Complexity Analysis of Computing Piecewise Quadratic Lyapunov Functions for Switched Linear Systems

Can Chen* Bingwen Yang † Ming Li ‡

Abstract

In this paper, we investigate the complexity of computing piecewise quadratic Lyapunov functions for switched linear systems. We exploit the Self-Dual-Minimisation (SeDuMi) package to solve the semidefinite programming (SDP) problem associated with piecewise Lyapunov equations. We establish results on the time and space complexity of computing piecewise quadratic Lyapunov functions. In particular, we find that the time complexity is polynomial in system dimension and is linear in number of cells or number of transition matrices. Moreover, we design a class of switched linear systems which can be used to verify the complexity in arbitrary system dimension and number of cells. Finally, we test the algorithm on a real world nonlinear dynamical system via piecewise linearization.

Keywords: switched linear systems, Lyapunov stability, complexity analysis, semidefinite programming

1 Introduction

A switched linear system is a hybrid system which consists of a collection of linear systems and a discrete switching rule [8, 16]. Evidence has shown that a well-tuned switched linear system can accomplish better performance compared to a standard linear system [5, 13, 16]. Switched linear systems with switched controller have wide applications in mechanical systems, process control, automotive industry, power systems, traffic control and so on [12]. Moreover, some nonlinear dynamical systems can be analyzed only through switching control schemes rather than the standard state feedback law [3, 12]. Many system theory properties of switched linear systems such as stability, controllability and observability have been heavily studied in order to achieve guaranteed performance [1, 2, 4, 9, 10, 12].

The stability analysis of switched linear systems has been explored since 1990s [2, 9, 12]. The stability of a switched linear system depends on both the state transition matrices and the discrete switching rule. It is interesting that even if all the state transition matrices are stable (i.e., all the eigenvalues are negative), the switched linear system can still be unstable, see examples in [2]. On the other hand, if one can find a common quadratic Lyapunov function for all the state transition matrices, the stability of the switched linear system can be achieved, often referred to as uniform stability. However, the common quadratic Lyapunov functions fail to exist for most stable switched linear systems. In order to resolve the conundrum, Johansson and Rantzer [9] proposed a computational approach to stability analysis of switched affine systems using piecewise quadratic Lyapunov functions.

The main idea of piecewise quadratic Lyapunov functions is using the geometry properties of the cells, which are polyhedra, to establish piecewise quadratic Lyapunov functions. The computations of piecewise quadratic Lyapunov functions can be readily summarized into semidefinite programming (SDP) problems. The contributions of this paper are as follows:

- We establish results on the time and space complexity of computing piecewise quadratic Lyapunov functions for switched linear systems using the SDP solver SeDuMi. We find that the time complexity is polynomial in system dimension and is linear in number of cells.

*Department of Mathematics and Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor MI (canc@umich.edu).

†Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor MI (ybingwen@umich.edu).

‡Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor MI (llswsfy@umich.edu).

- We design a class of stable switched linear systems that do not possess common Lyapunov functions with arbitrary system dimension and number of cells. We verify our complexity results using the designed switched linear systems.
- We apply the piecewise quadratic Lyapunov function computation framework to nonlinear dynamics - prey predatory systems. We demonstrate that the stability of such systems can be approximated by linearizing the systems to switched linear systems.

The paper is organized into six sections. We start with the basics of piecewise quadratic Lyapunov functions and SDP in Section 2.1 and 2.2, repsectively. Some backgrounds of the SDP solver SeDuMi are summarized in Section 2.3. In Section 3.1, we establish results on the time and space complexity of computing piecewise quadratic Lyapunov functions for switched linear systems. In Section 3.2, we detail the procedure of designing a class of stable switched linear systems that do not possess common Lyapunov functions with arbitrary system dimension and number of cells, and we verify the time complexity using the designed switched linear systems in Section 3.3. Finally, we discuss the other possibilities of constructing switched linear systems in Section 5 and conclude with future directions in Section 6.

2 Preliminaries

2.1 Piecewise quadratic Lyapunov functions

We take most of the concepts and notations for piecewise quadratic Lyapunov functions from the comprehensive work [9]. First, we discuss the notion of piecewise linear systems.

Definition 1. *The piecewise linear systems are of the form:*

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}_i \boldsymbol{x}(t) \text{ for } \boldsymbol{x}(t) \in X_i, \tag{1}$$

where, $\boldsymbol{A}_i \in \mathbb{R}^{n\times n}$ are state transition matrices, and $\{X_i\}_{i\in I} \subseteq \mathbb{R}^n$ are polyhedral cells partitioning the state space (I is the index set of the cells).

In order to establish Lyapunov stability of the system (1), one may consider common Lyapunov functions of the form $V(\mathbf{x}) = \mathbf{x}^\top \mathbf{P}\mathbf{x}$, which can be solved from convex optimization.

