


ITC 4/46 Journal of Information Technology and Control Vol. 46 / No. 4 / 2017 pp. 459-469 DOI 10.5755/j01.itc.46.4.16072 © Kaunas University of Technology	Computing Regions of Stability for Limit Cycles of Piecewise Affine Systems	
	Received 2016/08/171	Accepted after revision 2017/11/14
	 http://dx.doi.org/10.5755/j01.itc.46.4.16072	

Computing Regions of Stability for Limit Cycles of Piecewise Affine Systems

Yu Chen, Yue Sun, Chun-Sen Tang, Yu-Gang Su

Chongqing University, School of Automation, Chongqing University, Chongqing 400044, China,
email: thecy18@gmail.com, syue06@cqu.edu.cn, cstang@cqu.edu.cn, su7558@qq.com

Aiguo Patrick Hu

The University of Auckland, Department of Electrical and Computer Engineering, Auckland 1142, New Zealand,
email: a.hu@auckland.ac.nz

Corresponding author: syue06@cqu.edu.cn

This paper proposes an improved algorithm to compute the regions of stability for limit cycles of piecewise affine systems. Instead of using convex optimization algorithms, such as solving linear matrix inequalities and sum-of-square programming, the genetic algorithm (GA) is used to obtain the final results. With the help of GA, both the constraints and objective functions can be nonconvex. As a result, larger guaranteed regions of stability are achieved. On the basis of the impact map and Lyapunov stability theory, the stability conditions are analyzed. Algorithm-friendly criteria, both convex and nonconvex, are developed. Randomly generated solutions are usually infeasible; hence, we generate the initial GA population by convex optimization. To improve the initial population's diversity, we use multiple convex objective functions to produce different initial solutions. Other application-specific parts of GA, such as computing the fitness of solutions, are also introduced in detail. A sample system is analyzed to illustrate the effectiveness of the proposed method.

KEYWORDS: piecewise affine systems, region of stability, limit cycle, convex optimization, genetic algorithm.

1. Introduction

Piecewise affine (PWA) systems are a subset of hybrid dynamic systems. PWA systems can be used to model nonlinear behavior, such as switching, saturation,

delay, dead zones, gaps, and hysteretic. PWA systems can also be used to approximate general nonlinear systems. Except for equilibrium points [1], [2], the

steady state of PWA systems may also be limit cycles. Some limit cycles are introduced by nonlinear controllers [9]. Some other limit cycles are intended oscillations, such as soft-switching DC-AC converters [10], [12]. The numerical solutions of limit cycles can be acquired by stroboscopic mapping[6], and the local stability of limit cycles can be checked by linearizing the associated Poincaré map. However, checking the global stability and computing the region of stability are difficult.

Stability analysis is an essential step in designing a PWA system. A region of stability is a set of states that converge to a specific equilibrium point or limit cycle. This region is the stable working range of the corresponding PWA system. In a global stable system, the region of stability refers to the entire state space. If a system is not globally stable or if multiple stable equilibrium points and limit cycles are present, the entire region of stability is usually an irregularly shaped subregion of the state space. Obtaining the entire region of stability is difficult. Hence, most methods focus on computing a guaranteed region of stability. Guaranteed regions of stability are useful for many applications. However, they are only acceptable if the regions are sufficiently large. Otherwise, the designed system is conservative.

By using the impact map, i.e., the map from one switching surface to the next switching surface, Gonçalves analyzed both global and local stabilities of limit cycles [3]-[6]. The solution was computed by solving linear matrix inequalities (LMIs). However, the criteria are conservative, and the sizes of results are not considered. Thus, the resultant regions are very small compared with the entire regions of stability. In [8], [11], transverse linearization was employed to analyze regions of stability for limit cycles of hybrid dynamical systems. The result was obtained by sum-of-square (SoS) programming.

Both LMI solving and SoS programming are convex optimization methods. Their constraints should be convex and the target is minimizing convex objective functions. Nonconvex constraints must be relaxed to convex constraints, which leads to conservative results. Furthermore, nonconvex objectives, such as maximizing the volume of resultant regions of stability, cannot be addressed.

This paper proposes an improved algorithm for computing the regions of stability for limit cycles of PWA

systems. On the basis of the impact map, criteria softer than the ones in [4] are proposed. Some criteria are convex, whereas the others are not. On the basis of the convex conditions, with different convex objective functions, different guaranteed regions of stability are obtained by convex optimization. Using these regions as the initial population, the final results are obtained by genetic algorithm (GA). Owing to the GA, both the constraints and objective functions need not to be convex. With the proposed algorithm, the resulting guaranteed regions of stability become larger than the result of existing methods.

