optimal in terms of number or position of rays. Upper estimates on the number of rays required are given in [12],[13]. For single linear systems, optimal solutions can be easily specified analytically [41]. Unfortunately, these results do not extend to switched linear systems, hence numerical techniques with uniform distributions cannot be avoided.
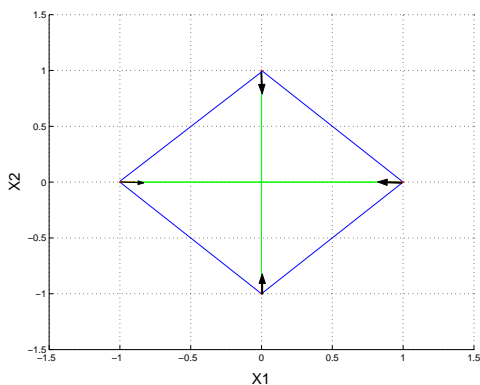

Figure 3: An invariant polytope with 4 trivial rays corresponding to system $\mathbf{A}_1$ in (18).
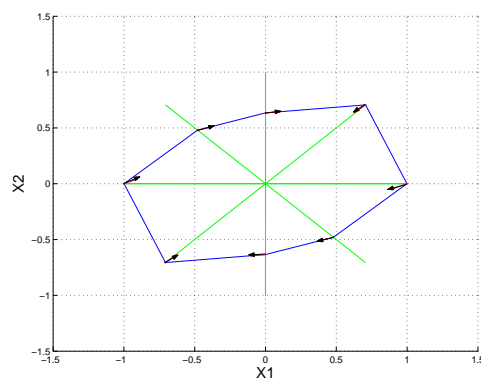
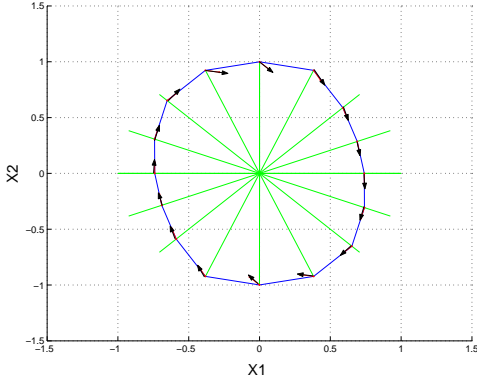Figure 4: An invariant polytope with 8 non trivial rays corresponding to system $\mathbf{A}_2$ in (18).

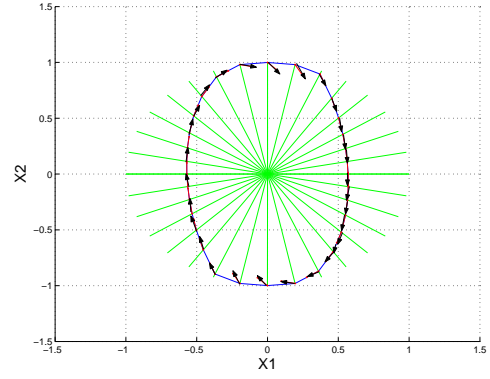Figure 5: An invariant polytope with 16 rays corresponding to system $\mathbf{A}$ in (19) with $k = 0$.

Figure 6: An invariant polytope with 32 rays corresponding to system $\mathbf{A}$ in (19) with $k = 2$.

3.2.2 Linear planar switched systems

The ray-gridding technique outlined in the previous section for linear systems has a number of attractive features: it reduces Problem 1 to easily verifiable conditions. These conditions can be checked for refinable families of ray-polytopes and if satisfied for a sufficiently large family, the scaling vector is easily calculated using Algorithm 1, which is extremely efficient in $\mathbb{R}^2$ since it is based on very simple calculations.

However, although the existence of an invariant polytope is theoretically guaranteed, a very important practical consideration is the computational complexity required for its calculation. For linear systems, there is fortunately an upper bound for n as remarked in [12],[13], which assures that the computational burden is limited. This fact has been observed in Example 2.

The main results obtained for linear systems using the ray-gridding idea can be extended to linear switched systems. However, an upper bound on the computational complexity is not yet available, and it is shown later in this section with an example that the complexity can grow arbitrarily under certain circumstances.

Proposition 1 can be easily extended to switched systems (1) or LDIs (2) in $\mathbb{R}^2$, where conditions of the form (13) have to be simultaneously satisfied for all linear subsystems involved.

Proposition 4. *A necessary and sufficient condition for the existence of an invariant ray-polytope $R_N(\boldsymbol{\lambda})$ under the dynamical flow of system (1) in $\mathbb{R}^2$ is the existence of positive scaling factors $\lambda_k > 0, k = 1, \ldots, N$ s.t. the following set of conditions is satisfied*

$$\lambda_k \leq \Delta_k^{q_i(k)} \cdot \lambda_{q_i(k)} \; , \;\; k = 1, \ldots, N \; , \;\; i = 1, \ldots, p \tag{20}$$

where

$$\Delta_k^{q_i(k)} = \frac{det[\mathbf{A}_i \cdot \boldsymbol{e}_k, \boldsymbol{e}_{q_i(k)}]}{det[\mathbf{A}_i \cdot \boldsymbol{e}_k, \boldsymbol{e}_k]} \tag{21}$$

and $q_i(k)$ determines the ray $r_{q_i(k)}$ which together with the current ray r_k specify the conic sector $S(k, q_i(k)) = cconv\{\boldsymbol{e}_k, \boldsymbol{e}_{q_i(k)}\}$ which contains the velocity vector at $\boldsymbol{e}_k$ (for all points on ray r_k) for the linear subsystem $\dot{\boldsymbol{x}} = \mathbf{A}_i \cdot \boldsymbol{x}$ of (1).

The dependency graph $\mathcal{G}_{\mathcal{R}}$ for a linear switched system is defined similarly as in Definition 6 and has in principle different characteristics. It is basically the superposition of the individual dependency graphs $\mathcal{G}_{\mathcal{R}}^{(i)}$ for all subsystems involved in (1) and it can be much more complex. It may contain many cycles and end-vertices and have vertices with degrees as high as $2 \cdot p$. However, feasibility can be still inferred by the cycles present in the graph. We have the following results :

Proposition 5. *The existence of an invariant polytope from the family $R_N(\boldsymbol{\lambda})$ for a linear switched system (1) in $\mathbb{R}^2$ is equivalent to the feasibility of all cycles present in its dependency graph.*

Proof: This is an obvious result. Provided that a sensible ray-partition has been selected, i.e. one for which all weights $\Delta_k^{q_i(k)} > 0$ are positive, there always exist values $\lambda_i > 0$ for all nodes of the

13

graph satisfying (20) if all cycles are feasible. To see this, assume all cycles are feasible. Then (15) is satisfied for all weights involved and therefore fixed values $\lambda_i > 0$ for the corresponding rays can be specified. Then all other rays can be assigned their values $\lambda_i > 0$ from the weights of the graph. Conversely, the feasibility of all cycles is also necessary. An infeasible cycle corresponds to infeasibility of the conditions involving its rays and hence the absence of an invariant polytope induced by the ray-partition chosen. ∎

When all subsystems involved have complex eigenvalues of the same direction, the conditions are significantly simplified :

Corollary 5. *If all subsystems for a linear switched system (1) in $\mathbb{R}^2$ have complex eigenvalues and are of the same direction, the existence of an invariant polytope from the family $R_N(\boldsymbol{\lambda})$ is equivalent to the conditions*

$$\prod_{k=1}^{N} \min_{i} \{\Delta_k^{q_i(k)}\} \geq 1 \ , \ k = 1, \ldots, N \ , \ i = 1, \ldots, p \tag{22}$$

Proof: Obvious, since satisfaction of (16) for a number of subsystems is possible if and only if (22) is satisfied. ∎

Similarly to the linear case, the scaling vector can be calculated using the following algorithm :

Algorithm 2. This algorithm differs from Algorithm 1 only in Step 2, where a number of conditions for all linear subsystems involved are checked. The algorithm starts again from the initial polytope $R_N(\mathbf{1}) = P(N)$ in which the scaling factors $\lambda_i = 1$ of all rays have their maximum value and visits progressively all rays in their ordered sequence in a number of iterations. When a ray is visited, all corresponding conditions (20) of all subsystems are checked and its scaling factor is reduced, if necessary, in order to satisfy all simultaneous invariance conditions. After visiting all rays, the algorithm continues in the same fashion until all scaling factors have converged to fixed positive values or reached zero. In the first case a solution is found, while in the second it is proved that no invariant set from the family $R_N(\boldsymbol{\lambda})$ exists.

The following result is also a straightforward extension of Proposition 3 :

Proposition 6. *Algorithm 2 converges to an optimal solution $R_N(\boldsymbol{\lambda}^*)$ iff an invariant polytope from family $R_N(\boldsymbol{\lambda})$, $\mathbf{0} < \boldsymbol{\lambda} \leq \mathbf{1}$, $\boldsymbol{\lambda} \in \mathbb{R}^{N \times 1}$ exists.*

Theorem 1 is next extended to linear switched systems :

Theorem 2. *A necessary and sufficient condition for the existence of a PLF for the linear switched system (1) in $\mathbb{R}^2$ is the existence of a sufficiently large N s.t. conditions (20) are satisfied, or equivalently that Algorithm 2 converges to a positive solution $\boldsymbol{\lambda}^* > 0$.*

An illustrative example is next given :

Example 3. We consider an example from [32] in which the absolute stability problem for planar systems is studied and the "most destabilizing" nonlinearity which corresponds to a closed trajectory (periodic motion) is characterized. We have the system

$$\dot{x}_1 = x_2 \ , \ \dot{x}_2 = -2x_1 - x_2 - \phi(t, x_1) \tag{23}$$

where $\phi(t, x_1)$ is a scalar time-varying function in the sector $[0, k]$, that is, $\phi(t, 0) = 0$ and $0 \leq z\phi(t, z) \leq kz^2$ for all $t \geq 0$. This system can be restated in switched linear form [32] as

$$\dot{\boldsymbol{x}} \in conv\{\mathbf{A}, \mathbf{B}_k\} \ , \quad \mathbf{A} = \begin{pmatrix} 0 & 1 \\ -2 & -1 \end{pmatrix} \ , \quad \mathbf{B}_k = \begin{pmatrix} 0 & 1 \\ -(2+k) & -1 \end{pmatrix} \tag{24}$$

Problems 2 and 3 defined below are considered in [32] :

Problem 2. Find the value $k^* = \inf\{k \geq 0 : \exists \phi^*$ for which system (23) is not asymptotically stable$\}$

Problem 2 is equivalent to Problem 3 :

Problem 3. Find the value $k^* = \inf\{k \geq 0 : \exists \phi^*$ for which the switched system (24) is not asymptotically stable$\}$

Using the notion of generalised first integrals and a numerical solution to the resulting equation the authors in [32] obtain the value $k^* = 6.98513$. Thus, a necessary and sufficient condition for asymptotic stability is is $k < k^* = 6.98513$.

We have considered a different problem, Problem 1, which can check stability of the linear switched system (24) provided k is given. The solution to Problem 3 can be approximated by an iterative application of Problem 1 using e.g. bisection. Both systems $\mathbf{A}$ and $\mathbf{B}_k$ are of the same direction of flow and possess complex eigenvalues, hence for each candidate value k Theorem 2 and the simple conditions in Corollary 5 can be applied and progressively larger values for n can be used until a solution is found. The value of the product $\Pi = \prod_{k=1}^{N} \min_i\{\Delta_k^{q_i(k)}\} \geq 1$, $k = 1, \ldots, N$, $i = 1, \ldots, p$ in (22) is calculated for different values of k and n in the search for a solution to Problem 3 using bisection (only half of the rays are considered due to symmetry). The results are shown in Table 1. The limiting polytope is shown in Figure 7.

k	N	Π	Time (secs)	k	N	Π	Time (secs)
5	50	0.67	0.00	6.98	10000	0.99873	0.72
5	100	1.16	0.00	6.98	20000	0.99966	0.87
6	100	0.79	0.00	6.98	40000	1.00012	1.13
6	150	0.94	0.00	6.985	10000	0.99447	0.74
6	200	1.04	0.00	6.985	50000	0.99892	1.35
6.9	200	0.77	0.00	6.985	100000	0.99948	2.83
6.9	500	0.917	0.02	6.985	200000	0.99976	5.77
6.9	700	0.947	0.02	6.985	300000	0.99985	8.77
6.9	1000	0.97	0.04	6.985	500000	0.99993	14.21
6.9	1600	0.99	0.05	6.985	800000	0.99997	23.11
6.9	2000	0.997	0.06	6.985	1000000	0.99998	28.28
6.9	2200	1.004	0.06	6.985	1500000	1.000001	43.05

Table 1: Calculation of $\Pi = \prod_{k=1}^{N} \min_i\{\Delta_k^{q_i(k)}\}$ in (22) for system (24) in [32]
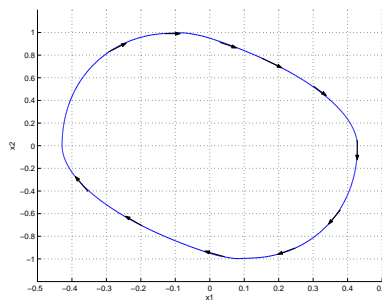

Figure 7: Invariant polytope found for example 3 (periodic motion).

Solving Problem 3 is much more computationally demanding than solving Problem 1. The reason is not only that many iterative attempts to solve Problem 1 are required, but also another problematic issue revealed by this experiment. In Table 1 we note that as the value k approaches $k^* = 6.98513$ the number of sides n of families of scaled polygons becomes very large (we needed 40000 rays for a two-digit accuracy with $k = 6.98$ and 1500000 rays for a three-digit accuracy with $k = 6.985$) before a solution can be found. Hence, although the number of attempts is fixed by the accuracy required, at each attempt an increased number of sides n needs to be considered. Although the

computational complexity is not significant for planar systems as can be seen from the computational time requirements in Table 1, we certainly expect significant increase of the computational burden in higher dimensions, which will restrict the applicability of any numerical technique for calculating PLFs and solving Problem 3. This issue has been remarked also in [11] (Section 3.5) for the iterative algorithm used in [17], [18] and certainly applies to all similar procedures.