Theorem 1. *If there exists a matrix $\boldsymbol{P} = \boldsymbol{P}^\top \succ 0$ such that $\boldsymbol{A}_i^\top \boldsymbol{P} + \boldsymbol{P}\boldsymbol{A}_i \prec 0$ for $i \in I$, then every trajectory of (1) tends to zero exponentially.*

However, the common quadratic Lyapunov functions fail to exist for most stable switched linear systems. It is known that if there exists positive definite matrices $\mathbf{R}_i$ such that

$$\sum_{i\in I} \mathbf{A}_i^\top \mathbf{R}_i + \mathbf{R}_i \mathbf{A}_i \succ 0,$$

then the the Lyapunov inequalities in Theorem 1 do not admit a solution $\mathbf{P}$ such that $\mathbf{P} = \mathbf{P}^\top \succ 0$. The essence of piecewise quadratic Lyapunov functions is exploiting the geometry properties of the cells plus $\mathcal{S}$-procedure to build the Lyapunov inequalities. It turns out that this technique outperforms the common Lyapunov function approach for switched linear systems [9]. In the following, we introduce two classes of matrices $\mathbf{E}_i$ and $\mathbf{F}_i$ based on the geometry properties of the cells:

$$\mathbf{E}_i \mathbf{x} \geq 0 \text{ for } \mathbf{x} \in X_i, \tag{2}$$

$$\mathbf{F}_i \mathbf{x} = \mathbf{F}_j \mathbf{x} \text{ for } \mathbf{x} \in X_i \cap X_j, \tag{3}$$

where, $\mathbf{z} \geq 0$ represents each entry of $\mathbf{z}$ is nonnegative. The two class of matrices play significant roles in establishing Lyapunov stability using piecewise quadratic Lyapunov function as shown below.

Theorem 2. *Consider symmetric matrices $\boldsymbol{T}$, $\boldsymbol{U}_i$ and $\boldsymbol{W}_i$ such that $\boldsymbol{U}_i$ and $\boldsymbol{W}_i$ have nonnegative entries. If the piecewise quadratic Lyapunov function is given by*

$$\boldsymbol{P}_i = \boldsymbol{F}_i^\top \boldsymbol{T}\boldsymbol{F}_i, \text{ for } i \in I \tag{4}$$

which satisfies

$$\begin{cases} \boldsymbol{A}_i^\top \boldsymbol{P}_i + \boldsymbol{P}_i \boldsymbol{A}_i + \boldsymbol{E}_i^\top \boldsymbol{U}_i \boldsymbol{E}_i \prec 0 \\ \boldsymbol{P}_i - \boldsymbol{E}_i^\top \boldsymbol{W}_i \boldsymbol{E}_i \succ 0 \end{cases}, \text{ for } i \in I \tag{5}$$

then every trajectory of the switched linear system (1) for $t \geq 0$ tends to zero exponentially.

2.2 Semidefinite programming

Semidefinite programming (SDP) is a subfield of convex optimization, which minimizes a linear function subject to the constraint that an affine combination of symmetric matrices is positive semidefinite [17]. The SDP problem takes the form of:

$$\text{minimize } \mathbf{c}^\top \mathbf{x} \tag{6}$$

$$\text{subject to } \mathbf{F}_0 + \sum_{i=1}^{m} \mathbf{F}_i \mathbf{x}_i \succ 0, \tag{7}$$

where, $\mathbf{c} \in \mathbb{R}^n$ and $\mathbf{F}_0, \mathbf{F}_1, \ldots, \mathbf{F}_m \in \mathbb{R}^{n\times n}$ are symmetric matrices. We call the inequality a linear matrix inequality (LMI). It is straightforward to see that the linear Lyapunov stability condition is a LMI, i.e., $\mathbf{A}^\top \mathbf{P} + \mathbf{P}\mathbf{A} \prec 0$ is linear in $\mathbf{P}$. Therefore, the piecewise quadratic Lyapunov function for switched linear systems can be solved via SDP. Moreover, the SDP problem has been well studied and can be solved very efficiently via interior-point methods [6].

2.3 The SeDuMi solver

SeDuMi, standing for Self-Dual-Minimization, is an optimization toolbox in MATLAB, which can solve optimization problems with linear, quadratic and semidefiniteness constraints [14, 15]. It exploits the efficient interior-point algorithms with a polynomial-time complexity. The number of flops needed to compute an ϵ-accurate solutions is bounded by $\mathcal{O}(n^3 r \log \frac{1}{\epsilon})$ where n is the dimension of the variables, and r is the total rows of the LMI [7]. In [14], it is also mentioned that the complexity is about $\mathcal{O}(n^2 r^{2.5} + r^{3.5})$ if n is large. We will use the first complexity in the complexity analysis in this paper since our simulations indicate that the complexity is closer to the former one. Other major advantages of SeDuMi also include:

- It exploits the sparse format in solving the SDP problem, which can save a great amount of memory;
- It can tackle medium-size SDP problems very efficiently;
- It allows complex data inputs for complex-valued SDP problems.