The rest of this paper is organized as follows. In Section 2, the model, notations, and the problem are introduced. In Section 3, algorithm-friendly criteria of stability are proposed. The algorithm is introduced in Section 4. In Section 5, an example system is analyzed. Section 6 concludes this paper. Finally, some proofs are given in the Appendix.

2. Problem Description

2.1. Model and Limit Cycle

PWA systems are characterized by a set of affine linear systems as follows:

$$\dot{x} = A_i x + b_i \quad (1)$$

where $x \in \mathbb{R}^n$ is the state of the system, $A_i \in \mathbb{R}^{n \times n}$, $b_i \in \mathbb{R}^n$.

In the state space, the switching occurs at some switching surfaces consisting of $(n-1)$ dimensional hyperplanes:

$$S_i = \{x \mid C_i x = d_i\} \quad (2)$$

where $C_i \in \mathbb{R}^{1 \times n}$, $d_i \in \mathbb{R}$, $i \in \{1, \dots, N\}$, and N is the number of switching surfaces. Of each C_i , at least one element should be nonzero, i.e.,

$$C_i \neq \mathbf{0}. \quad (3)$$

In this paper, it is assumed that the concerned limit cycle is already known. The algorithm for computing the limit cycles of PWA systems is described in [6]. For a system with multiple stable limit cycles, the algorithm can be applied to each of them respectively.

Assume the limit cycle goes through k subsystems and passes k switching surfaces per cycle. Without loss of generality, assume that the limit cycle starts at the state x_1^* on the switching surface S_{i_1} , enters into subsystem 1, passes S_{i_2} at x_2^* after time t_1^* , enters into subsystem 2, and so on. When the limit cycle enters subsystem k , it reaches x_1^* again after time t_k^* . If the limit cycle passes some switching surfaces more than once, we simply obtain some i and j , such that $S_i = S_j$. For convenience, we let index $k + 1$ and index 1 represent the same object, i.e., $x_{k+1}^* = x_1^*$, $S_{k+1} = S_1$ [4].

Assume there are N_i possible switching surfaces for states in subsystems i . Denote

$$S_{i,j} = \{x \mid C_{i,j}x = d_{i,j}\} \tag{4}$$

as these switching surfaces, where $j \in \{0, \dots, N_i - 1\}$. Without loss of generality, we let $S_{i,0} = S_{i+1}$. Thus, when the limit cycle reaches S_i , the next switching surface the limit cycle passes is $S_{i,0}$.

2.2. Impact Map

Denote $0_{n \times m}$ as the $n \times m$ matrix whose all entries are zero, and I_n as the $n \times n$ identity matrix. Define $L_{n \times m} := [0_{(n-m) \times n} \quad I_m]^T$, where $n > m > 0$.

Proposition 1. Assume $C^T = QR$ is the QR decomposition of C^T , where $C \in \mathbb{R}^{1 \times n} \setminus \{0\}$, Q is an $n \times n$ unitary matrix, and R is an $n \times 1$ upper triangular matrix. That is, all the elements are zero except the one in the first row. Assume $\Pi = QL_{n \times (n-1)}$, $x \in \mathbb{R}^{n \times 1}$, $\Delta \in \mathbb{R}^{(n-1) \times 1}$, then the following two statements are equivalent:

- 1 $x = \Pi\Delta$ and
- 2 $\Delta = \Pi^T x$ and $Cx = 0$.

Proof. The proof is given in the Appendix.

According to Proposition 1, all $x_i \in S_i$ can be represented by a unique $\Delta \in \mathbb{R}^{n-1}$. The relations are written as $x_i = x_i^* + \Pi_i \Delta$, where $\Delta \in \mathbb{R}^{n-1}$, $\Pi_i = Q_i L_{n \times (n-1)}$ and $C_i^T = Q_i R_i$.

Define $\Phi_i(x, t)$ as

$$\Phi_i(x, t) = e^{A_i t} x + \int_0^t e^{A_i(t-\theta)} b_i d\theta. \tag{5}$$

$\Phi_i(x, t)$ is the state that starts at x , and run for time t in the affine subsystem i without switching.

Define $\tau_i(\Delta)$ as the next switching time of the state $x_i^* + \Pi_i \Delta$.