3.3 Ray-gridding in $\mathbb{R}^n$, $n > 2$

The main results obtained for planar systems can be extended to higher-dimensional families of ray-polytopes $R(\boldsymbol{\lambda})$ in $\mathbb{R}^n$. Inspired by the use of regular polytopes in the planar case –which ensure a uniform coverage of the phase plane by approximating uniformly the unit circle– we would like to consider similar families $R(\boldsymbol{\lambda})$ of ray-polytopes which correspond to *uniform simplicial* partitions of the unit n-sphere $S_n = \{\boldsymbol{x} \in \mathbb{R}^n : \|\boldsymbol{x}\| = 1\}$. Strictly speaking, uniform partitions of the unit n-sphere cannot be achieved in the same sense as in $\mathbb{R}^2$, since there are only few regular polytopes in $\mathbb{R}^n$. However, nearly uniform simplicial subdivisions of the unit n-sphere with corresponding refinable ray-partitions can be generated.

3.3.1 Simplicial partitions in $\mathbb{R}^n$, $n > 2$

Global uniform partitions We propose a technique based on regular simplicial subdivision of a hypercube in $\mathbb{R}^{n-1}$, followed by a "lifting" map which results in a corresponding regular simplicial subdivision of the boundary of a crosspolytope in $\mathbb{R}^n$. All points generated on the boundary are subsequently pulled up to the unit n-sphere forming a nearly regular simplicial subdivision. All vertices on the unit n-sphere induce a ray-partition and the corresponding families of ray-polytopes $R(\boldsymbol{\lambda})$ in $\mathbb{R}^n$ are invariant polytope candidates.

For our families of ray-polytopes $R(\boldsymbol{\lambda})$ in $\mathbb{R}^n$ a decision has been made to work with *simplicial* polytopes, because this simplifies the facial structure and permits easier handling of the geometric objects formed. Our invariance conditions are imposed on the boundary of ray-polytopes. A simplicial decomposition of a boundary in $\mathbb{R}^{n-1}$ guarantees uniformity in the conditions imposed for each simplex, since the neighboring vertices involved are a priori known and are not modified at each step of the iterative algorithm.

If $\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ are $(n+1)$ points in $\mathbb{R}^n$, a *simplex* $S(\boldsymbol{x}_0, \ldots, \boldsymbol{x}_n)$ is their convex combination

$$S(\boldsymbol{x}_0, \ldots, \boldsymbol{x}_n) = \left\{ \boldsymbol{x} : \boldsymbol{x} = \sum_{i=0}^{n} \mu_i \boldsymbol{x}_i \ , \ 0 \le \mu_i \le 1 \ , \ \sum_{i=0}^{n} \mu_i = 1 \right\} \tag{25}$$

It is called *proper* iff it cannot be contained in an (n-1)-dimensional hyperplane $H = \{\boldsymbol{x} : \boldsymbol{n}^T \boldsymbol{x} = c\}$, where c a constant.

Systematic simplicial subdivisions of n-dimensional rectangular domains have a long history in the circuits community, starting from Chien and Kuh [19] and continuing with Chua [20] and recently Julian [27], [26]. The procedure consists of a tessellation into small rectangles (producing a grid of points and regions) followed by a canonical decomposition of each region into simplices. This approach is useful for piecewise linear modelling and the parametrization of PL LFs, as shown in [27]. However, it is not useful, as it is, for our purpose of checking invariance on (n-1)-dimensional boundaries of candidate ray-polytopes. We propose another technique for generating nearly uniform simplicial subdivisions on (n-1)-dimensional boundaries :

- **Step 1 :** Start from a regular simplicial subdivision $\Delta_{(n-1)}$ (as proposed in [19]) of a hypercube in $\mathbb{R}^{n-1}$ placed symmetrically around the origin, with an adjustable grid size δ.

- **Step 2 :** Apply a *lifting* map $\mathcal{L} : \mathbb{R}^{n-1} \to \mathbb{R}^{n-1}$ which "lifts" all points from the rectangular grid to the boundary of a crosspolytope $C_n^\Delta \in \mathbb{R}^n$. The result is a regular simplicial subdivision of the boundary ∂C_n^Δ. The lifted points covering the state-space nearly uniformly induce a ray-partition $\mathcal{R}$.

16

- **Step 3 :** All ray-points of $\mathcal{R}$ are "pulled-up" along their rays to the boundary of the unit n-sphere to form a unit normalized ray-partition. Repeating steps 1-3 with progressively smaller grid sizes generates denser ray-partitions and corresponding polytopes that can approximate the unit n-sphere to any prescribed degree of accuracy.

In Step 1 we start from an initial symmetric rectangular domain $D_n(L)$ in $\mathbb{R}^n$

$$D_n(L) = \{\boldsymbol{x} : -L \cdot \mathbf{1} \le \boldsymbol{x} \le L \cdot \mathbf{1}\} \ , \ \mathbf{1} = [1, \ldots, 1]^T \in \mathbb{R}^n \ , \ L \in \mathbb{Z}^+ \tag{26}$$

which corresponds to a carving up of the state-space with hyperplanes parallel to the main axes $x_i = c_i \ , \ c_i = -L, \ldots, 0, \ldots, L$ into $N(L) = (2 \cdot L)^n$ unit hypercubes with vertices $V = \{\boldsymbol{x} : x_i = c_i\}$.

We next follow Brandts [14] to construct uniform partitions Δ_n of $D_n(L)$ into n-simplices satisfying the following uniformity condition : All unit hypercubes are decomposed into $n!$ path simplices such that n of the $\frac{1}{2}n(n+1)$ edges of each simplex coincide with n orthonormal edges of the hypercube. This construction generalizes the partition of the unit square K^2 into two congruent triangles S_{12}, S_{21}

$$S_{12} = \{\boldsymbol{x} \in \mathbb{R}^2 : 0 \le x_1 \le x_2 \le 1\} \ , \ S_{21} = \{\boldsymbol{x} \in \mathbb{R}^2 : 0 \le x_2 \le x_1 \le 1\} \tag{27}$$

to $\mathbb{R}^n$. The unit hypercube is decomposed into n-simplices

$$S_\sigma = \{\boldsymbol{x} = (x_1, \ldots, x_n) \in \mathbb{R}^n : 0 \le x_{\sigma(1)} \le \ldots \le x_{\sigma(n)} \le 1\} \ , \ \sigma \in \Sigma^n \tag{28}$$

where Σ^n is the group of all $n!$ permutations of the numbers $1, \ldots, n$. A decomposition of a unit cube in $\mathbb{R}^3$ into path simplices is shown in Figure 8.

In the uniform simplicial decomposition described the simplices around any vertex (which will specify the invariance conditions) can be easily determined and the construction also guarantees that all simplices have equal volume. A similar procedure for the three-dimensional case (without an explicit construction procedure applicable in $\mathbb{R}^n$) has been implemented in [44].

In Step 2 a "lifting" map $\mathcal{L} : \mathbb{R}^{n-1} \to \mathbb{R}^n$ is applied

$$\boldsymbol{x} = [x_1 \, x_2 \ldots x_{n-1}]^T \in \mathbb{R}^{n-1} \longrightarrow \pm\hat{\boldsymbol{x}} = \frac{1}{L} \cdot \left[x_1 \, x_2 \ldots x_{n-1} \pm (L - \max_i |x_i|)\right]^T \in \mathbb{R}^n \tag{29}$$

which "lifts" all points from the rectangular grid to the boundary of a crosspolytope $C_n^\Delta \in \mathbb{R}^n$. A crosspolytope $C_n^\Delta = \{\boldsymbol{x} \in \mathbb{R}^n : \sum_i |x_i| \le 1\}$ is the equivalent of an octahedron in $\mathbb{R}^3$ or a rhombus in $\mathbb{R}^2$. The result is a regular simplicial subdivision of the boundary ∂C_n^Δ. The grid size is $\delta = \frac{1}{L}$. The vectors $\pm\hat{\boldsymbol{x}}$ in (29) can be shown to belong to the boundary of a transformed crosspolytope $\partial\hat{C}_n^\Delta = \{\boldsymbol{x} \in \mathbb{R}^n : \sum_i |y_i| \le 1\}$, $\boldsymbol{y} = \mathbf{T} \cdot \boldsymbol{x}$ for appropriate $\mathbf{T}$. Points with $\max_i |x_i| = k$, $0 \le k \le L$ are said to belong to the $(L-k)$-th layer. Hence, points on the boundary of the initial rectangular grid –with $\max_i |x_i| = L$– belong to the 0-th layer and have $x_n = 0$, i.e. they are not lifted and therefore remain at their initial starting position. All other intermediate points in the grid –with $\max_i |x_i| = k$, $0 \le k < L$– are lifted to the $(L-k)$-th layer and have $x_n = \frac{L-k}{L}$. The origin is lifted to the top vertex of the crosspolytope which satisfies $x_n = 1$ and constitutes the uppermost L-th layer.

Finally in Step 3 all points from $\partial\hat{C}_n^\Delta$ are mapped to the boundary of the unit n-sphere forming a nearly uniform simplicial subdivision, provided that the grid size is sufficiently small. The simplicial facial structure constructed in Step 1 is now transferred to the boundary of the unit sphere in $\mathbb{R}^n$. This construction allows the efficient application of the ray-gridding approach in high dimensions and the specification of invariant ray-polytopes in the interior of the unit n-sphere.

The "lifting" procedure is illustrated for $n = 2$ and $n = 3$ in Figure 10.

Edgewise simplex subdivisions Another policy frequently used for performing partition refinements is to start from a regular initial choice (a unit hypercube or crosspolytope) and proceed to progressive refinements using successive edgewise subdivisions. This process can operate globally as well as locally, allowing denser refinement in subsets of the state-space, in which more flexibility is perhaps required, while leaving other subsets with coarser partitions. This policy can avoid a great deal of redundant operations, but it requires a priori good knowledge of the problem for a successful
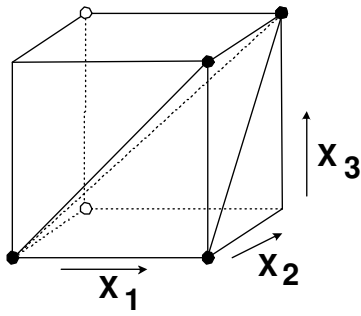
Figure 8: The path simplex $\sigma = (2, 3, 1)$ following the path of orthogonal edges of the cube in respective directions x_1, x_3 and x_2.
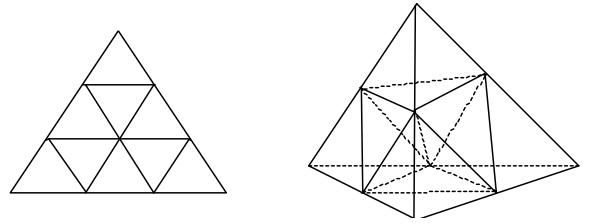
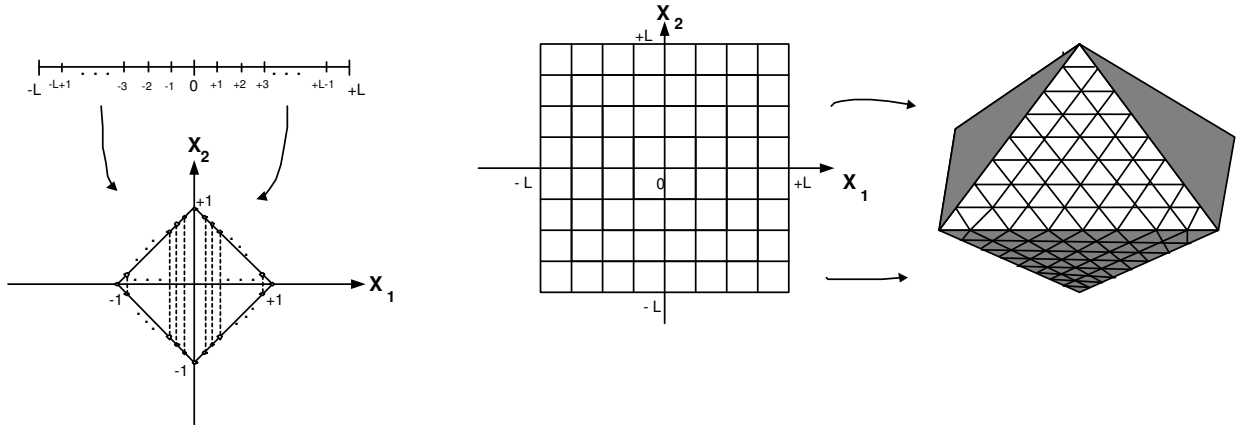Figure 9: Edgewise subdivisions of a triangle and a tetrahedron

Figure 10: The lifting procedure for $n = 2$ and $n = 3$.

application (see e.g. Example 4). The main difficulty is to build a logic intelligent enough to avoid getting stuck in cumbersome local search which can increase the computational burden. Local refinement strategies can perhaps produce results with much coarser ray partitions in some cases, but they can be also significantly slower when dense global partitions are required and their efficiency may also be largely dependent on the initial choice. It is therefore advisable to apply a global refinement strategy with uniform partitions at first in order to check feasibility. If a solution is guaranteed one can apply local refinement strategies to search for another solution with a minimal number of rays.

Efficient techniques for edgewise subdivision of a simplex have been proposed in [38] and [22]. In Figure 9 edgewise subdivisions of a triangle and a tetrahedron are shown. Note the similarity with the result of "lifting" points from uniform simplicial rectangular partitions. Similar ideas of subdividing edges have been proposed in [23].