We will use SeDuMi as the main solver in all the numerical examples presented in the paper.

3 Results

3.1 Complexity analysis

The algorithm of computing piecewise quadratic Lyapunov functions for switched linear systems based on Theorem 2 is given in Algorithm 1 below. In step 2 and 3, the variable matrices and the matrix constraints vector can be constructed using the Yalmip syntax `sdpvar`. In step 6, the colon : denotes the colon operator as used in MATLAB. We estimate the time and space complexitity of Algorithm 1 in the following propositions.

Proposition 1. *The time complexity of Algorithm 1 is about $\mathcal{O}(n^4 s \log \frac{1}{\epsilon} + n^3 ms \log \frac{1}{\epsilon} + np^2 s + nm^2 s)$.*

Proof. We first consider the time complexity of constructing the matrix constraints vector L. In each loop, the number operations of matrix multiplications and additions can be estimated as $5n^2 + 2np^2 + 4nm^2$. Therefore, the total complexity is about $\mathcal{O}(n^2 s + np^2 s + nm^2 s)$. Second, from Section 2.3, we know that the

Algorithm 1: Computing piecewise quadratic Lyapunov functions

1: Given a switched linear system with transition matrices $\mathbf{A}_i \in \mathbb{R}^{n\times n}$, matrices $\mathbf{E}_i \in \mathbb{R}^{m\times n}$ and $\mathbf{F}_i \in \mathbb{R}^{p\times n}$ described in Section 2.1 for $i = 1, 2, \dots, s$, and a prescribed accuracy ϵ
2: Define variable matrices $\mathbf{T} \in \mathbb{R}^{p\times p}$, $\mathbf{U}_i \in \mathbb{R}^{m\times m}$ and $\mathbf{W}_i \in \mathbb{R}^{m\times m}$ for $i = 1, 2, \dots, s$
3: Let L be the matrix constraints vector of length $4s$
4: **for** $i = 1, 2, \dots, s$ **do**
5: Set $\mathbf{P}_i = \mathbf{F}_i^\top \mathbf{T}\mathbf{F}_i \in \mathbb{R}^{n\times n}$
6: Set $L(4i-3:4i) = \left[\mathbf{A}_i^\top \mathbf{P}_i + \mathbf{P}_i\mathbf{A}_i + \mathbf{E}_i^\top \mathbf{U}_i\mathbf{E}_i \prec 0 \quad \mathbf{P}_i - \mathbf{E}_i^\top \mathbf{W}_i\mathbf{E}_i \succ 0 \quad vec(\mathbf{U}_i) \succ 0 \quad vec(\mathbf{W}_i) \succ 0\right]$ where vec is the vectorization operator
7: **end for**
8: Apply the SeDuMi solver to solve the SDP problem with matrix constraints vector L
9: **return** The piecewise quadratic Lyapunov functions $\mathbf{P}_i$ if the optimization is feasible.

time complexity of solving the SDP problem using the SeDuMi solver is at most $\mathcal{O}(n^3 r \log \frac{1}{\epsilon})$ where r is total rows of the matrix inequality constraints. In our case, r is equal to $(2n+2m)s$. Thus, it follows immediately that the total time complexity of Algorithm 1 is given by $\mathcal{O}(n^4 s \log \frac{1}{\epsilon} + n^3 ms \log \frac{1}{\epsilon} + np^2 s + nm^2 s)$. □

Clearly, the estimated time complexity of computing piecewise quadratic Lyapunov functions for switched linear systems is polynomial in the system dimension and linear in number of cells. Next, we consider the space complexity of Algorithm 1 excluding step 8. Although we know that the SeDuMi solver is very efficient in memory, see Section 2.3, we fail to find any reference about the exact memory cost of the SeDuMi solver.

Proposition 2. *The space complexity of Algorithm 1 excluding step 8 is about* $\mathcal{O}(n^2 s + m^2 s + p^2 + nms + nps)$.

Proof. The result follows immediately from combining the sizes of all the matrices and the matrix constraints vector in Algorithm 1. □

3.2 Switched linear system design

In this subsection, we design a class of switched linear systems for arbitrary dimension that can be divided into arbitrary number of cells with corresponding $\mathbf{E}_i$. For each cell, the number of rows of the matrix $\mathbf{F}_i$ can be varied. Moreover, our design of the state transition matrices $\mathbf{A}_i$ ensures the nonexistence of common quadratic Lyapunov function.