The impact map is defined as the map from one switching surface to the next switching surface. Assume that an initial state on the switching surface i is $x_i^* + \Pi_i \Delta_i$. After $\tau_i(\Delta_i)$, the state becomes $x_{i+1}^* + \Pi_{i+1} \Delta_{i+1}$. The relationship between Δ_{i+1} and Δ_i can be expressed as [6]

$$\Delta_{i+1} = H_i(t) \Delta_i \tag{6}$$

and

$$H_i(t) = \Pi_{i+1}^T \left[I + \frac{(\Phi_i(x_i^*, t) - x_{i+1}^*) C_{i+1}}{d_{i+1} - C_{i+1} \Phi_i(x_i^*, t)} \right] e^{A_i t} \Pi_i \tag{7}$$

where $t = \tau_i(\Delta_i)$. If $t = t_i^*$, $H_i(t)$ is defined as the limitation where $t \rightarrow t_i^*$.

The value of $H_i(t)$ only depends on t . Thus, the impact map can be regarded as a linear transformation parametrized by the switching time. Additional information on the impact map is described in [6].

2.3. Problem

Given the impact map, the following Lyapunov function could be constructed:

$$V(\Delta) = \|\Delta\|_{P_\Delta}^2 \tag{8}$$

where $\|\Delta\|_{P_i} = \sqrt{\Delta^T P_i \Delta}$ is the Mahalanobis distance from Δ to the origin, $i \in \{1, 2, \dots, k\}$, $P_1, P_2, \dots, P_k \in \mathbb{R}^{(n-1) \times (n-1)}$ are positive definite symmetric matrices, and P_Δ is the corresponding P_i of Δ .

Define Ω_i as

$$\Omega_i := \left\{ \Delta \mid \|\Delta\|_{P_i}^2 \leq 1 \right\}. \tag{9}$$

If for all $\Delta \in \Omega_i \setminus \{0\}$, the next switching surface of $x_i^* + \Pi_i \Delta$ is S_{i+1} and $\|\Delta\|_{P_i} > \|H_i(\tau_i(\Delta))\|_{P_{i+1}}$, then (8) is decreasing for all $\Delta \in \Omega_i \setminus \{0\}$. In other words, the limit cycle is guaranteed to be stable if the Mahalanobis distance is decreasing for all $\Delta \in \Omega_i \setminus \{0\}$. At present, the problem is finding an optimized group of Ω_i .

Each Ω_i corresponds to an $(n - 1)$ dimensional ellipsoid on the corresponding switching surface. All states in these ellipsoids converge to the limit cycle. The shapes and sizes of these ellipsoids are determined by P_i . The target is finding eligible P_1, P_2, \dots, P_k , and optimizing the sizes of the corresponding ellipsoids.

3. Criteria

In this section, algorithm-friendly criteria are developed. At first, as basis, the formulation for calculating the minimum Mahalanobis distance from a linear manifold to the origin point is introduced. For each P_i , a time interval $T_i = [t_{i-}, t_{i+}]$, where $t_{i-} < t_i^* < t_{i+}$, should be found to assist the judgment. Stability is guaranteed if the following three conditions hold:

- 1 For all $\Delta \in \Omega_i$, in the time interval $(0, t_{i+}]$, trajectories starting at $x_i^* + \Delta$ do not pass any switching surfaces except for S_{i+1} .
- 2 For all $\Delta \in \Omega_i$, trajectories starting at $x_i^* + \Delta$ passes S_{i+1} after time $\tau_i(\Delta) \in [t_{i-}, t_{i+}]$.
- 3 For all Δ that satisfies $\tau_i(\Delta) \in [t_{i-}, t_{i+}]$, the inequality $\|\Delta\|_{P_i}^2 > \|H_i(\tau_i(\Delta))\Delta\|_{P_{i+1}}^2$ holds.

3.1. Minimum Mahalanobis Distance

Assume $P \in \mathbb{R}^{n \times n}$ is a positive definite symmetric matrix; $S = \{x \mid Fx = g\}$ is a linear manifold in \mathbb{R}^n , where $F \in \mathbb{R}^{m \times n}$, $g \in \mathbb{R}^m$, $n > m \geq 1$. Under the matrix P , the minimum Mahalanobis distance from S to the origin point is defined as

$$\Gamma_P(S) = \min \{ \|x\|_P \mid x \in S \}. \tag{10}$$

$\Gamma_P(S)$ can be obtained by

$$\Gamma_P^2(S) = \min \{ x^T P x \mid x \in S \}. \tag{11}$$

Applying Lagrange multiplier method to (11), the augmented function is

$$f(x, \lambda) = x^T P x + \lambda^T (F x - g) \tag{12}$$

where $\lambda \in \mathbb{R}^m$.