3.3.2 Invariance conditions

All basic results can be extended from planar to linear switched systems (1) or LDIs (2) in $\mathbb{R}^n$:

Proposition 7. *A necessary and sufficient condition for the existence of an invariant ray-polytope* $R(\boldsymbol{\lambda})$, $\boldsymbol{\lambda} \in \mathbb{R}^{N \times 1}$ *under the dynamical flow of system (1) in* $\mathbb{R}^n$ *is the existence of positive scaling factors* $\lambda_r > 0$, $r = 1, \dots, N$ *s.t. the following set of conditions is satisfied*

$$\sum_{k=1}^{n} \delta_k^{(i)} \cdot \lambda_{k_1^{(i)}} \lambda_{k_2^{(i)}} \dots \lambda_{k_{n-1}^{(i)}} \leq 0 \tag{30}$$

where $\delta_k^{(i)}$, $i = 1, \dots, p$, $k = 1, \dots, n$ *appropriate coefficients and* $k_j^{(i)}$ *(with* $k_j^{(i)} \neq k_l^{(i)}$ *for* $j \neq l$, $j, l = 1, \dots, n-1$) *determine the rays* $r_{k_j^{(i)}}$ *which specify the active conic sector (simplex) which contains the velocity vector for all points on the current ray for the linear subsystem* $\dot{\boldsymbol{x}} = \mathbf{A}_i \cdot \boldsymbol{x}$ *of (1).*

18

The dependency graph $\mathcal{G}_\mathcal{R}$ for a linear switched system in $\mathbb{R}^n$ is defined similarly and has in principle vertices with higher degrees, since each vertex is associated with $(n-1)$ different neighboring vertices in the active simplex. Unfortunately, feasibility cannot be directly inferred by the cycles present in the graph, since Definition 8 is not extendable to $n \geq 3$.

Similarly to the planar case, the scaling vector can be calculated using the following iterative algorithm :

Algorithm 3. The algorithm starts again from the initial polytope $R_n(\mathbf{1})$ generated by a uniform simplicial decomposition of the unit n-sphere in which the scaling factors $\lambda_i = 1$ of all rays have their maximum value and visits progressively all rays in their ordered sequence in a number of cycles. When a ray is visited, all corresponding conditions (30) of all subsystems are checked and its scaling factor is reduced, if necessary, in order to satisfy all simultaneous invariance conditions. After visiting all rays, the algorithm continues in the same fashion until all scaling factors have converged to fixed positive values or reached zero. In the first case a solution is found, while in the second it is proved that no invariant set from the family $R_N(\boldsymbol{\lambda})$ exists .

Proposition 8. *Algorithm 3 converges to an optimal solution iff an invariant polytope from the family of ray-polytopes considered exists.*

Theorem 3. *A necessary and sufficient condition for the existence of a PLF for the linear switched system (1) in $\mathbb{R}^n$ is the existence of a sufficiently large N s.t. conditions (30) are satisfied, or equivalently that Algorithm 3 converges to a positive solution $\lambda^* > 0$.*

Remark 6 (computational complexity). The computational complexity of Algorithm 3 in $\mathbb{R}^n$ increases exponentially with the dimension n of the problem. The iterative algorithm is performed in three steps.

The first step of constructing a uniform partition of the unit n-sphere and building the incidence relations of neighboring rays in simplices is less demanding compared to the next two steps. However, the number of rays considered, which is $2 \cdot N = (1 + 2 \cdot L)^{n-1}$, is exponentially increasing with the grid size L and dimension n. The next two steps are found to be the most demanding. The second step is the determination of the active rays for all subsystems involved (which specify the active simplices in (30)) and the third is the iterative procedure in which the scaling factors are progressively reduced until they converge to a constant value.

The second part is performed once only. All incident simplices to a ray have to be considered one after another before the active one is found. The number of incident simplices to a ray are $n!$ (with the exception of vertices belonging to layers 0 and L, which have a reduced number). An upper bound of the tests that need to be carried out at the second step is $N_t = p \cdot n! \cdot N = \frac{1}{2} p \cdot n! \cdot (1 + 2 \cdot L)^{n-1}$. However, usually a fraction of N_t tests is only required, since when a test is successful the remaining tests –for the same ray– are omitted. Detection of the active simplices is necessary in order to keep the number of conditions N_c checked in the next step (Step 3) polynomially increasing with the number of rays N.

The computational requirements of the third step depend on the number of iterations N_I. Since p conditions –p is the number of subsystems in (1)– are checked for each ray (one for each subsystem), the total number of conditions checked in Step 3 are $N_c = N_I \cdot p \cdot N = \frac{1}{2} p \cdot N_I \cdot (1 + 2 \cdot L)^{n-1}$. The third part is most demanding when the scaling vector found is small or converges to zero. In this case a large number N_I of iterations may be required before the scaling factors approaches its final value.

The iterative algorithm described in this report has been implemented in a number of examples in 2 and 3 dimensional problems. Planar examples can be tackled very efficiently, even for very large populations of rays, as reported in Example 3 for a solution to the very demanding Problems 2 and 3 (absolute stability). Implementation in three dimensional problems has also shown very good results in terms of computational efficiency, and a significant improvement compared to previous techniques, as evidenced by Example 5 in the next section.

Remark 7 (working domain). The reader may have observed the use of *unit* ray-partitions for the implementation of the iterative algorithms 1, 2 and 3 described in the previous sections. This fact results in the consideration of ray-polytopes which are contained in the closure of the unit n-sphere,

since the algorithms start with polytopes inscribed on the unit n-sphere and progressively shrink them to the interior. It is important to note that such restriction is chosen for computational simplicity and does not affect the efficiency of the technique. It can be easily proved that, due to linear dynamics for all subsystems involved, if an invariant polytope P not fully contained in the unit n-sphere is found, then there always exists a scaled version of it that can be found using our iterative algorithms for a sufficiently dense ray partition.

3.3.3 Examples

Example 4. [rays of common flow] We consider a linear switched system (1) with two stable subsystems

$$\mathbf{A}_1 = \begin{bmatrix} 0.3 & 0.7 \\ -2.3 & -2.3 \end{bmatrix} \quad , \quad \mathbf{A}_2 = \begin{bmatrix} -1.8 & 1.0 \\ -0.8 & 0.1 \end{bmatrix} \tag{31}$$

For this system it has been shown [47] that there does not exist a PLF with the minimal number of rays $N = 4$, because the eigenvalue-locus of $\mathbf{A}_1 + \mu \mathbf{A}_2$, $0 \le \mu \le 1$ is outside of the 45°-region. Starting with the two axes as an initial choice for ray directions and refining uniformly in steps in the search of a solution we obtain the polytope depicted in Figure 11 with 32 rays.

There exist two points of common flow between the two systems. Using these points as an initial choice and a selective local refinement strategy based on edgewise subdivision (section 3.3.1) we obtain the polytope depicted in Figure 12 with 14 rays. Note the selective placement of new rays in areas where they are needed. The velocity vectors on all rays are also shown in solid arrows for system $\mathbf{A}_1$ and larger arrows for $\mathbf{A}_2$. The common velocity points are the four corner vertices of the polytope.

Example 5. [absolute stability] Absolute stability of the following LDI in $\mathbb{R}^3$

$$\mathbf{A}_1 = \begin{bmatrix} -10 & -10\,\alpha & -10\,\alpha \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad , \quad \mathbf{A}_2 = \begin{bmatrix} -10 & -10\,\beta & -10\,\beta \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \tag{32}$$

has been considered in [44], where α has been assigned a fixed value $\alpha = 0.2$. The problem of finding the maximum value of β for absolute stability using a PLF gave the result $\beta = 1.00$ with a polytope constructed with 40 layers. Note that the circle criterion (quadratic function) gives $\beta = 0.5467$. This example is appropriate for testing the efficiency of the ray-gridding technique and for comparison with existing techniques.

The results in [44] reveal that the implementation using the formulation in (10) with linear programming suffers from high computational complexity. Indeed, experiments with more than 40 layers fail due to high memory requirements, since –although being sparse– extremely large matrices are required. Implementation of the ray-gridding technique in this example revealed its superiority both in terms of computational time and complexity of the polytope structures. The ray-gridding technique is much faster and polytopes with up to $L = 250$ layers (with 2×125001 vertices) can be dealt with, although the computational time increases significantly compared to smaller values for L. The results found for different values for L and β are shown in Table 2 [1]. Computational time versus number of layers L is plotted in Figure 15. For $\alpha = 0.2$, $\beta = 0.6$ and $L = 15$ layers the polytope found is depicted in Figure 13. The maximal value found with $L = 200$ layers (2×80001 vertices) is $\beta = 1.57$ with a two digit accuracy and is shown in Figure 14. The value $L = 200$ has been found a good choice for checking absolute stability, resulting in a good trade-off between computational complexity and efficiency in finding solutions.

We observe significant improvement (from $\beta = 1.0$ with 40 layers to $\beta = 1.57$ with 200 layers) in the maximum value found for β which assures absolute stability. Although partitions with $L = 200$ layers correspond to polytopes with a very complex structure (2×80001 vertices), Figure 15 suggests that such polytopes can be found in relatively reasonable computational times. This implies that the technique suggested can provide complete solutions even in very demanding cases and problems, such as absolute stability. Moreover, the computational complexity measures discussed in the previous

---

[1]The computational times shown have been obtained from experiments with Matlab v.6.5 running on a Pentium 4, 1.8 GHZ personal computer
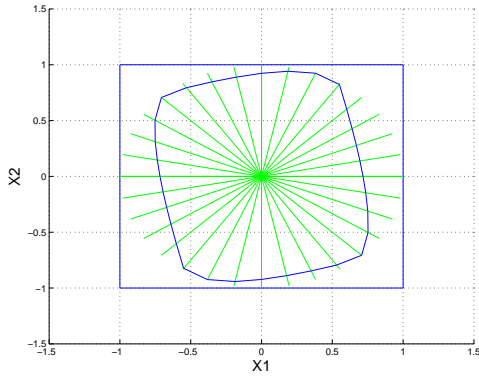
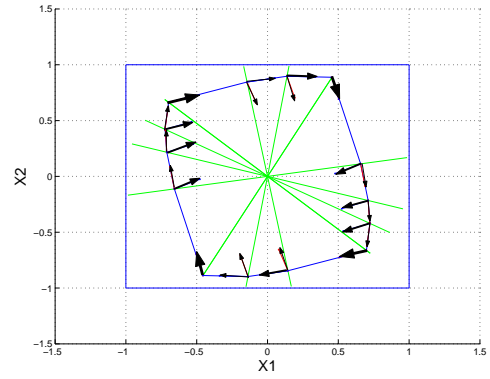Figure 11: An invariant polytope with 32 rays corresponding to uniform global refinement.

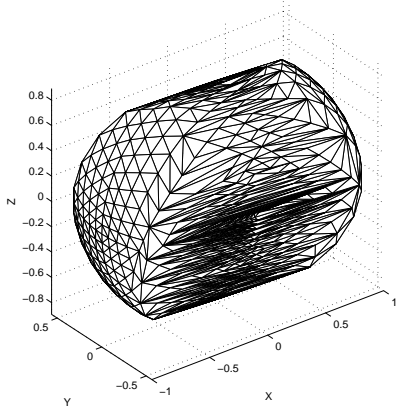Figure 12: An invariant polytope with 14 rays corresponding to selective local refinement.

Figure 13: An invariant polytope (after computing the convex hull) with 15 layers corresponding to the LDI (32) with $\alpha = 0.2, \beta = 0.6$.
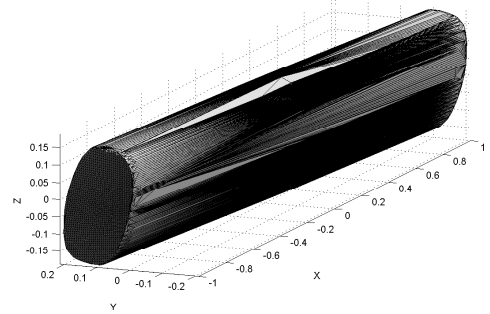

Figure 14: An invariant polytope(after computing the convex hull) with 200 layers corresponding to the LDI (32) with $\alpha = 0.2, \beta = 1.57$.

section suggest that application of the technique in higher dimensions $n = 4$ or $n = 5$ can be carried out for medium-sized problems, i.e. those that can be resolved with a reduced number of layers and corresponding polytopes of limited complexity.

4 Existence of stabilising switching sequences for switched systems with unstable subsystems

In the previous sections the ray-gridding approach has been applied to the problem of calculating common PLFs for stable linear subsystems via the use of a simple and efficient iterative algorithm. The extension of the same ideas to unstable subsystems and the problem of calculating stabilising switching sequences is investigated in this section.

Associated to any ray-partition formed by a uniform simplicial decomposition as described in Section 3.3.1 there is a corresponding family of ray-polytopes. A family of candidate *switching domains* $S_{ij}^D = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{x} \in \mathcal{C}_i \cap \mathcal{C}_j\}$, $i, j \in \mathcal{I}$, $i \neq j$ which lie in the intersection of two neighboring convex cones (simplices) $\mathcal{C}_i, \mathcal{C}_j$, $i \neq j$ of the simplicial decomposition can be considered for the following problem :

Problem 4. Check the existence of stabilising switching sequences for system (1) in $\mathbb{R}^n$, where all subsystems are unstable, and the switching domains are the common boundaries between simplices in the simplicial decomposition.

In Proposition 7 the conditions of the form (30) are checked for all subsystems since the existence of a common PLF is investigated. For a solution to Problem 4 satisfaction of (30) for a single subsystem
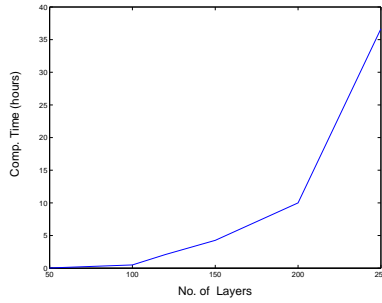
21

Figure 15: Computational time versus number of layers for Example 5.