3.2.1 Design of $\mathbf{E}_i$ and $\mathbf{F}_i$

In order to design switched linear systems with arbitrary number of cells that can be expressed by the matrices $\mathbf{E}_i$ and $\mathbf{F}_i$, we can evenly and symmetrically divide the space as the example shown in [9]. In this way, each cell will have two edges, and we can assume that the matrix $\mathbf{E}_i$ is of the form:

$$\mathbf{E}_i = \begin{bmatrix} \mathbf{e}_{i1}^\top \\ \mathbf{e}_{i2}^\top \end{bmatrix},$$

where, $\mathbf{e}_{ij}$ is a column vector perpendicular to the edge j of the cell i and pointing inside the cell for $j = 1, 2$. Therefore, it is easy to verify that the matrix $\mathbf{E}_i$ satisfies the condition

$$\mathbf{E}_i\mathbf{x} \geq 0 \text{ for } \mathbf{x} \in X_i.$$

A 2-dimensional space with 5 cells is shown in Figure 1A.

Second, we want to design a matrix $\mathbf{F}_i$ with arbitrary number of rows satisfying the condition

$$\mathbf{F}_i\mathbf{x} = \mathbf{F}_j\mathbf{x} \text{ for } \mathbf{x} \in X_i \cap X_j.$$

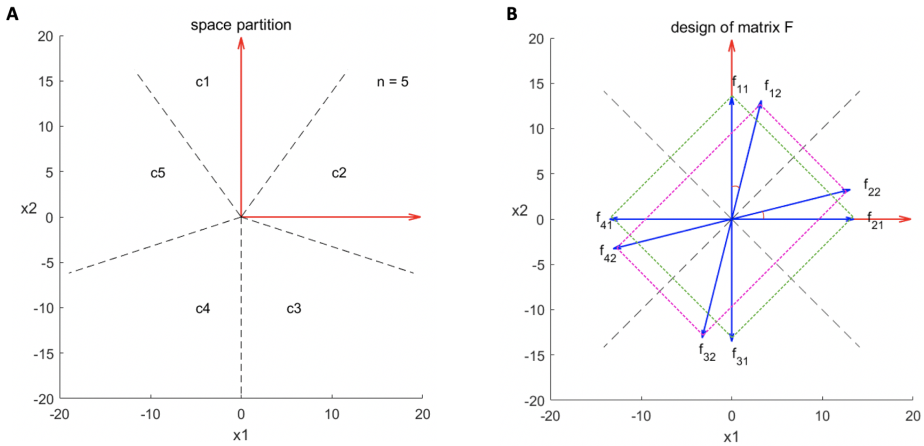

Figure 1: **Design of the matrices $\mathbf{E}_i$ and $\mathbf{F}_i$. A:** Partition of the space state with 5 cells. **B:** Constructions of $\mathbf{f}_{in}$ in designing $\mathbf{F}_i$.

Let $\mathbf{F}_i = \begin{bmatrix}\mathbf{f}_{i1} & \mathbf{f}_{i2} & \ldots & \mathbf{f}_{in}\end{bmatrix}^\top$ with

$$\mathbf{f}_{in}^\top \mathbf{x} = \mathbf{f}_{jn}^\top \mathbf{x} \text{ for } \mathbf{x} \in X_i \cap X_j.$$

It indicates that the projections of $\mathbf{f}_{in}$ and $\mathbf{f}_{jn}$ on $X_i \cap X_j$ are equal. Thus, our design of the matrix $\mathbf{F}_i$ is given by

- The column vector $\mathbf{f}_{i1}$ is the angle bisector of the cell i.
- The angle between $\mathbf{f}_{im}$ and $\mathbf{f}_{i(m+1)}$ is fixed to be $\pi/90$.

A case of 4 cells with matrices $\mathbf{F}_i$ of 2 rows is shown in Figure 1B.

Example: In Figure 1, the matrices $\mathbf{E}_i$ and $\mathbf{F}_i$ are given by

$$\mathbf{E}_1 = \begin{bmatrix} 0.707 & 0.707 \\ -0.707 & 0.707 \end{bmatrix}, \quad \mathbf{E}_2 = \begin{bmatrix} 0.707 & -0.707 \\ 0.707 & 0.707 \end{bmatrix}, \quad \mathbf{E}_3 = \begin{bmatrix} -0.707 & -0.707 \\ 0.707 & -0.707 \end{bmatrix}, \quad \mathbf{E}_4 = \begin{bmatrix} -0.707 & 0.707 \\ -0.707 & -0.707 \end{bmatrix},$$

$$\mathbf{F}_1 = \begin{bmatrix} 0 & 1 \\ 0.0349 & 0.9994 \end{bmatrix}, \quad \mathbf{F}_2 = \begin{bmatrix} 1 & 0 \\ 0.9994 & 0.0349 \end{bmatrix}, \quad \mathbf{F}_3 = \begin{bmatrix} 0 & -1 \\ -0.0349 & -0.9994 \end{bmatrix}, \quad \mathbf{F}_4 = \begin{bmatrix} -1 & 0 \\ -0.9994 & -0.0349 \end{bmatrix}.$$

3.2.2 Design of system dynamics $\mathbf{A}_i$

In this subsection, we are going to build a class of stable switched linear systems that do not admit common quadratic Lyapunov functions. The main idea is to design an ellipse-like *Focus* phase portrait within each polyhedral cell. To avoid the existence of a common Lyapunov function, each two adjacent "ellipse" have a certain angle, which is equal to the angle between each two adjacent cells. An example of the phase portrait of a 2-dimensional switched linear system with 4 cells is shown in Figure 2A.