The value of (11) is minimum when

$$\begin{cases} \partial f / \partial x = 2 P x + F^T \lambda = 0 \\ \partial f / \partial \lambda = F x - g = 0 \end{cases} \tag{13}$$

The solution of (13) is:

$$\begin{cases} \lambda_* = -2(FP^{-1}F^T)^{-1}g \\ x_* = -\frac{1}{2}P^{-1}F^T\lambda_* = P^{-1}F^T(FP^{-1}F^T)^{-1}g \end{cases} \tag{14}$$

The minimum value of $x^T P x$ can be obtained by substituting (14) into (11). We have

$$[\Gamma_P(S)]^2 = [P^{-1}F^T(FP^{-1}F^T)^{-1}g]^T P [P^{-1}F^T(FP^{-1}F^T)^{-1}g] \tag{15}$$

$P = P^T$; hence we have

$$(FP^{-1}F^T)^{-1} = [(FP^{-1}F^T)^{-1}]^T. \tag{16}$$

By substituting (16) into (15), we obtain

$$[\Gamma_P(S)]^2 = g^T (FP^{-1}F^T)^{-1} g. \tag{17}$$

If $m = 1$, (17) can be further simplified, and it is a concave function of the variable P . However, it is neither convex nor concave when $m > 1$.

3.2. Switching Time

On a switching surface, if multiple states pass the same future switching surface after the same time, they belong to the same $(n - 2)$ dimensional hyperplane. This property is a basis for later analysis and is proven in this section.

Assume the state $x = x_i^* + \Pi_i \Delta$ is on S_i ; its next switching surface is $S_{i,j}$, and its switching time is t . Therefore, we obtain

$$C_{i,j} \Phi_i(x_i, t) = d_{i,j}. \tag{18}$$

By substituting (5) into (18), we attain

$$C_{i,j} e^{A t} \Pi_i \Delta = -C_{i,j} [e^{A t} x_i^* + \int_0^t e^{A(t-\theta)} b_i d\theta] + d_{i,j}. \tag{19}$$

If t is known, the only variable at the left side of (19) is Δ , and the right side is a constant. Define

$$\begin{cases} F_{i,j,t} := C_{i,j} e^{A t} \Pi_i \\ g_{i,j,t} := -C_{i,j} [e^{A t} x_i^* + \int_0^t e^{A(t-\theta)} b_i d\theta] + d_{i,j} \end{cases} \tag{20}$$

On S_i , all the points, that the next switching surface is $S_{i,j}$ and the switching time is t , belong to an $(n - 2)$ dimensional hyperplane determined by $F_{i,j,t}$ and $g_{i,j,t}$. Denote by $\Lambda_{i,j,t}$ as the following hyperplane:

$$\Lambda_{i,j,t} := \{ \Delta \mid F_{i,j,t} \Delta = g_{i,j,t} \}. \tag{21}$$

Notably, being on $\Lambda_{i,j,t}$ is necessary but insufficient for passing S_i after t . This notion is due to the fact that the points on $\Lambda_{i,j,t}$ may pass a switching surface before t , or may not be in the corresponding subsystem.

3.3. Condition 1

If $\Gamma_{P_i}(\Lambda_{i,j,t}) > 1$ holds for all $j \in \{1, \dots, N_i - 1\}$ and $t \in (0, t_+]$, then Condition 1 holds. This result is achieved because in this case, all points on S_i that could pass the switching surface $S_{i,j}$ before t_{i+} are at the outside of Ω_i .

By substituting (21) into (17), we write the condition as

$$g_{i,j,t}^T (F_{i,j,t} P_i^{-1} F_{i,j,t}^T)^{-1} g_{i,j,t} > 1 \tag{22}$$

$g_{i,j,t} \in \mathbb{R}$; thus, (22) is equivalent to

$$g_{i,j,t}^2 - F_{i,j,t} P_i^{-1} F_{i,j,t}^T > 0. \tag{23}$$

On the basis of the Schur complement method, (23) is equivalent to

$$\begin{bmatrix} P_i & F_{i,j,t}^T \\ F_{i,j,t} & g_{i,j,t}^2 \end{bmatrix} \succ 0 \tag{24}$$

where $P \succ 0$ means P is a positive definite matrix. This LMI of the variable P_i is a convex constraint.