Description of the ray-polytope R		Result of applying the algorithm			
No. of layers $2L =$	No. of vertices $2N =$	Succeeded for $\beta =$	Time (mins)	Failed for $\beta =$	Time (mins)
2×50	2×5101	1.0	2.18	1.5	3.5
2×100	2×20001	1.2	30	1.5	55
2×150	2×45001	1.5	213	1.7	256
2×200	2×80001	1.55	675	1.6	610
2×200	2×80001	1.57	620	1.5	580
2×250	2×125001	1.5	1560	2.0	2200

Table 2: Implementation results of the ray-gridding technique for the absolute stability of the LDI with vertex matrices (32) for $\alpha = 0.2$ and changing values for β and L.

is necessary and may also be sufficient for an invariant polytope under the dynamical flow of the switched system (1) for unstable subsystems. Hence, Proposition 7, Algorithm 3 and Theorem 3 can be extended to provide solutions to Problem 4.

4.1 Planar case

For planar systems, the switching domains proposed coincide with the rays of the ray-partition. We have the following :

Proposition 9. *A necessary condition for the existence of an invariant ray-polytope $R(\boldsymbol{\lambda})$, $\boldsymbol{\lambda} \in \mathbb{R}^{N \times 1}$ under the dynamical flow of system (1) in $\mathbb{R}^2$ for unstable subsystems is the existence of positive scaling factors $\lambda_k > 0, k = 1, \ldots, N$ s.t. the following condition is satisfied for each ray r_k and its associated subsystem $\dot{\boldsymbol{x}} = \mathbf{A}_{i(k)} \cdot \boldsymbol{x}$*

$$\lambda_k \leq \Delta_k^{q_i(k)} \cdot \lambda_{q_i(k)} \ , \ k = 1, \ldots, N \ , \ i = 1, \ldots, p \tag{33}$$

where

$$\Delta_k^{q_i(k)} = \frac{det[\mathbf{A}_i \cdot \boldsymbol{e}_k, \boldsymbol{e}_{q_i(k)}]}{det[\mathbf{A}_i \cdot \boldsymbol{e}_k, \boldsymbol{e}_k]} \tag{34}$$

and $q_i(k)$ determines the ray $r_{q_i(k)}$ which together with the current ray r_k specify the conic sector $S(k, q_i(k)) = cconv\{\boldsymbol{e}_k, \boldsymbol{e}_{q_i(k)}\}$ which contains the ve0locity vector at $\boldsymbol{e}_k$ (for all points on ray r_k) for the linear subsystem $\dot{\boldsymbol{x}} = \mathbf{A}_{i(k)} \cdot \boldsymbol{x}$ of (1). Conditions (33) with the following additional conditions

$$\Delta_{q_i(k)}^k \cdot \Delta_k^{q_i(k)} \geq 1 \ , \ k = 1, \ldots, N \ , \ i = 1, \ldots, p \tag{35}$$

are sufficient for the existence of an invariant polytope and a stabilising switching sequence.

Proof: The existence of a Lyapunov function for asymptotically stable switching systems is shown by a converse Lyapunov theorem in [21]. This result implies that if there exists a stable switching

22

sequence based on the switching domains defined above, then there exists a Lyapunov function to show stability, although its form and degree cannot be specified in [21]. PLFs can approximate any shape with arbitrary accuracy, thus the existence of a stabilising switching sequence is equivalent to the existence of a PLF, in theory.

We have investigated the problem of existence of a common PLF in the previous sections using sufficiently dense ray-partitions which allowed solutions to be found. In this section we consider switching systems with a priori specified switching domains specified by the ray-partition chosen. The same ray-partition induces a corresponding family of polytopes. The necessary and the sufficient conditions for the existence of an invariant polytope from this family under the switched dynamics are obviously (33) and (35), respectively. Conditions (33) impose invariance on the switching domains (i.e. on the vertices of the ray-polytopes) and are only *necessary* for an invariant polytope, whereas (33),(35) impose invariance for all points on the boundary of the ray-polytopes and are *sufficient* for an invariant ray-polytope.

Although this family is not universal, it can provide solutions as shown in the next examples. Moreover, by increasing sufficiently the density of the ray-partition and the number of switching domains, the flexibility of the PLF Lyapunov function candidates is increased and at some stage one expects to find an invariant polytope if a stabilising switching sequence exists. This implies the following result. ∎

Theorem 4. *A necessary and sufficient condition for the existence of a stabilising switching sequence on the switching domains induced by the selected ray-partition is the existence of an invariant ray-polytope for a sufficiently dense ray-partition.*

The dependency graph $\mathcal{G}_\mathcal{R}$ for a linear switched system with unstable subsystems can be also of great help in identifying different possibilities. Specifying a subgraph containing all rays and feasible cycles only is equivalent to finding a stabilising switching sequence.

Proposition 10. *A necessary condition for the existence of an invariant ray-polytope for a linear switched system (1) in $\mathbb{R}^2$ with unstable subsystems is the existence of a subgraph of its dependency graph containing all rays and feasible cycles only.*

The scaling vector and the switching strategy can be also calculated using the following iterative algorithms :

Algorithm 4 (necessary type). The algorithm operates on the basis of the necessary conditions (33) in Proposition 9. It starts from the initial polytope $R_N(\mathbf{1})$ generated by a uniform simplicial decomposition of the unit circle in which the scaling factors $\lambda_i = 1$ of all rays have their maximum value and visits progressively all rays in their ordered sequence in a number of iterations. When a ray is visited, the corresponding *necessary* conditions (33) of all subsystems are checked and the subsystem which specifies the smaller decrease for the current scaling factor is chosen. This choice can be interpreted as switching to the subsystem which imposes the less restrictive conditions. After visiting all rays, the algorithm continues in the same fashion until all scaling factors have converged to fixed positive values or reached zero. In the first case a solution is found, while in the second it is proved that no invariant set from the family $R_N(\lambda)$ exists. This implies that no solution to Problem 4 exists for the family of switching domains corresponding to the ray-partition chosen. Denser ray-partitions should be checked.

Algorithm 5 (sufficient type). This modified algorithm operates on the basis of the sufficient conditions in Proposition 9. When a ray is visited, the corresponding *sufficient* conditions (33) and (35) of all subsystems are checked and the subsystem which specifies the smaller decrease for the current scaling factor is chosen. This choice can be interpreted as switching to the subsystem which imposes the less restrictive conditions.

Proposition 11. *Algorithm 4 converges to a solution if an invariant polytope induced by the selected ray-partition exists.*

Proof: This result follows directly from Proposition 10. If an invariant polytope exists, then a subgraph exists and Algorithm 4 cannot miss it, since all possibilities are covered based on the necessary conditions. Note that there are cases in which Algorithm 4 may find a solution which does not correspond to a stabilising switching sequence. Moreover, the polytopes specified by a solution found by Algorithm 4 are not necessarily invariant. They are invariant only if (35) is additionally satisfied. ∎

Proposition 12. *If Algorithm 5 converges to a solution then there exists a stabilising switching sequence based on the family of switching domains induced by the selected ray-partition.*

Proof: If Algorithm 5 converges to a solution then the sufficient conditions (33) and (35) in Proposition 9 are satisfied. This implies that the polytopes specified by a solution found by Algorithm 5 are invariant. Hence the existence of a stabilising switching sequence is guaranteed. ∎

Remark 8. The switching strategy implied by the previous arguments chooses to switch to a linear subsystem $\mathbf{A}_{ij}$ when one of the boundaries $S_{ij}^D = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{x} \in \mathcal{C}_i \cap \mathcal{C}_j\}$, $i, j \in \mathcal{I}$, $i \neq j$ is hit. Each boundary is associated to a single subsystem. It might be the case that for a number of conic regions switching to the same subsystem is identified. Unification of all these conic domains is possible. Moreover, for neighboring regions associated with switching to different subsystems identification of sliding modes or sliding-like motion is possible. If undesirable, avoidance of sliding modes can be incorporated into the switching strategies considered in the iterative algorithm.

We consider three planar switched systems from [48] and show that Algorithms 4 and 5 can find stabilising switching sequences.

Example 6. The following LTI systems are both unstable foci of clockwise direction.

$$\mathbf{A}_1 = \begin{bmatrix} 1 & 13 \\ -2 & 3 \end{bmatrix} \quad , \quad \mathbf{A}_2 = \begin{bmatrix} -1 & 2 \\ -10 & 3 \end{bmatrix} \tag{36}$$

After appropriate refinement (16 rays) a solution is found. The ray-partition considered and the polytope found are shown in Figure 16. The velocity vectors on all rays are also shown in solid arrows for system $\mathbf{A}_1$ and larger arrows for $\mathbf{A}_2$. By unifying the conic sectors for which switching to the same subsystem has been identified, two switching lines L_1, L_2 shown dotted are obtained. This result is very close to the result found in [48]. The dependency graph of the switched system is shown in Figure 17 (Only half of the rays are considered due to symmetry). It can be checked that the solution found (shown on the right graph of Figure 17) is the only one corresponding to a feasible cycle with $\Pi = \prod_{k=1}^{8} \Delta_k = 1.29 > 1$ and satisfies the sufficient conditions (33),(35). The existence of stabilising switching sequences is proved.

Example 7. The following LTI systems are both unstable foci of opposite directions.

$$\mathbf{A}_1 = \begin{bmatrix} -2 & 52 \\ -8 & 6 \end{bmatrix} \quad , \quad \mathbf{A}_2 = \begin{bmatrix} 11 & -10 \\ 50 & -9 \end{bmatrix} \tag{37}$$

A solution with 16 rays is shown in Figure 18. The velocity vectors on all rays are also shown in solid arrows for system $\mathbf{A}_1$ and larger arrows for $\mathbf{A}_2$. By unifying the conic sectors for which switching to the same subsystem has been identified, two sectors S_1, S_2 shown dotted are obtained, where the motion is directed to and sliding-like motion follows. The dependency graph of the switched system is also shown in Figure 19 (Only half of the rays are considered due to symmetry). It can be checked that the solution found is one corresponding to a feasible cycle (nodes 2,3), and there are also other 2 solutions (with the cycle between nodes 1,2 or between nodes 1,8). Note that nodes 5 and 6 are not connected, which implies that either subsystem can be chosen in their conic sector S_3, S_4. The trajectories are attracted by sectors S_1, S_2 and then a stable sliding-like motion (chattering with finite frequency) occurs. This motion is reflected in the dependency graph by the presence of a feasible cycle between the corresponding rays (nodes 2,3). The strategy selected is shown in the right graph of Figure 19. Note that in the solution found only the necessary conditions (33) are satisfied. This

implies that the polytope found is *not* invariant (there exist boundary points for which invariance fails). However, the switching law considered is still asymptotically stable thanks to stable chattering motion. This fact is reflected by its non-convex shape.

Hence, although the necessary conditions (33) in Algorithm 4 cannot prove the existence of stabilising switching sequences, they can still be useful in practice. In this example Algorithm 4 found a solution satisfying the necessary conditions which with further investigation proved the existence although it doesn't specify an invariant polytope. Since stabilising switching sequence exist in this example, the existence of invariant ray-polytopes is guaranteed by Theorem 4 for sufficiently dense ray-partitions. Denser partitions can also reveal further possibilities for stable switching.

Example 8. The following LTI systems are both saddle points.

$$\mathbf{A}_1 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad , \quad \mathbf{A}_2 = \begin{bmatrix} -1.5714 & -0.8571 \\ 1.7143 & 1.5714 \end{bmatrix} \tag{38}$$

A solution with 8 rays is shown in Figure 20. The velocity vectors on all rays are also shown in solid arrows for system $\mathbf{A}_1$ and larger arrows for $\mathbf{A}_2$. The dependency graph of the switched system is also shown in Figure 21 (Only half of the rays are considered due to symmetry). Note that node 4 is the eigenvector line for system $\mathbf{A}_1$ and thus a simple path without any cycles can be chosen to prove stability (right graph in Figure 21). The switching strategy implied is simple : switch to system $\mathbf{A}_2$ until the eigenvector line of $\mathbf{A}_1$ is approached, then switch to $\mathbf{A}_1$. Again only the necessary conditions are satisfied.
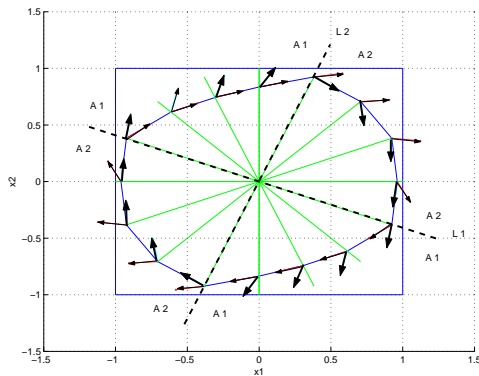

Figure 16: An invariant polytope corresponding to a stabilising switching sequence for (36)
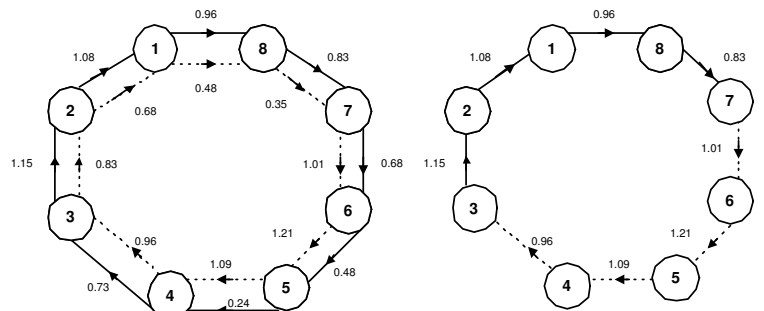

Figure 17: The dependency graph of (36) left) and the switching strategy chosen (right).
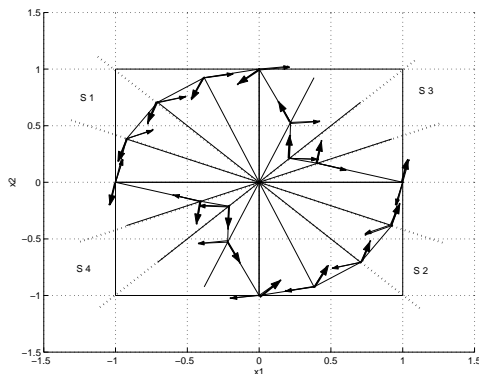