To construct a system with n cells in Figure 3, we define

$$\mathbf{A}_i = \mathbf{V}_i \mathbf{\Sigma} \mathbf{V}_i^{-1}, \text{ for } i = 1, 2, ..., n, \tag{8}$$

where, $\mathbf{V}_i = \mathbf{R}^{i-1}\mathbf{B}$, and

$$\mathbf{R} = \begin{bmatrix} \cos(2\pi/n) & -\sin(2\pi/n) \\ \sin(2\pi/n) & \cos(2\pi/n) \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 1 & 0 \\ 0 & 3 \end{bmatrix}, \quad \mathbf{\Sigma} = \begin{bmatrix} -2 & 10 \\ -10 & -2 \end{bmatrix}.$$

Thus, the switched linear system with $\dot{\mathbf{x}} = \mathbf{A}_i \mathbf{x}$ will behave as shown in Figure 2A. Trajectories of any initial states will tend to origin, and one trajectory with $\mathbf{x}_0 = [-3 \quad 3]^\top$ is shown in Figure 2B. It is clear that the trajectory is bounded by a flower-like piecewise quadratic Lyapunov function(denoted by black dash lines).

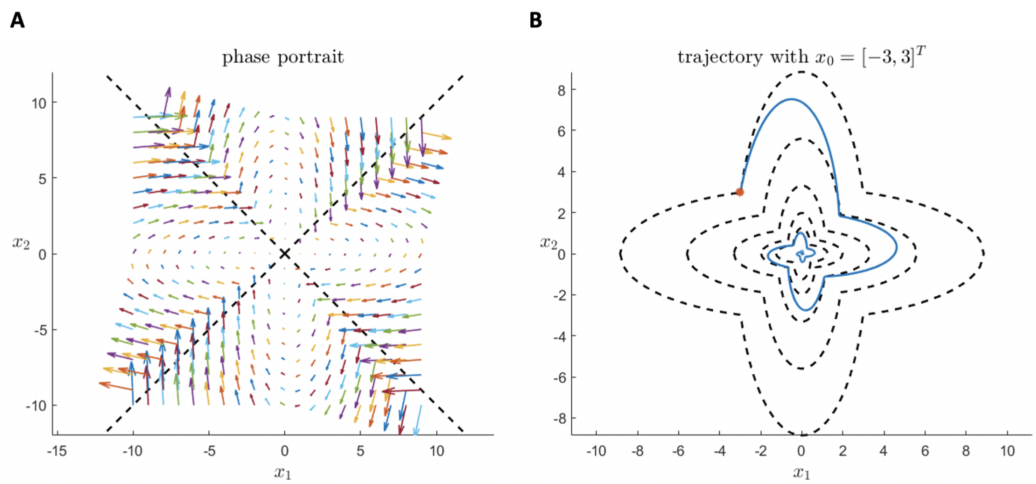

Figure 2: **An example of 2-dimensional switched linear system with 4 cells. A:** Phase portrait. **B:** Trajectory with initial condition $[-3 \quad 3]^\top$.

3.2.3 Realization of arbitrary dimension

To verify the time complexity, we need to extend the switched linear system designed in the last subsection to arbitrary dimension. Based on the 2-dimensional system design, which ensures the nonexistence of global Lyapunov function, we follow the same rules of space partition and add a stable eigenvalue for each new dimension. Under the same partition rules, it is easy to extend matrices $\mathbf{E}_i$ and $\mathbf{F}_i$ to arbitrary dimension. A 3-dimensional space partition with 4 cells is shown in Figure 3A.

Example: Assume we have 2-dimensional system with 4 cells:

$$\mathbf{A}_i, \quad \mathbf{E}_i = \begin{bmatrix} \mathbf{e}_{i1}^\top \\ \mathbf{e}_{i2}^\top \end{bmatrix}, \quad \mathbf{F}_i = \begin{bmatrix} \mathbf{f}_{i1}^\top \\ \mathbf{f}_{i2}^\top \end{bmatrix}.$$

If we increase the dimension to 3, these matrices will be

$$\mathbf{A}_i' = \begin{bmatrix} \mathbf{A}_i & 0 \\ 0 & p \end{bmatrix}, \quad \mathbf{E}_i' = \begin{bmatrix} \mathbf{E}_i & 0 \end{bmatrix}, \quad \mathbf{F}_i' = \begin{bmatrix} \mathbf{F}_i & 0 \end{bmatrix} \text{ for } p < 0.$$

One trajectory of this 3-dimensional switched linear system is shown in Figure 3B.