3.4. Condition 2

Assumption 1. For all x_0 in the following set:

$$\{x | x = \Phi_i(\Pi_i \Delta + x_i^*, t), t \in [t_{i-}, t_{i+}], C_{i+1} x = d_{i+1}\} \tag{25}$$

the following inequalities holds:

$$\begin{cases} C_{i+1} \dot{\Phi}_i(x_0, 0) \neq 0 \\ C_{i+1} \dot{\Phi}_{i+1}(x_0, 0) \neq 0 \end{cases} \tag{26}$$

Theorem 2. Under Assumption 1 and Condition 1, if

$$\|\Delta\|_{P_i} \leq \min\{\Gamma_{P_i}(\Lambda_{i,0,t_{i-}}), \Gamma_{P_i}(\Lambda_{i,0,t_{i+}})\} \tag{27}$$

then $\tau_i(\Delta) \in [t_{i-}, t_{i+}]$.

Proof. The proof is given in the Appendix.

Assumption 1 states that when a trajectory reaches a switching surface, the trajectory should not be

tangential to the switching surface at both sides. The points that may be tangential after a same time value t forms a linear manifold. Moreover, Assumption 1 holds if all the Mahalanobis distances from these linear manifolds to the origin are larger than 1, i.e., outside of Ω_i . Therefore, Assumption 1 holds if for all $t \in [t_{i-}, t_{i+}]$, the following inequality holds:

$$\begin{cases} \Gamma_{P_i} \left(\left\{ \Delta \begin{cases} C_{i+1} \Phi_i(x_i^* + \Pi_i \Delta, t) = d_{i+1}, \\ C_{i+1} \dot{\Phi}_i(x_i^* + \Pi_i \Delta, t) = 0 \end{cases} \right\} \right) > 1 \\ \Gamma_{P_i} \left(\left\{ \Delta \begin{cases} C_{i+1} \Phi_i(x_i^* + \Pi_i \Delta, t) = d_{i+1}, \\ C_{i+1} \dot{\Phi}_{i+1}(\Phi_i(x_i^* + \Pi_i \Delta, t), 0) = 0 \end{cases} \right\} \right) > 1 \end{cases} \tag{28}$$

The result in (17) can be applied herein. $F \in \mathbb{R}^{2 \times (n-1)}$ in this case; hence, (28) is a nonconvex constraint. It can be relaxed to the following convex constraint:

$$\begin{cases} \Gamma_{P_i} \left(\left\{ \Delta \begin{cases} C_{i+1} \dot{\Phi}_i(x_i^* + \Pi_i \Delta, t) = 0 \end{cases} \right\} \right) > 1 \\ \Gamma_{P_i} \left(\left\{ \Delta \begin{cases} C_{i+1} \dot{\Phi}_{i+1}(\Phi_i(x_i^* + \Pi_i \Delta, t), 0) = 0 \end{cases} \right\} \right) > 1 \end{cases} \tag{29}$$

However, Assumption 1 can be omitted when computing initial solutions with convex optimization. The assumption can be checked in the GA and will not influence the correctness of the final results.

According to Theorem 2, if Assumption 1 holds, and

$$\min\{\Gamma_{P_i}(\Lambda_{i,0,t_{i-}}), \Gamma_{P_i}(\Lambda_{i,0,t_{i+}})\} \geq 1 \tag{30}$$

then $\|\Delta\|_{P_i} \leq 1$ implies $\tau_i(\Delta) \in [t_{i-}, t_{i+}]$. The result of (24) can be applied to (30) directly.

3.5. Condition 3

Condition 3 can be written as

$$\Delta^T P_i \Delta - \Delta^T H_i(t)^T P_{i+1} H_i(t) \Delta > 0 \tag{31}$$

where $t = \tau_i(\Delta)$. One of the sufficient conditions of (31) is as follows:

$$P_i - H_i(t)^T P_{i+1} H_i(t) \succ 0. \tag{32}$$

It is a convex constraint of the variable P_i .

4. Algorithm

At this point, the problem is finding an optimized group of P_i , t_{i-} , and t_{i+} , where $i \in \{1, 2, \dots, k\}$, such that (24), (29), (30), and (35) holds. Furthermore, the total volume of the ellipsoids determined by P_i matrices should be the largest. is nonconvex and maximizing the volume is not a convex objective; hence, convex optimization algorithms could not deal with this problem.

In this paper, GA is used to obtain the optimized result. To apply GA, three main problems must be solved. First, (24), (29), and (35) require the inequalities satisfied for all $t \in [t_{i-}, t_{i+}]$. However, checking all these time values is impossible because $[t_{i-}, t_{i+}]$ is an infinity set. Second, the initial populations of GA are usually randomly generated. However, in this problem, the conditions are strict; thus, randomly generated solutions are usually infeasible. Third, for the same reason, if genetic operators, i.e., mutation and crossover, are applied to P_i , t_{i-} , and t_{i+} directly, the newly generated solutions are usually infeasible.