Figure 18: A non-invariant polytope specifying a stabilising switching sequence for (37)
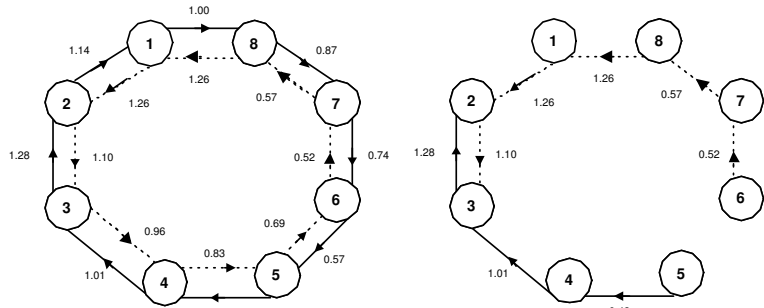

Figure 19: The dependency graph of (37) left) and the switching strategy chosen (right).
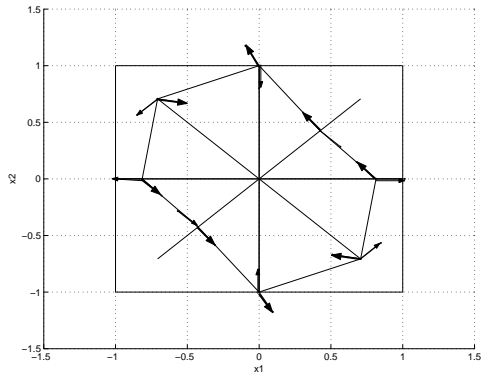
Figure 20: A non-invariant polytope specifying a stabilising switching sequence for (38)
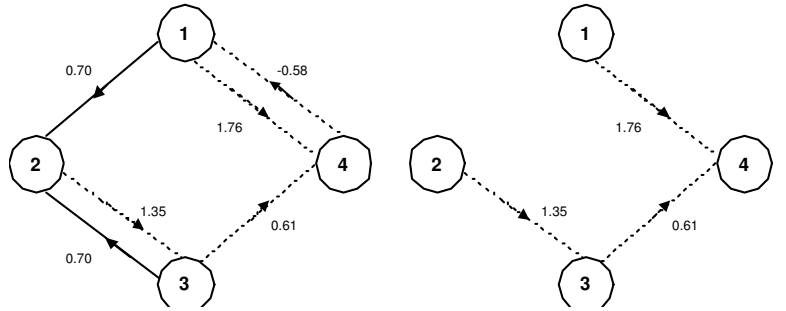
Figure 21: The dependency graph of (38) (left) and the switching strategy chosen (right).

4.2 Extension to higher dimensions

The switching domains considered in the previous section for planar systems are the ray lines and therefore consideration of the usual conditions (30) for ray points covers all points on the switching boundaries and is extremely efficient even for very large populations of rays due to simplicity of the conditions. The same ideas can be trivially extended to $\mathbb{R}^n$ if the switching boundaries remain the ray lines. However, this is not a reasonable choice for non-planar systems. One obvious choice could be the (n-1)-dimensional boundaries between neighboring simplices of the simplicial decomposition. However, in this case invariance conditions cannot be imposed in terms of a single ray. Further work is required for investigating whether an extension of the ideas in higher dimensions is possible.

5 Calculation of multiple Lyapunov functions for linear switched systems with stable subsystems

When a linear switched system (1) with stable subsystems does not admit a common LF, this implies that destabilising switching sequences exist, even if all subsystems are stable [16], [30]. In this case, a stability analysis requires finding the family –or a subset– of all switching sequences that result in stable systems. Such sequences can be specified using the *multiple* Lyapunov function idea [15],[16], [6],[29]. Such a possibility is investigated in this section.

In [29], the authors propose stabilising switching control laws for discrete and continuous-time linear systems using PLFs. In this section we extend the applicability of the methodology introduced in [29] using the ray-gridding approach developed in the previous sections.

5.1 Multiple PLFs and associated problems

The proposed methodology in [29] is based on the assumption that each individual subsystem i admits a PL LF $V_i(\boldsymbol{x})$ with the minimal number of generators $m = n$ –i.e. in the form of a transformed hyper-rectangle in dimension n–, so that efficient computation of the PL LFs is guaranteed. Then a multiple Lyapunov function is defined by $V(\boldsymbol{x}) = V_i(\boldsymbol{x})$ for the time interval for which subsystem i is active. If for any switching from subsystem i to subsystem j the inequality

$$V_j(\boldsymbol{x}) \leq V_i(\boldsymbol{x}) \tag{39}$$

is satisfied then the switching system is stable. Identification of all conic domains for which the previous inequality is satisfied specifies the switching law proposed. When the state reaches a region in which (39) is satisfied then obviously switching from subsystem i to subsystem j is safe and produces a stabilising switching sequence. Switching laws based on (39) for all subsystems involved are easily specified. An algorithm is proposed to compute the domains

$$\Omega_i^j = \{\boldsymbol{x} \in \mathbb{R}^n : V_j(\boldsymbol{x}) \leq V_i(\boldsymbol{x})\} \tag{40}$$

26

as unions of polyhedral cones. The cones are the regions defined by the ray partitions of the invariant polytopes which correspond to the PL LFs of the individual subsystems. Since, in general, a different ray-partition for each subsystem is selected, the algorithm has to construct a new composite ray-partition which is the union of the individual ray-partitions and iterate through all conic sectors of the composite partition to specify subsets of them that satisfy (39) for all pairs of indexes i, j. The regions Ω_i^j in (40) are then specified as unions of conic subsectors.

Although this methodology is simple and can pave the way for future work in the area, its applicability is limited by a number of issues that need to be addressed :

- It is explicitly mentioned in [29] that the efficiency of the technique is based on the efficient computation of the individual functions. For a single linear system inside or outside the 45°-degree region straightforward analytical computation of the PLFs is possible [28],[41]. There is no need to invoke the technique proposed in [17],[18], which suffers from high computational complexity for non-planar systems. However, the PLFs found is one possible choice and there is no guarantee that they can specify a large class of stabilising switchings sequences.

- The applicability of the technique depends on the size and shape of the invariant polytopes specified. Appropriate normalisation (scaling) is required so that the algorithm can find non-trivial results. The technique is not robust, since for different scaling and different initial polytopes different results are found. Methods for specifying optimal or suboptimal or refined choices of individual Lyapunov functions that can specify larger classes of stabilising switching laws are not proposed.

- Chattering motion is excluded, although it is one of the basic stabilising mechanisms for switching systems. Moreover, for each conic sector specified, switching to a *single* linear system is only allowed, although situations in local regions in which arbitrary switching between two or more subsystems can result in stable trajectories are usual, e.g. stable chattering motion with finite or infinite frequency.

The previous remarks suggest that the class of stabilising switching sequences identified by the technique proposed is limited and depends on the Lyapunov functions $V_i(\boldsymbol{x})$ chosen initially. Moreover, techniques for generating efficient and flexible LFs that can specify larger classes of stabilising switching laws are not provided.

The following example reveals some of the issues mentioned above :

Example 9. We consider the example used in [29], a switched continuous-time system $\dot{\boldsymbol{x}} = \mathbf{A}_p \cdot \boldsymbol{x}$, $p \in \{1, 2\}$, where

$$\mathbf{A}_1 = \begin{bmatrix} 1.7 & 1.8 \\ -4.5 & -3.7 \end{bmatrix} \quad , \quad \mathbf{A}_2 = \begin{bmatrix} 0.7 & -1 \\ 1.6 & -1.7 \end{bmatrix} \tag{41}$$

Both subsystems have eigenvalues in the 45°-degree region and thus admit PLFs with a minimal number of generators. The PLF functions used in [29] are $V_1(\boldsymbol{x}) = \max\{|x_1 + x_2|, |2x_1 + x_2|\}$ and $V_2(\boldsymbol{x}) = \max\{|-x_1 + x_2|, |2x_1 - 0.5x_2|\}$. In Figure 22 (upper left picture) the invariant polytopes P_1, P_2, $P_1 \cap P_2 \neq \emptyset$ corresponding to $V_1(\boldsymbol{x}), V_2(\boldsymbol{x})$ and the result returned by the technique are shown. The conic partition and the number of the subsystem (to which we can safely switch) specified at each region are also shown. If instead of P_1 a scaled version $2 \cdot P_1$ is used, a different result is obtained (upper right picture in Figure 22).

For scaled versions of P_1 or P_2 for which one of the polytopes is included in the interior of the other, a trivial result is returned. In the lower left picture of Figure 22 switching to subsystem 1 is only possible (since $V_1(\boldsymbol{x}) < V_2(\boldsymbol{x}) \; \forall \boldsymbol{x}$), while in the lower right picture of Figure 22 the opposite is true.

5.2   Extension of ray-gridding to multiple PLFs

Application of the ray-gridding technique for the calculation of the multiple LFs (39) and the regions (40) can provide solutions to many of the aforementioned problematic issues.
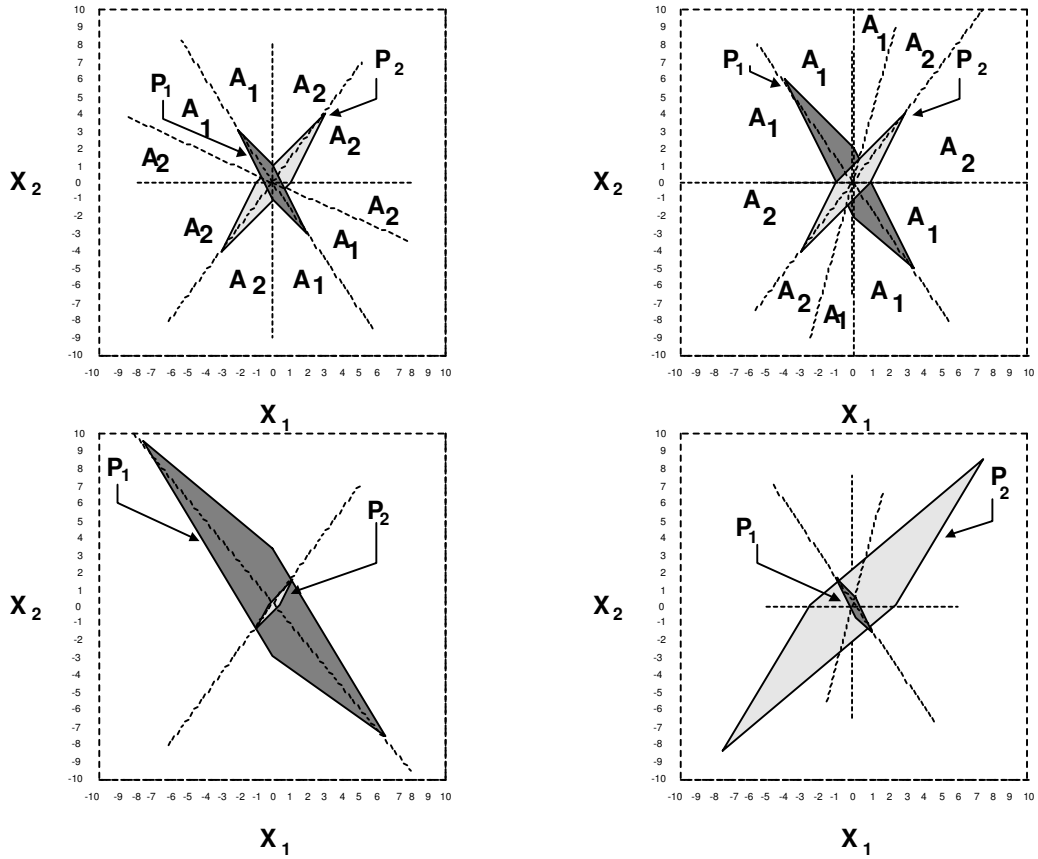
Figure 22: The results returned by the technique in [29] for system (41) for $P_1 \cap P_2 \neq \emptyset$ without scaling (upper left) , with scaling (upper right), $P_2 \subset P_1$ (lower left) and $P_1 \subset P_2$ (lower right).

Suppose that an appropriate ray-partition $\mathcal{R} = \{r_i \ , \ i = 1, 2, \ldots, N\}$ is selected such that a PLF $V_k(\boldsymbol{x})$, $k = 1, 2, \ldots, p$ is found for each linear subsystem $\dot{\boldsymbol{x}} = \mathbf{A}_k \cdot \boldsymbol{x}$ from the induced polytope family $R(\boldsymbol{\lambda})$. Let $\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \ldots, \boldsymbol{\lambda}_p > 0$ s.t. the ray-polytopes $R(\boldsymbol{\lambda}_k)$, $k = 1, 2, \ldots, p$ are invariant polytopes induced by the PLFs $V_k(\boldsymbol{x})$, $k = 1, 2, \ldots, p$. Then application of the same algorithm (in [29]) is simplified and can overcome the shortcomings identified :

Ray-gridding is much more efficient and flexible than the algorithm in [18] for linear systems. Since all PLFs and corresponding ray-polytopes are induced by the same partition, there is no need to specify the conic regions 40 by calculating intersections. They can be specified by the scaling factors. Moreover, since all ray polytopes lie in the interior of the unit circle the scaling factors can easily inform us when normalisation is required. Normalisation (scaling) can then easily be carried out with the scaling factors. An additional benefit is the ability to refine and build up more complex polytopes and corresponding PLFs which can possibly produce larger classes of stabilising sequences. Finally, the possibility of switching to more than one subsystems in a local region can be investigated.

Example 10. We consider again the system used in Example 9 and apply the ray-gridding technique with a number of progressively refined ray-partitions for the specification of stabilising switching sequences.

Before applying the multiple LF technique, an attempt to find a common PLF for the two subsystems has been made, which failed to find a solution. Hence, there exist destabilising switching laws. The results obtained for progressively refined uniform ray-partitions with 16,32,64 and 128 rays are shown in Figure 23. The conic sectors and the number of the subsystem (to which we can safely switch) are also marked.