3.3 Time complexity verification

In this section, we verify the relationship between time complexity and the number of cells and the dimension of space. We use our design of switched linear system in Section 3.2 to generate system with arbitrary number of cells and dimension and record the execution time of Algorithm 1.

3.3.1 Number of cells

First, we test the time complexity with the number of cells. Since for each case, there may be numerical problems with some number of rows of $\mathbf{F}_i$, we try the number of rows from 1 to 10 for each case and use the output, without the numerical problem, with the least number of rows. The final results show that all numbers of rows of $\mathbf{F}_i$ are less than 5 and thus we ignore the effect of the size of $\mathbf{F}_i$ on the time complexity. We simulate the system with cell number from 4 to 400. The simulation result and the linear fitting function is shown in Figure 4A, which is consistent with Proposition 1, i.e., the relationship between execution time and number of cells is linear.

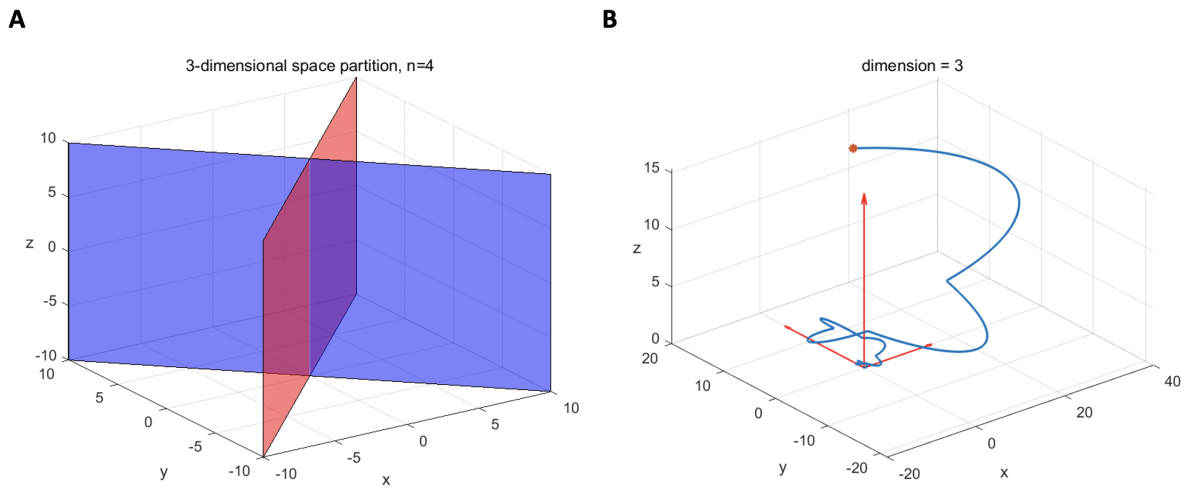

Figure 3: **An example of 3-dimensional switched linear system with 4 cells. A:** State space partition with 4 cells. **B:** Trajectory with an arbitrary initial condition.

3.3.2 Dimension

Next, we test the time complexity with dimension. As mentioned in Section 3.2.3, the basic 2-dimensional system we use is same as Example 1 in [9]. We increase the dimension of the switched linear system while ensuring the stability and nonexistence of common quadratic Lyapunov function. We simulate the system with dimension from 2 to 100. The result and the fourth-order fitting function is shown in Figure 4B, which is consistent with Proposition 1, i.e., the relationship between execution time and dimension is polynomial.