The first problem is solved by gridding the time intervals and checking their finite subsets instead. The second problem is solved by generating initial solutions with convex optimization. The third problem is solved by converting candidate solutions to lists of numbers with only the shape information of ellipsoids and reconstructing P_i , t_{i-} , and t_{i+} when necessary. The detailed methods are introduced later in this section.

The other parts of GA, such as crossover, mutation, selection, and termination, are not the main concerns of this paper. Related research is fairly mature and can be easily applied herein. Thus, the studies are omitted in this paper.

4.1. Objective Functions of Optimization

Objective functions are necessary in both convex optimization and GA. Proper objective functions help attain better results. For convex optimization, the objective functions must be convex. By contrast, for GA, the objective functions could be any real-valued functions.

In many cases, the regions of stability with maximum volumes are preferred. The volume of the ellipsoid Ω_i is proportional to $\det(P_i^{-1})$. Therefore, maximizing the volume is equivalent to maximizing $\det(P_i^{-1})$. Unfortunately, $\det(P_i^{-1})$ is a convex function and cannot be maximized by convex optimization.

To obtain initial populations by convex optimization, convex objective functions are necessary. For GA, the diversity of the initial population is important, i.e., the number of different initial solutions are essential. However, whether the initial populations are optimized or not is irrelevant. Minimizing $\lambda_{\max}(P_i)$ and $tr(P_i)$, which lead to relatively large resultant ellipsoids, can be used as the objectives in convex optimization. In the presence of multiple switching surfaces, the objectives can be

- 1 minimizing the sum($\lambda_{\max}(P_i)$) and sum($tr(P_i)$);
- 2 minimizing the min($\lambda_{\max}(P_i)$) and min($tr(P_i)$);
- 3 minimizing a specific $\lambda_{\max}(P_i)$ and $tr(P_i)$, such as $\lambda_{\max}(P_1)$ and $tr(P_1)$.

4.2. Gridding the Time Interval

Checking all the values in the switching time sets $[t_{i-}, t_{i+}]$ is impossible. By gridding, finite subsets of switching time values can be obtained. The correctness can be guaranteed as long as the gridding step is sufficiently small. However, if the step is excessively small, the algorithm speed would be slow.

During solving, using a relatively large gridding step can increase the speed significantly. In most cases, such strategy does not influence the correctness of the solutions. Lastly, to ensure correctness, we must check the solutions with a sufficiently small gridding step. If checking failed, we can either decrease the initial gridding step, or shrink the final solutions to satisfy the criteria.

Assume the gridding step is $t_{d,i} = t_i^*/d$. In practice, d from 100 to 1000 can be used as the large step, and should be larger than 10000 for the final checking. After gridding, $(t_{i+} - t_{i-})/t_{d,i}$ discrete time values are generated.

4.3. Computing Initial Populations

The P_i matrices are obtained as the result of convex optimization. To generate the constraints of convex optimizations, t_{i-} and t_{i+} should be known. These variables can be guessed by starting at t_i^* and increasing their differences to t_i^* until the convex optimization problem is infeasible. For each guess, different objective functions can be used and multiple groups of results are obtained. Each of the results can be used as an initial solution of GA.

The process for computing initial populations with convex optimization is as follows:

- 1 Define a variable δ and set its value as a small positive number, such as 0.05.
- 2 Use $t_{i-} = t_i^* \times (1 - \delta)$, $t_{i+} = t_i^* / (1 - \delta)$. Grid the time intervals $[t_{i-}, t_{i+}]$.
- 3 Substitute the gridded time values into (24), (30) and (35) to generate the convex constraints.
- 4 With each convex objective function, solve a convex optimization problem, and a group of P_i matrices is obtained.
- 5 Increase δ and repeat step 2 until the convex optimization problem is infeasible.

Assume δ is increased by p times, and q convex objective functions are applied. The total $p \times q$ solutions constitute the initial population of the GA. Only the P_i matrices are necessary in the later steps.

4.4. Representing the Solutions

In GA, only the shapes of the ellipsoids determined by P_i matrices are necessary. The optimized sizes can be computed if the shapes are known. All ellipsoids are zoomed in or out with the same ratio, which could be represented by $\gamma \in \mathbb{R}^+$. The ellipsoids determined by P_i and γP_i possess the same shape but different sizes. For a group of P_1, \dots, P_k , finding the largest γ , such that $P_1/\gamma, \dots, P_k/\gamma$ satisfy all three conditions, is equivalent to zooming in the ellipsoids until the ellipsoids intersect with one of the hyperplanes determined by the constraints. If γ is known, the result of the objective function can be easily computed with $P_1/\gamma, \dots, P_k/\gamma$.