We observe that as the ray-partition is progressively refined larger sectors are found which converge to the sectors shown in Figure 23. Further refinement is not necessary. Note that the sufficiently dense ray-partition in Figure 23 revealed a small sector (marked with A_{12}) in which the two polytopes P_1 , P_2 coincide. This implies that in this sector $V_1 = V_2$ and thus arbitrary switching to any subsystem guarantees stability. Switching from $\mathbf{A}_1$ to $\mathbf{A}_2$ and conversely will result in a chattering motion (the
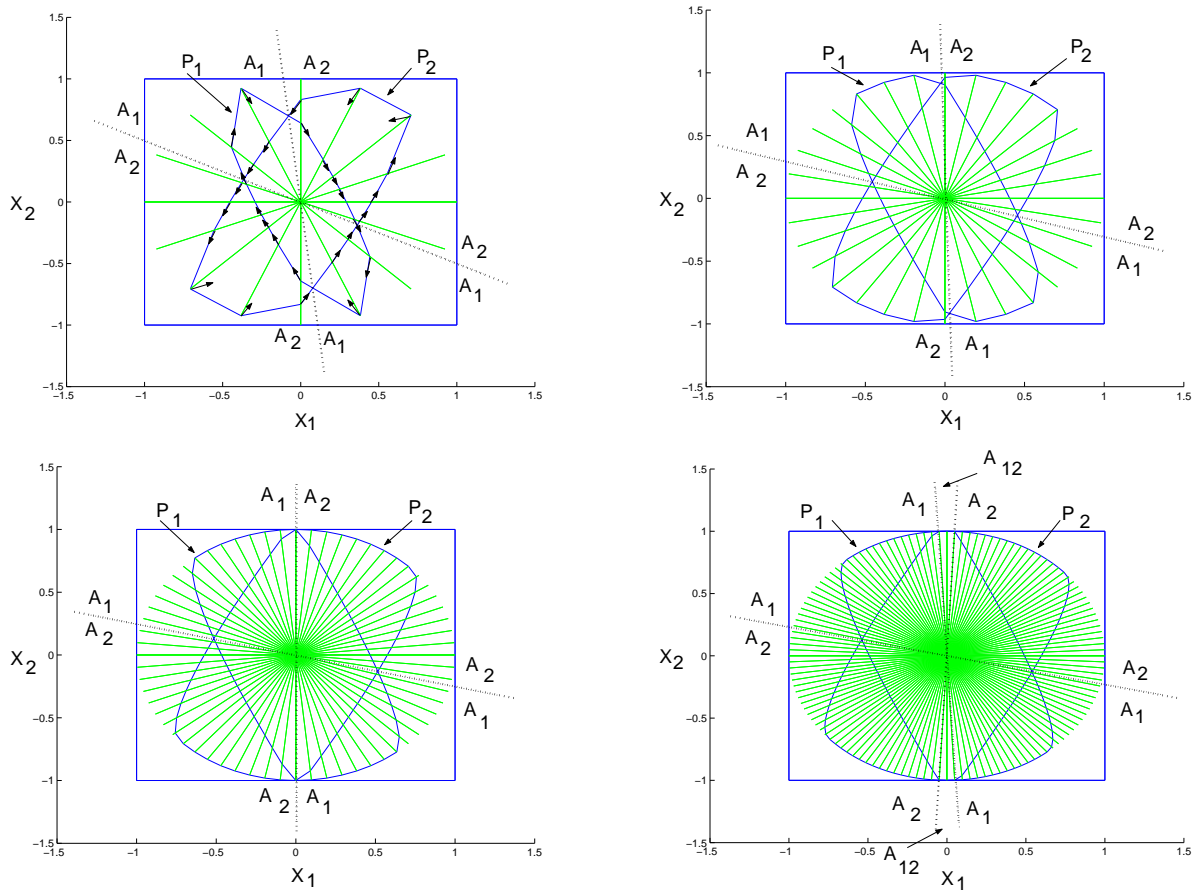
Figure 23: The results returned by the ray-gridding technique for system (41) with 16,32,64 and 128 rays.

two subsystems are of opposite direction, $\mathbf{A}_1$ is clockwise and $\mathbf{A}_2$ is anticlockwise). Infinite frequency switching will produce a sliding mode, which is proved to be stable since $V_1 = V_2$ in A_{12}. If sliding modes are undesirable, the analysis informs us that infinite switching in A_{12} should be avoided.

In Example 10 we observe that the ability of the ray-gridding to refine allows the identification of larger sectors and also additional subsectors in which switching to more than one subsystems is possible.

However, a shortcoming of this multiple Lyapunov Function technique is that it is largely dependent on the initial conditions. Indeed, observe the sectors found in Figures 22 and 23. For different scaled versions of the polytopes P_1, P_2 different results are obtained. It is easy to see in planar examples that a contraction of a polytope, e.g. P_1 will give part of the sector associated with $\mathbf{A}_1$ to the sector associated with $\mathbf{A}_2$. Moreover, there might exist other invariant polytopes (and corresponding Lyapunov functions) which can specify larger sectors for which switching to more than one subsystems is possible. This discussion motivated by a simple planar example suggests that the class of switching possibilities returned by the technique we described are limited. It is then natural to look for robust methods that can specify optimal polytopes or ways for polytope manipulation specifying larger classes of stabilising switching laws.

A further step for improving the multiple Lyapunov function technique introduced in [29] is next described. Since *global* common PLFs do not exist, the main idea is to investigate the possibility for the existence of *local* common PLFs. For the sake of simplicity assume that a linear switched system (1) with two subsystems $\mathbf{A}_1$, $\mathbf{A}_2$ is considered.

Definition 9. Two linear systems $\mathbf{A}_1, \mathbf{A}_2 \in \mathbb{R}^{n \times n}$ admit a local common PLF $V(\boldsymbol{x})$ (usually called a Lyapunov-like function) if for some subset of the state space $\Omega \in \mathbb{R}^n$ we have positive definiteness

$V(\mathbf{0}) = 0$, $V(\boldsymbol{x}) > 0 \; \forall \boldsymbol{x} \neq \mathbf{0} \in \Omega$, and negative definite derivative for both systems

$$\dot{V}(\boldsymbol{x}) = \frac{\partial V(\boldsymbol{x})}{\partial \boldsymbol{x}} \cdot \mathbf{A}_i \boldsymbol{x} \leq 0 \quad , \quad i = 1, 2 \tag{42}$$

Local common PLFs (LCPLFs) are useful for our purposes of finding stabilising switching sequences because they specify regions of the state-space in which arbitrary switching to both subsystems is safe.

Algorithm 6. We propose a modified iterative algorithm for calculating LCPLFs:

Step 1 : Assume that on the basis of a ray-partition $\mathcal{R}$ a PLF $V_i(\boldsymbol{x})$, $i = 1, 2$ with a corresponding invariant set P_i , $i = 1, 2$ is found for each subsystem. We attempt to find LCPLFs for all conic sectors of the simplicial decomposition for which it is possible using the individual V_i's. The starting point are the scaling vectors $\boldsymbol{\lambda}_1$, $\boldsymbol{\lambda}_2$ associated to the invariant sets P_i , $i = 1, 2$.

Step 2 : The algorithm visits progressively all simplices in the simplicial decomposition on a number of iterations and updates the scaling vector $\boldsymbol{\lambda}_1$ with the help of $\boldsymbol{\lambda}_2$ until $\boldsymbol{\lambda}_1$ converges to a constant value. In this step the scaling factors of $\boldsymbol{\lambda}_2$ are kept constant while the scaling factors of $\boldsymbol{\lambda}_1$ are updated according to the following rule :

For the current simplex S_i and its associated rays r_j , $j = 1, \ldots, n$ with scaling factors λ_j , $j = 1, \ldots, n$ search for the existence of new larger scaling factors $\hat{\lambda}_j > \lambda_j$, $j = 1, \ldots, n$ and a positive constant $k > 0$ such that $\hat{V}_1(\boldsymbol{x}) = k \, V_2(\boldsymbol{x})$, $\boldsymbol{x} \in S_i$ and the invariance conditions (30) are satisfied. $\hat{V}_1(\boldsymbol{x})$ is the PLF $V_1(\boldsymbol{x})$ with modified scaling factors $\hat{\lambda}_j$ in the current simplex. If there exist $\hat{\lambda}_j > \lambda_j$, $j = 1, \ldots, n$ these constitute the new updated scaling factors.

Step 3 : A convergence test is performed. The algorithm checks whether any modifications have been made during the last iteration. If yes it returns to Step 2, otherwise convergence has occurred to a new modified scaling vector $\hat{\boldsymbol{\lambda}}_1$.

Step 4 : Steps 2 and 3 are repeated with a changing role for the two scaling vectors and corresponding LFs. Keep the initial scaling vector $\boldsymbol{\lambda}_1$ constant and update the scaling vector $\boldsymbol{\lambda}_2$ to a modified $\hat{\boldsymbol{\lambda}}_2$ using the same rule as in Step 2.

Step 5 : A convergence test is performed. The algorithm checks whether any modifications have been made during the last iteration. If yes it returns to Step 4, otherwise convergence has occurred to a new modified scaling vector $\hat{\boldsymbol{\lambda}}_2$.

Proposition 13. *The convergence of Algorithm 6 to invariant polytopes is guaranteed.*

Proof: Algorithm 6 in Steps 2 and 4 starts from an initial choice which corresponds to an invariant polytope and aims at approximating a target polytope in as many sectors as possible. The algorithm converges since it operates by specifying monotonically increasing sequences of one of the scaling vectors while there is an upper bound determined by the other scaling vector. It converges to new invariant polytopes since it starts from an invariant polytope and at each step an increase of some scaling factors is performed such that the invariance conditions are satisfied. Note that increase of some scaling factors does not affect the invariance of neighboring rays and simplices, thus there is no need to check other neighboring invariance conditions. ∎

Algorithm 6 is applied to the switched system of Examples 9 and 10 to yield an improved result :

Example 11. Algorithm 6 has been applied successfully to the polytopes P_1, P_2 in Figure 10 with 64 rays and the result is shown in Figure 24. Two new invariant polytopes W_1, W_2 with 64 rays are found which specify two conic sectors S_1, S_2 corresponding to LCPLFs.

This result provides more information on the local properties of the two vector fields and specifies a larger class of stabilising switching sequences. We know now that in the sectors S_1, S_2 in Figure 24 arbitrary switching between the two subsystems is possible. Since the two subsystems are of opposite directions, the result of switching in S_1, S_2 will be chattering. Infinitely fast switching will result in sliding modes, which, however, can be proved to be stable using the LCPLFs found. If undesirable, they can be avoided by keeping the frequency of switching low. Outside the sector $S_1 \cup S_2$ switching to one of the subsystems is only allowed. These rules specify a large class of different stabilising switching laws, and stability can be proved using the same multiple Lyapunov function arguments and an appropriate selection of any scaled versions of polytopes P_1, P_2, W_1, W_2, depending on the switching strategy chosen.
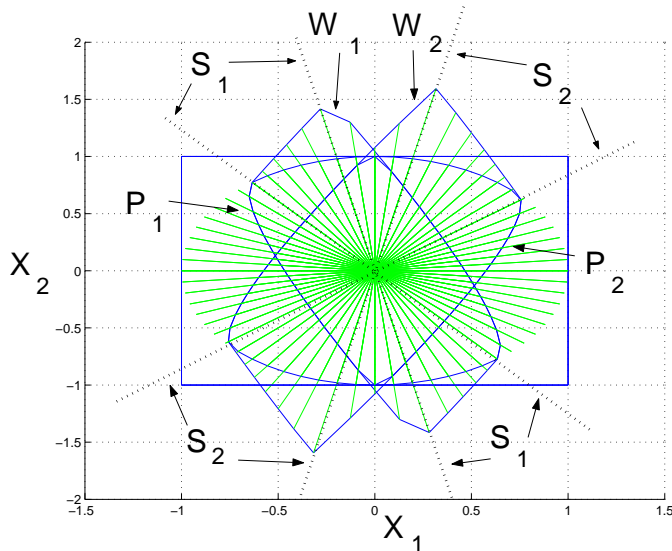
Figure 24: The LCPLF results returned by the ray-gridding technique for system (41) with 64 rays.

Algorithm 6 can be extended to more than two linear subsystems or can be applied to different pairs of subsystems. Further work can look at other multiple Lyapunov function approaches, for which algorithms based on the ray-gridding approach can be developed.

6 Concluding remarks

In this report the ray-gridding approach, a new framework for stability analysis of linear switched systems has been presented. It is based on uniform simplicial partitions of the state-space of adjustable complexity. The technique is supported by simple and efficient numerical iterative algorithms that have been shown to provide solutions to important problems related to stability analysis. The main problem considered was the computation of *common polyhedral* Lyapunov functions that guarantee stability of arbitrary switching between stable subsystems, without any a priori knowledge. The intrinsic difficulties in the numerical calculation of such functions have been noted once again : there is an exponential increase of the computational demands w.r.t. to dimension and complexity of representation. There are also situations in which arbitrarily complex representations and many iterations of solutions may be required for an accurate result (e.g. absolute stability problem). However, improved results in terms of computational complexity have been found allowing a complete treatment of the three-dimensional case. It is also believed that implementation to medium or small-sized problems in higher dimensional systems is also possible.

The same framework has been also extended to the problem of specifying stabilising switching laws for switched systems with stable or unstable subsystems, although further work is required in this area. An algorithm for the calculation of stabilising switching sequences for unstable subsystems has been outlined, which can operate very efficiently in the planar case. The technique proposed in Koutsoukos and Antsaklis (2002) for the calculation of multiple Lyapunov functions for linear switched systems is extended and improved by the ray-gridding approach. Calculation of <u>local</u> common polyhedral LFs can be performed very efficiently and allows the identification of significantly wider classes of stabilising switching sequences.

7 Acknowledgements

---

[2]This work is the sole responsibility of the authors and does not reflect the European Union's opinion

References

[1] BARABANOV, N. Absolute Characteristic Exponent of a class of Linear Nonstationary Systems of Differential Equations. *Automation and Remote Control 29*, 4 (1986), 12–21.

[2] BARABANOV, N. On the Lyapunov exponent of discrete inclusions. I. *Automation and Remote Control*, 2 (1988), 40–46.