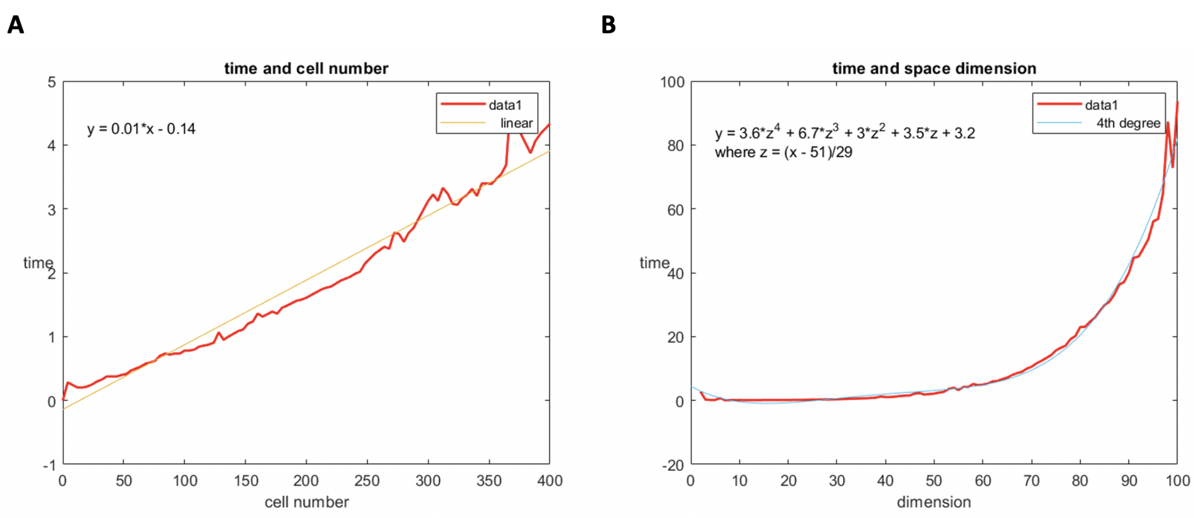

Figure 4: **Complexity verification. A:** Execution time and cell number. **B:** Execution time and dimension.

4 Nonlinear dynamics approximation

For nonlinear systems, it is possible to use a piecewise linear system to make approximation of the nonlinear system and apply the piecewise quadratic Lyapunov method to reveal stability property of the nonlinear system. To construct a piecewise linear system approximating the original nonlinear system, we partition the space around a equilibrium into several polyhedral cells as in Figure 1A. Within each of the cell, we choose a point near the equilibrium and linearize the system at this point to get $\mathbf{A}_i$ for this cell.

4.1 Prey predatory system

In the book [11], we get a prey predatory system, which can be described as

$$\dot{x_1} = x_1(1 - x_1 - ax_2) \quad \dot{x_2} = bx_2(x_1 - x_2),$$

where, x_1 and x_2 are dimensionless variables proportional to the prey and predator populations, respectively, and a and b are positive constants. For simplicity, we choose $a = 1$ and $b = 0.5$. One can verify that there are three equilibrium points and only one is stable, which is $[0.5 \quad 0.5]^\top$. To show stability of this equilibrium point by a piecewise quadratic Lyapunov function, we partition the state space into 4 cell and linearize the system at each of the 4 red circles, then compute the piecewise Lyapunov function.

The results are shown in Figure 5, where arrows show the phase portrait of the original nonlinear system, the four red circles are locations where linearizations take place, the closed area circled by black dash line denotes $V(\mathbf{x}) = c$, where c is a positive real number and $V(\mathbf{x})$ is the piecewise quadratic Lyapunov function. Hence, this equilibrium point is locally stable.

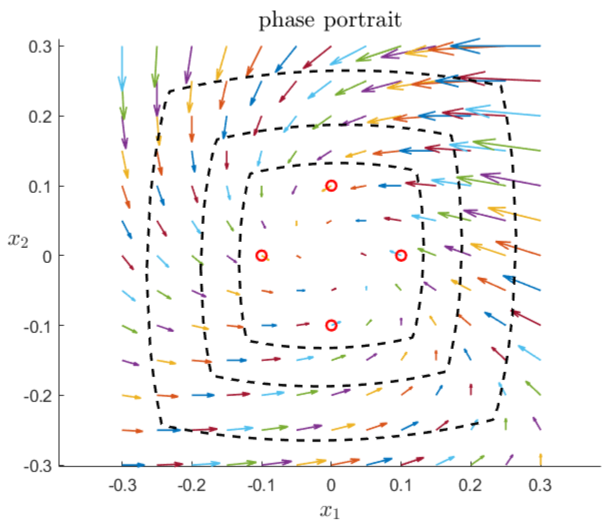

Figure 5: **Phase portrait of nonlinear system.**

5 Discussion

When generating $\mathbf{A}_i$, we used matrix $\mathbf{B}$ to represent the length of major axis and minor axis. This results in a ellipse-like phase portrait. However, systems without a common Lyapunov function could have phase portraits of other shapes. This is a limitation of the model generation. Besides, in higher dimension system, we keep the ellipse-like dynamics within the first two dimensions and let the eigenvalues of rest dimensions be just negative real numbers. This makes the phase portrait simple. Hence, systems with more complicated phase portrait could be further studied.

For matrices $\mathbf{F}_i$, there is only one requirement mentioned in paper [9]:

$$\mathbf{F}_i\mathbf{x} = \mathbf{F}_j\mathbf{x} \quad \mathbf{x} \in X_i \cap X_j$$

Our design of $\mathbf{F}_i$ meets the requirement. But when testing the complexity, the SDP solver may have numerical problems with size of F_i increasing. In some cases, there is even no size that can prevent numerical problems of the solver. For this reason, investigation of the effect of the size of $\mathbf{F}_i$ is impossible and as a result, we must ignore this effect when increasing cell number and dimension. In Section 3.3.1, the number of rows of matrix $\mathbf{F}_i$ is between 1 and 5. In Section 3.3.2, the size of matrix $\mathbf{F}_i$ is $(d+2)d$ where d is the dimension of the space.