When representing a candidate solution, the amount of necessary variable real numbers is smaller than the amount of elements in all P_i matrices. First, P_i matrices are symmetric; hence, only the upper or lower triangular elements of P_i are necessary. Second, all elements in P_i is multiplied by γ in the algorithm; thus, P_1, \dots, P_k and $\alpha P_1, \dots, \alpha P_k$ are equivalent. To eliminate duplication, we can enforce one element from these matrices to be a fixed value. For example, we can divide all elements in P_i with the upper-left element of P_i (if the upper-left element of P_i is nonzero). This action makes the upper-left element of P_i fixed as 1, and thus can be omitted. Third, for symmetric systems, the number of variables can be further reduced. In conclusion, all candidate solutions can be represented as a list of $k \times [n \times (n-1)/2] / s - 1$ real numbers, where $s \in \{1, 2, 4, 8, \dots\}$. The value of s depends on the system symmetry. For example, $s = 1$ if the system is asymmetric.

Except for computing the fitness, all the other genetic operators can be easily applied to these lists of real numbers. Furthermore, only the shape information is stored; after mutation and crossover, the resultant solutions are usually applicable.

4.5. Computing the Fitness

To compute the fitness for a candidate solution, an optimized γ should be obtained. With γ known, the P_i matrices can be easily reconstructed, and the value of the objective functions can be computed. The process is as follows:

- 1 P_1, \dots, P_k matrices are reconstructed from the list of real numbers.
- 2 Let $t_{i-} = t_{i+} = t_i^*$.
- 3 γ_{i1} is computed as the smallest γ for $t_i \in [0, t_{i+}]$, such that (22) still holds. This step is for Condition 1.
- 4 γ_{i2-} is computed as the smallest γ for $t_i \in [t_{i-}, t_i^*]$, such that (28) holds. Similarly, γ_{i2+} is computed as the smallest γ for $t_i \in [t_i^*, t_{i+}]$, such that (28) holds. This step is for Assumption 1.
- 5 For every t_{i-} and t_{i+} , a γ_{i3-} and a γ_{i3+} can be determined by (30). This step is for Condition 2.
- 6 Let $\gamma_{i-} = \min\{\gamma_{i2-}, \gamma_{i3-}\}$, $\gamma_{i+} = \min\{\gamma_{i1}, \gamma_{i2+}, \gamma_{i3+}\}$;
- 7 For every pair of t_{i-} and t_{i+} , check whether (35) still holds. This step is for Condition 3.
- 8 Let $\gamma_{\min} = \min\{\gamma_{i-}, \gamma_{i+} \mid i \in \{1, \dots, k\}\}$:
 - a. If $\gamma_{\min} \leq 0$, this group of P_1, \dots, P_k is invalid.
 - b. Otherwise, a t_{i-} or t_{i+} exists, such that γ_{\min} is equal to the associate γ_{i-} or γ_{i+} . This t_{i-} is decreased or t_{i+} is increased by $t_{d,i}$.
 - c. Steps 3~7 are repeated with the new t_{i-} or t_{i+} to obtain new γ values. Among these variables, $\gamma_{i1}, \gamma_{i2-}$, and γ_{i2+} decrease or remain unchanged because of their definition, whereas γ_{i3-} and γ_{i3+} increase because of Theorem 2.
- 9 Step 8 is repeated until inequality (35) does not hold, or $\min(\gamma_{i3-}, \gamma_{i3+}) \geq \min\{\gamma_{\min}, \gamma_{i2}, \gamma_{i1}\}$.

With $\gamma = \gamma_{\min}$, we can compute the fitness, i.e., value of the objective function, which depends on the application. The only requirement is that its result should be a real number, such as the total volume of ellipsoids determined by P_i .

Notably, when steps 3 and 4 are repeated, only the new t_{i-} and t_{i+} are necessary to be checked. The old ones are already checked in previous iterations.

5. An Example

The example is a biological application known as neural oscillator [4]. It is a fourth-order system with states $x = (x_1, x_2, x_3, x_4)^T$ given by

$$\begin{cases} \dot{x}_1 = -10x_1 - 20x_2 - 20[x_3]^+ + 10 \\ \dot{x}_2 = 5[x_1]^+ - 5x_2 \\ \dot{x}_3 = -20[x_1]^+ - 10x_3 - 20x_4 + 10 \\ \dot{x}_4 = 5[x_3]^+ - 5x_4 \end{cases} \quad (33)$$

where $[u]^+ = \max\{0, u\}$ is an on/off controller. The system has a local stable limit cycle with $x_1^* \approx [0, 0.061, 0.341, 0.353]^T$, $x_2^* \approx [0.399, 0.142, 0, 0.259]^T$, $x_3^* \approx [0.341, 0.353, 0, 0.061]^T$, $x_4^* \approx [0, 0.259, 0.399, 0.142]^T$, and $t^* \approx 0.16+0.288+0.16+0.288$.