[3] BARABANOV, N. On the Lyapunov exponent of discrete inclusions. II. *Automation and Remote Control*, 3 (1988), 24–29.

[4] BARABANOV, N. On the Lyapunov exponent of discrete inclusions. III. *Automation and Remote Control*, 5 (1988), 17–24.

[5] BARABANOV, N. Method for the Computation of the Lyapunov exponent of a Differential Inclusion. *Automation and Remote Control*, 4 (1989), 53–58.

[6] BISHOP, B. E. Lyapunov Function Normalization for Controlled Switching of Hybrid Systems. In *American Control Conference* (1999), pp. 2461–2465.

[7] BLANCHINI, F. Nonquadratic Lyapunov functions for Robust Control. *Automatica 31*, 3 (1995), 451–461.

[8] BLANCHINI, F. Set invariance in control. *Automatica 35* (1999), 1747–1767.

[9] BLANCHINI, F., AND MIANI, S. A new class of universal Lyapunov functions for the control of uncertain linear systems. In *35th IEEE Conference on Decision and Control* (1996), pp. 1027–1032.

[10] BLANCHINI, F., AND MIANI, S. A New Class of Universal Lyapunov Functions for the Control of Uncertain Linear Systems. *IEEE Transactions on Automatic Control 44*, 3 (Mar 1999), 641–647.

[11] BLONDEL, V., AND TSITSIKLIS, J. A survey of computational complexity results in systems and control. *Automatica 36* (2000), 1249–1274.

[12] BOBYLEVA, O. Piecewise-Linear Lyapunov Functions and Localization of Spectra of Stable Matrices. *Automation and Remote Control*, 9 (2001), 25–36.

[13] BOBYLEVA, O. Piecewise-Linear Lyapunov Functions for Linear Stationary Systems. *Automation and Remote Control*, 4 (2002), 26–35.

[14] BRANDTS, J., AND KRIZEK, M. Gradient Superconvergence on uniform simplicial partitions of polytopes. *Pokroky matematiky, fyziky a astronomie*, 47 (2002), 103–113.

[15] BRANICKY, M. S. Stability of switched and hybrid systems. In *33rd IEEE Conference on Decision and Control* (1994), pp. 3498–3503.

[16] BRANICKY, M. S. Multiple Lyapunov Functions and Other Analysis Tools for Switched and Hybrid Systems. *IEEE Transactions on Automatic Control 43*, 4 (1998), 475–482. Special issue on Hybrid Systems.

[17] BRAYTON, R. K., AND TONG, C. H. Stability of dynamical systems: A constructive approach. *IEEE Transactions on Circuits and Systems*, 26 (1979), 224–234.

[18] BRAYTON, R. K., AND TONG, C. H. Constructive stability and asymptotic stability of dynamical systems. *IEEE Transactions on Circuits and Systems*, 27 (1980), 1121–1130.

[19] CHIEN, M. J., AND KUH, E. S. Solving Nonlinear Resistive Networks Using Piecewise-Linear Analysis and Simplicial Subdivision. *IEEE Transactions on Circuits and Systems 24*, 6 (1977), 305–317.

[20] Chua, L. O., and Ying, R. L. Canonical Piecewise-Linear Analysis. *IEEE Transactions on Circuits and Systems 30*, 3 (1983), 125–140.

[21] Dayawansa, W., and Martin, C. A Converse Lyapunov Theorem for a Class of Dynamical Systems which Undergo Switching. *IEEE Transactions on Automatic Control 44*, 4 (1999), 751–760.

[22] Edelsbrunner, H., and Grayson, D. R. Edgewise Subdivision of a Simplex. *Discrete Comput. Geom.*, 24 (2000), 707–719.

[23] Johansson, M. Analysis of piecewise linear systems via convex optimization–a unifying approach. In *Proceedings of the 1999 IFAC World Congress, Beijing, China* (1999).

[24] Johansson, M. *Piecewise Linear Control Systems*. PhD thesis, Department of Automatic Control, Lund University of Technology, Sweden, 1999.

[25] Johansson, M. Analysis of Piecewise Linear Systems via convex optimization – a unifying approach. Technical report, Lund Institute of Technology, 2000.

[26] Julian, P., Dogaru, R., Itoh, M., hanggi, M., and Chua, L. Simplicial rtd-based cellular nonlinear networks. *IEEE Transactions on Circuits and Systems 50*, 4 (2003), 500–509.

[27] Julian, P., Guivant, J., and Desages, A. A parametrization of piecewise linear lyapunov functions via linear programming. *Int. Journal of Control 72*, 7/8 (1999), 702–715.

[28] Kiendl, H., Adamy, J., and Stelzner, P. Vector Norms as Lyapunov Functions for Linear Systems. *IEEE Transactions on Automatic Control 37*, 6 (1992), 839–842.

[29] Koutsoukos, X. D., and Antsaklis, P. J. Design of stabilizing switching control laws for discrete and continuous-time linear systems using piecewise-linear lyapunov functions. *Int. Journal of Control 75*, 12 (2002), 932–945.

[30] Liberzon, D., and Morse, A. Basic problems in stability and design of switched systems. *IEEE Control Systems Magazine* (1999), 59–70.

[31] Liu, D., and Molchanov, A. P. Criteria for robust absolute stability of time-varying nonlinear continuous-time systems. *Automatica* (2002), 627–637.

[32] Margaliot, M., and Langholz, G. Necessary and Sufficient Conditions for Absolute Stability: The Case of Second-Order Systems. *IEEE Transactions on Circuits and Systems 50*, 2 (February 2003), 227–234.

[33] Michel, A. N., Nam, B. H., and Vittal, V. Computer generated Lyapunov functions for interconnected systems: Improved results with application to power systems. *IEEE Transactions on Circuits and Systems 31* (1984), 189–198.

[34] Molchanov, A., and Pyatnitskii, E. Lyapunov functions that specify Necessary and Sufficient Conditions of Absolute Stability of Nonlinear Nonstationary Control Systems. I. *Automation and Remote Control*, 3 (1986), 63–73.

[35] Molchanov, A., and Pyatnitskii, E. Lyapunov functions that specify Necessary and Sufficient Conditions of Absolute Stability of Nonlinear Nonstationary Control Systems. II. *Automation and Remote Control*, 5 (May 1986), 5–15.

[36] Molchanov, A., and Pyatnitskii, E. Lyapunov functions that specify Necessary and Sufficient Conditions of Absolute Stability of Nonlinear Nonstationary Control Systems. III. *Automation and Remote Control*, 5 (May 1986), 38–49.

[37] Molchanov, A., and Pyatnitskiy, Y. Criteria of asymptotic stability of differential and difference inclusions encountered in control theory. *Systems and Control Letters 13* (1989), 59–64.

[38] MOORE, D. W. Simplicial Mesh Generation with Applications. Technical report, Sept. Comput. Sci., Cornell Univ., Ithaca, New York, 1992.

[39] OHTA, Y., IMANISHI, H., AND HANEDA, H. Computer Generated Lyapunov Functions for a Class of Nonlinear Systems. *IEEE Transactions on Circuits and Systems 40*, 5 (May 1993), 343–353.

[40] OHTA, Y., NONOMURA, T., IMANISHI, H., AND HANEDA, H. Computer generation of Lyapunov functions and estimation of a stability region. *Electronics & Communications in Japan, Part III: Fundamental Electronic Science (English translation of Denshi Tsushin Gakkai Ronbunshi) 77*, 4 (1994), 44–54.

[41] POLAŃSKI, A. On Infinity Norms as Lyapunov Functions for Linear Systems. *IEEE Transactions on Automatic Control 40*, 7 (1995), 1270–1274.

[42] POLAŃSKI, A. Lyapunov Function Construction by Linear Programming. *IEEE Transactions on Automatic Control 42*, 7 (1997), 1013–1016.

[43] POLANSKI, A. Destabilisng Strategies for Uncertain Linear Systems. *IEEE Transactions on Automatic Control 45*, 12 (2000), 2378–2382.

[44] POLANSKI, A. On absolute stability analysis by polyhedral Lyapunov functions. *Automatica 36* (2000), 573–578.

[45] PYATNITSKIY, Y., AND RAPOPORT, B. Criteria of Asymptotic Stability of Differential Inclusions and Periodic Motions of Time-Varying Nonlinear Control Systems. *IEEE Transactions on Circuits and Systems 43*, 3 (1996), 219–229.

[46] ROMANCHUK, B. G. Computing regions of attraction with polytopes: Planar case. *Automatica 32*, 12 (1996), 1727–1732.

[47] WULFF, K., SHORTEN, R., AND CURRAN, P. On the 45-region and the uniform asymptotic stability of classes of second order parameter-varying and switched systems. *Int. Journal of Control 75*, 11 (2002), 812–823.

[48] XU, X., AND ANTSAKLIS, P. Stabilization of second-order LTI switched system. *Int. Journal of Control* (2000), 1261–1279.

[49] YFOULIS, C. *Stabilisation of Nonlinear Systems:the piecewise linear approach.* Research Studies Press Limited, Hertfordshire, England., 2001.

[50] YFOULIS, C., MUIR, A., AND P.E.WELLSTEAD. A new approach for estimating controllable and recoverable regions for systems with state and control constraints. *Int. Journal of Robust and Nonlinear Control 12* (2002), 561–589.

[51] ZIEGLER, G. *Lectures on Polytopes.* Springer-Verlag, 1995.

Appendix

Invariance conditions for planar systems : proof of Proposition 1

Let r_{i-1}, r_i, r_{i+1} define three successive rays arranged in a counterclockwise order around the origin (Figure 25), suppose that e_{i-1}, e_i, e_{i+1} are the corresponding extreme ray points (a *constrained* ray partition is assumed), and let $x_{i-1} = \lambda_{i-1} \cdot e_{i-1}$, $x_i = \lambda_i \cdot e_i$, $x_{i+1} = \lambda_{i+1} \cdot e_{i+1}$ be points on them. The invariance conditions for the point x_i which lies in the intersection of the two sectors $S_1 = cconv\{r_{i-1}, r_i\}$ (formed by the rays r_{i-1}, r_i) and $S_2 = cconv\{r_i, r_{i+1}\}$ (formed by the rays r_i, r_{i+1}) take the form (see (8))

$$f_1^T \dot{x} \leq 0 \quad , \quad f_2^T \dot{x} \leq 0 \tag{43}$$

where f_1, f_2 the normals of the lines $\overrightarrow{x_i\, x_{i-1}}$ and $\overrightarrow{x_i\, x_{i+1}}$, respectively. For S_1 an equivalent condition is that the velocity vector of the linear dynamics $\dot{x}_i = \mathbf{A}\, x_i$ (for points on ray r_i) points inwards across the line $\overrightarrow{x_i\, x_{i-1}}$, or that $\dot{x}_i$ lies on the same side of the line which contains the origin; this can be translated into the algebraic statement that the area spanned by the vectors $x_{i-1} - x_i$ and $\dot{x}_i$, namely $\det[x_{i-1} - x_i\,,\, \dot{x}_i]$ has the same sign as $\det[x_{i-1} - x_i\,,\, -x_i]$. Since $\det[x_{i-1} - x_i\,,\, -x_i] = \det[x_i\,,\, x_{i-1}]$ and, with the vectors in the order stated this is negative, the condition becomes

$$\det[x_{i-1} - x_i\,,\, \mathbf{A}\, x_i] = \det[x_{i-1}\,,\, \mathbf{A}\, x_i] + + \det[-x_i\,,\, \mathbf{A}\, x_i] \leq 0$$

and the following relation is obtained

$$d_i\, \lambda_i^2 - d_{i-1}\, \lambda_{i-1}\, \lambda_i \leq 0 \tag{44}$$

For the sector S_2 a similar relation is obtained

$$-d_i\, \lambda_i^2 + d_{i+1}\, \lambda_{i+1}\, \lambda_i \leq 0 \tag{45}$$

where

$$d_i = \det[\mathbf{A}e_i, e_i]\,; \; d_{i-1} = \det[\mathbf{A}e_i, e_{i-1}]\,; \; d_{i+1} = \det[\mathbf{A}e_i, e_{i+1}]\,;$$

From (44),(45) we obtain linear relations

$$d_i\, \lambda_i - d_{i-1}\, \lambda_{i-1} \leq 0 \quad , \quad -d_i\, \lambda_i + d_{i+1}\, \lambda_{i+1} \leq 0 \tag{46}$$

or since $d_i > 0$ if $\dot{e}_i = \mathbf{A} \cdot e_i \in S_1$ (the sequence of the two points $\mathbf{A} \cdot e_i\,,\, e_i$ is counterclockwise) and $d_i < 0$ if $\mathbf{A} \cdot e_i \in S_2$ (clockwise sequence)

$$\lambda_i \leq \Delta_i^{i-1}\, \lambda_{i-1} \quad , \quad \lambda_i \leq \Delta_i^{i+1}\, \lambda_{i+1} \tag{47}$$

with $\Delta_i^{i-1} = \frac{d_{i-1}}{d_i}$, $\Delta_i^{i+1} = \frac{d_{i+1}}{d_i}$.