For the nonlinear dynamics approximation, the original purpose is to find a nonlinear system that is stable but the stability is inconclusive with traditional linearization. If we can find a piecewise Lyapunov function of the approximation of that nonlinear system, we may have some conclusions of the original nonlinear system. This can be a good future work for this project. Besides, we also think about dividing the space into many hyperboxs and then linearize the nonlinear system, which should improve the precision of the linearization but increase the complexity.

6 Conclusion

In this paper, we implemented the algorithm for computing piecewise quadratic Lyapunov functions for switched linear system. We proposed the time and space complexity of this algorithm using the SDP solver SeDuMi. To verify our proposition, we designed a class of stable switched linear systems that do not have common Lyapunov functions with arbitrary dimension and number of cells. Finally, we applied the algorithm to a nonlinear system and analyze the stability of it via piecewise linearization. For future works, the effect of the size of matrix $\mathbf{F}_i$ on the performance of the algorithm should be discussed, which may need a new design of matrix $\mathbf{F}_i$ for each cell. More theoretical derivation should be done on the relation of the stability between switched linear system and the original nonlinear system.

Contributions

Can Chen, Bingwen Yang and Ming Li contribute equally to this work. Can Chen worked on the implementation of the algorithm in MATLAB, and the complexity analysis of the algorithm. Bingwen Yang worked on the realization of a system having no common quadratic Lyapunov functions, plotting graphs of Lyapunov functions and realization of nonlinear dynamics approximation. Ming Li worked on the space partition and simulation verifying algorithm complexity with number of cells and dimension.

References

[1] A. Bemporad, G. Ferrari-Trecate, and M. Morari. Observability and controllability of piecewise affine and hybrid systems. *IEEE transactions on automatic control*, 45(10):1864–1876, 2000.

[2] M. S. Branicky. Multiple lyapunov functions and other analysis tools for switched and hybrid systems. *IEEE Transactions on automatic control*, 43(4):475–482, 1998.

[3] R. W. Brockett et al. Asymptotic stability and feedback stabilization. *Differential geometric control theory*, 27(1):181–191, 1983.

[4] D. Cheng. Controllability of switched bilinear systems. *IEEE Transactions on Automatic Control*, 50(4):511–515, 2005.

[5] A. Feuer, G. C. Goodwin, and M. Salgado. Potential benefits of hybrid control for linear time invariant plants. In *Proceedings of the 1997 American Control Conference (Cat. No.97CH36041)*, volume 5, pages 2790–2794 vol.5, 1997.

[6] R. M. Freund. Introduction to semidefinite programming (sdp). *Massachusetts Institute of Technology*, pages 8–11, 2004.

[7] P. Gahinet and I. MathWorks. *LMI Control Toolbox for Use with MATLAB*. MathWorks partner series. MathWorks, 1995.

[8] D. Gómez-Gutiérrez, C. R. Vázquez, A. Ramírez-Teviño, and S. Di Gennaro. On the observability and observer design in switched linear systems. In *New Trends in Observer-Based Control*, pages 73–118. Elsevier, 2019.

[9] M. Johansson and A. Rantzer. Computation of piecewise quadratic lyapunov functions for hybrid systems. In *1997 European Control Conference (ECC)*, pages 2005–2010. IEEE, 1997.

[10] R. M. Jungers, A. Kundu, and W. Heemels. Observability and controllability analysis of linear systems subject to data losses. *IEEE Transactions on Automatic Control*, 63(10):3361–3376, 2017.

[11] H. K. Khalil and J. W. Grizzle. *Nonlinear systems*, volume 3. Prentice hall Upper Saddle River, NJ, 2002.

[12] H. Lin and P. J. Antsaklis. Stability and stabilizability of switched linear systems: a survey of recent results. *IEEE Transactions on Automatic control*, 54(2):308–322, 2009.

[13] K. S. Narendra and J. Balakrishnan. Adaptive control using multiple models. *IEEE Transactions on Automatic Control*, 42(2):171–187, 1997.

[14] D. Peaucelle, D. Henrion, Y. Labit, and K. Taitz. User's guide for sedumi interface 1.04. *LAAS-CNRS, Toulouse*, 2002.

[15] J. F. Sturm. Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones. *Optimization methods and software*, 11(1-4):625–653, 1999.

[16] R. A. van den Berg, A. Y. Pogromsky, and J. Rooda. Convergent design of switched linear systems. *IFAC Proceedings Volumes*, 39(5):6–11, 2006.

[17] L. Vandenberghe and S. Boyd. Semidefinite programming. *SIAM review*, 38(1):49–95, 1996.