It is also a matter of simple geometric intuition to note that when the velocity vector coincides with the (inwardly pointing) ray vector, then both conditions are trivially satisfied and they do not need to be checked. In any other case, the velocity vector will belong to one of the two sectors, and then only the corresponding condition from (47) needs to be imposed. This proves Proposition 1, where $q(k) = k - 1$ (if $\dot{e}_i \in S_1$) or $q(k) = k + 1$ (if $\dot{e}_i \in S_2$) $\qquad\square$

Proof of proposition 2 and corollaries 3 and 4

For a planar linear system with complex eigenvalues the flow has constant direction in the whole phase plane (either clockwise or counterclockwise). All rays are non-trivial and the dependency graph is a single cycle, a chain connecting all rays in their ordered sequence. In Figure 27 a ray-partition with non-trivial rays in $\mathbb{R}^2$ for a clockwise system is shown, whereas Figure 28 depicts the corresponding dependency graph. It is 2-regular graph (each node has degree 2, i.e. there are two edges incident to it, one coming and one leaving) with no isolated vertices and no loops. The arrow directions and weights encode the conditions (13), which take a simpler form

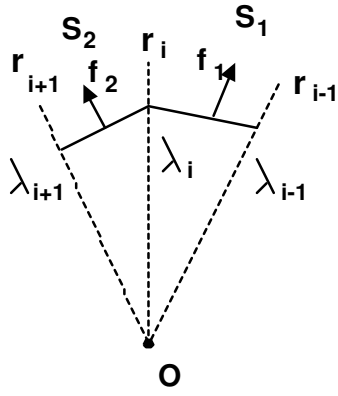$$\lambda_i \leq \Delta_i^{i-1}\lambda_{i-1}\,, \; i = 1, \ldots, 8. \tag{48}$$
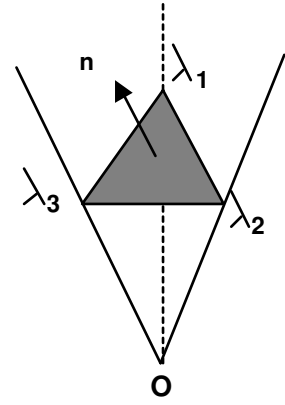
Figure 25: Three neighboring rays in $\mathbb{R}^2$.

Figure 26: A simplex in $\mathbb{R}^3$.

It is obvious that successive application of (48) for all rays in the cycle yields a solution if and only if $\prod_i \Delta_i^{i-1} \geq 1$. Then a scaling vector $\lambda > \mathbf{0}$ satisfying (48) can be easily found from the weights in the graph using Algorithm 1. Corollary 3 is proved.

Planar systems with real eigenvalues possess trivial rays, i.e. rays along which the derivative direction is radial. These are the eigenvector lines. For trivial rays no conditions of the form (13) need to be imposed, hence the graph nodes corresponding to them are either of degree 0 (isolated) or of degree two with two edges leaving from them. Both cases are shown in Figure 29 and 30 with the corresponding graphs. It is easy to see that if there exist two separate eigenvector lines and 4 rays on them, no conditions are imposed and an invariant polytope is found for arbitrary scaling factors (Figure 29). This is the case where a minimal number of generators $m = 2$ is sufficient for a PLF, specified by the transformation matrix to block diagonal form, which of course uses the eigenvector directions.

In the presence of trivial rays, no restrictions to them apply, thus after a selection of some scaling factors for them the remaining rays are assigned values according to their weights in the graph and a solution is always feasible (Figure 30). If there exist cycles in the graph then the only condition imposed is obviously that the product of their weights must satisfy $\Delta_{k_1}^{k_2} \Delta_{k_2}^{k_1} \geq 1$, where k_1, k_2 the two nodes that form the cycle (nodes 1,2 and 5,6 in Figure 31). Corollary 4 is proved.
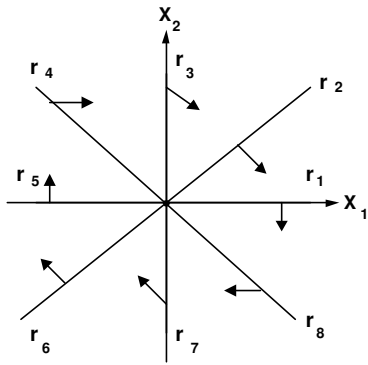

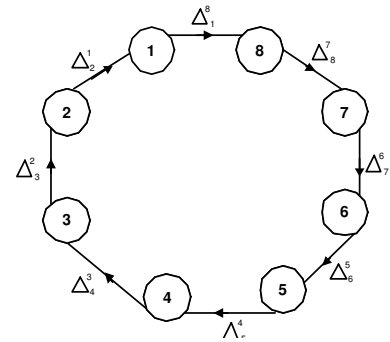Figure 27: A ray-partition with 8 rays in $\mathbb{R}^2$ for a clockwise system.

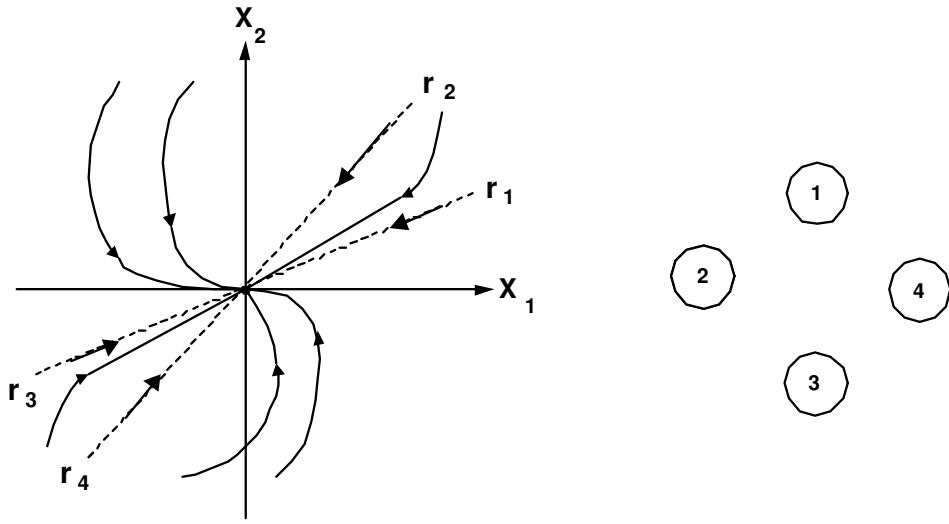Figure 28: The dependency graph of a clockwise system in $\mathbb{R}^2$.

Figure 29: A ray-partition with 4 trivial rays in $\mathbb{R}^2$ for a system with real eigenvalues.
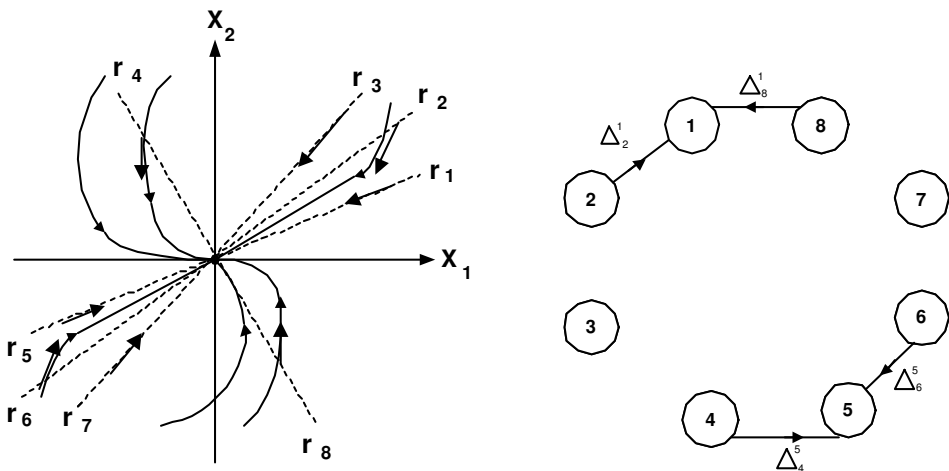

Figure 30: A ray-partition with 8 rays in $\mathbb{R}^2$ for a system with real eigenvalues (4 trivial rays are included).
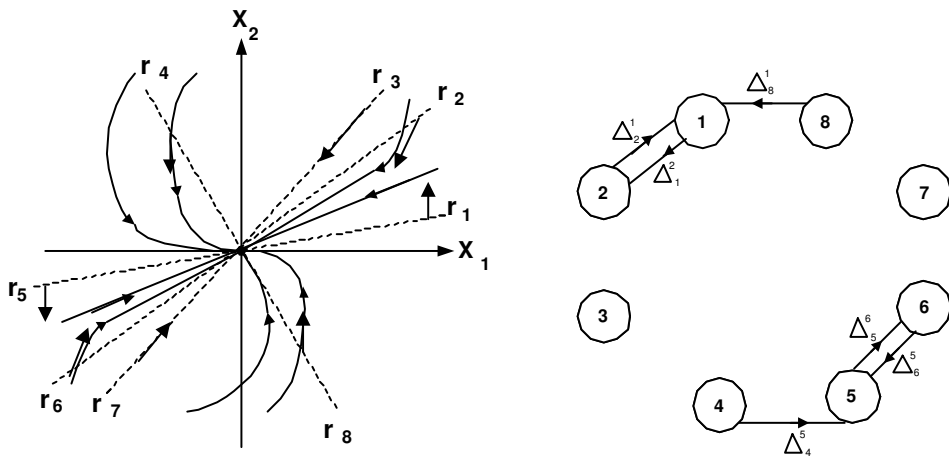

Figure 31: A ray-partition with 8 rays in $\mathbb{R}^2$ for a system with real eigenvalues (no trivial rays are included and 2 cycles are formed).

Invariance conditions for a simplex in $\mathbb{R}^n$: proof of proposition 7

We consider a simplex S in $\mathbb{R}^n$ formed by the origin and n linearly independent points $\boldsymbol{x}_i = \lambda_i \cdot \boldsymbol{e}_i$, $i = 1, ..., n$ on distinct rays r_i with corresponding ray vectors $\boldsymbol{e}_i$. Such a simplex in $\mathbb{R}^3$ is shown in Figure 26. Our aim is to specify the invariance condition specified for one of the rays, e.g. r_1 under the linear switched dynamics (1) in terms of the remaining ones. Assuming that the simplex S considered is the active simplex for the linear system $\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x}$, i.e the velocity vector $\dot{\boldsymbol{x}}$ for points on ray r_1 points to S, the invariance condition takes the form

$$\boldsymbol{n}^T \cdot \dot{\boldsymbol{x}}_1 \leq 0 \tag{49}$$

where $\boldsymbol{n}$ the vector normal to the $(n-1)$-dimensional hyperplane defined by $\boldsymbol{n}^T \cdot \boldsymbol{x} = 1$ and containing the n vertices $\boldsymbol{x}_i$. Let $\mathbf{X} = [\boldsymbol{x}_1 \boldsymbol{x}_2 \dots \boldsymbol{x}_n] \in \mathbb{R}^{n \times n}$ and $\mathbf{E} = [\boldsymbol{e}_1 \boldsymbol{e}_2 \dots \boldsymbol{e}_n] \in \mathbb{R}^{n \times n}$. Then the normal $\boldsymbol{n}$ can be found as $\boldsymbol{n} = (\mathbf{X}^T)^{-1} \cdot \mathbf{1}$ and can be directly related to $\mathbf{E}^{-1}$ and the scaling factors λ_i , $i = 1, \dots, n$. If the cofactors in $\det(\mathbf{E})$ are E_{ij} then simple calculations yield

$$n_i = \sum_{j=1}^n E_{ij} \lambda_{k_1} \lambda_{k_2} \dots \lambda_{k_{n-1}} , \; k_1 \neq k_2 \dots \neq k_{n-1} \neq j , \; 1 \leq k_r \leq n , \; 1 \leq r \leq (n-1) \tag{50}$$

and with further manipulations we arrive at

$$\boldsymbol{n}^T \cdot \dot{\boldsymbol{x}}_1 = \boldsymbol{n}^T \cdot \mathbf{A} \cdot \boldsymbol{x}_1 = \sum_{l=1}^n e_{1l} \cdot \sum_{i=1}^n a_{il} \cdot \sum_{j=1}^n E_{ij} \lambda_{k_1} \lambda_{k_2} \dots \lambda_{k_{n-1}} \leq 0 \tag{51}$$

or in simpler form

$$\sum_{k=1}^n \delta_k \cdot \lambda_{k_1} \lambda_{k_2} \dots \lambda_{k_{n-1}} \leq 0 \tag{52}$$

Solving for the scaling factor λ_1 yields

$$\lambda_1 \leq -\frac{\delta_1 \cdot \lambda_2 \dots \lambda_n}{\sum_{k=2}^n \delta_k \cdot \lambda_{k_1} \lambda_{k_2} \dots \lambda_{k_{n-2}}} , \; k_1 \neq k_2 \dots \neq k_{n-2} \neq 1 \tag{53}$$

E.g. for $n = 2$ (53) gives $\lambda_1 \leq -\frac{\delta_1}{\delta_2} \cdot \lambda_2$ while for $n = 3$ it yields $\lambda_1 \leq -\frac{\delta_1 \cdot \lambda_2 \lambda_3}{\delta_2 \cdot \lambda_2 + \delta_3 \cdot \lambda_3}$ and for $n = 4$ $\lambda_1 \leq -\frac{\delta_1 \cdot \lambda_2 \lambda_3 \lambda_4}{\delta_2 \cdot \lambda_2 \lambda_3 + \delta_3 \cdot \lambda_2 \lambda_4 + \delta_4 \cdot \lambda_3 \lambda_4}$. In the derivation of (53) we have implicitly assumed that

$$\sum_{k=2}^n \delta_k \cdot \lambda_{k_1} \lambda_{k_2} \dots \lambda_{k_{n-2}} > 0 \tag{54}$$

This a natural hypothesis, since for a sensible setting of ray directions satisfaction of the stability condition (49) for $\lambda_1 = \lambda_1^*$ implies satisfaction for all $\lambda_1 \leq \lambda_1^*$ (since for smaller values a larger geometrical volume for inclusion of the velocity vector is specified). If inequality (54) is not satisfied, this implies that a a new refined ray-partition with an increased number of rays needs to be considered.

For the linear switched dynamics (1) a number of linear subsystems $\dot{\boldsymbol{x}} = \mathbf{A}_i \boldsymbol{x}$ are involved, hence relation (52) becomes

$$\sum_{k=1}^n \delta_k^{(i)} \cdot \lambda_{k_1^{(i)}} \lambda_{k_2^{(i)}} \dots \lambda_{k_{n-1}^{(i)}} \leq 0 \tag{55}$$

where $\delta_k^{(i)}$, $i = 1, \dots, p$, $k = 1, \dots, n$ the coefficients obtained for all individual subsystems. It is obvious that the scaling factor upper bound is determined by the scaling factors of incident rays in the same simplex in terms of a linear inequality (55) and this offers simplicity and efficiency in the implementation of the technique. The use of linear programs or convex hull computations is not necessary.