The initial population is generated with $t_{i-} = t_i^* \times (1 - \delta)$, $t_{i+} = t_i^* / (1 - \delta)$, $\delta \in \{0.05, 0.1, \dots, 0.7\}$. The objective functions to be minimized are $\text{sum}(\text{trace}(P_i))$, $\text{sum}(\lambda_{\max}(P_i))$, $\min(\text{trace}(P_i))$, and $\min(\lambda_{\max}(P_i))$. The final result is obtained by minimizing $\text{sum}(\det(P_i))$, which is equivalent to maximizing the volume of the region of stability.

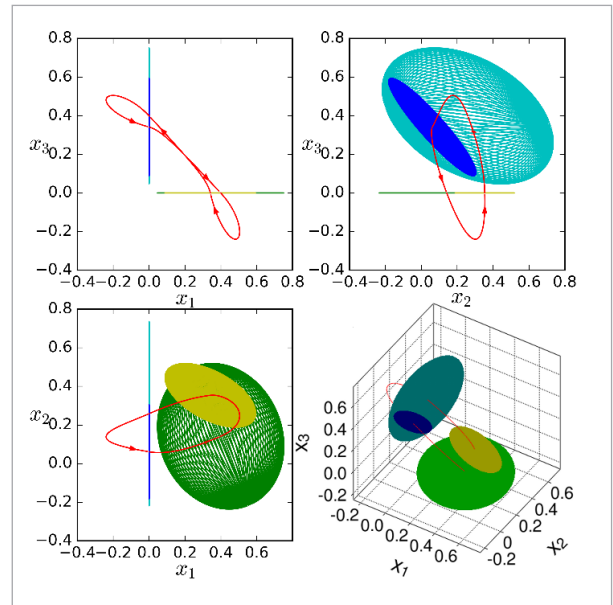
The resulting P_i matrices are

$$\begin{aligned} P_1 &= \begin{bmatrix} 129.2 & 111.22 & -13.53 \\ 111.22 & 118.41 & 7.66 \\ -13.53 & 7.66 & 54.55 \end{bmatrix} \\ P_2 &= \begin{bmatrix} 28.18 & 6.87 & -20.72 \\ 6.87 & 11.17 & -7.8 \\ -20.72 & -7.8 & 20.43 \end{bmatrix} \\ P_3 &= \begin{bmatrix} 54.55 & -7.66 & -13.53 \\ -7.66 & 118.41 & -111.22 \\ -13.53 & -111.22 & 129.2 \end{bmatrix} \\ P_4 &= \begin{bmatrix} 20.43 & 7.8 & -20.72 \\ 7.8 & 11.17 & -6.87 \\ -20.72 & -6.87 & 28.18 \end{bmatrix} \end{aligned} \quad (34)$$

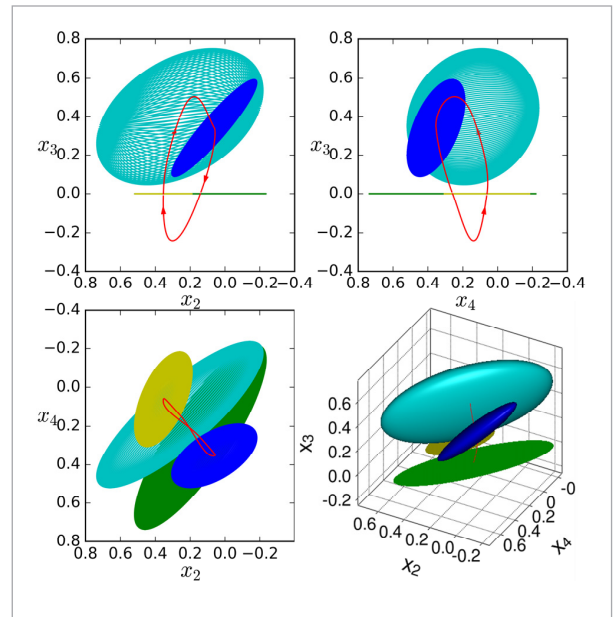
The result in \mathbb{R}^4 is difficult to illustrate directly; therefore, the result is shown as projections to different subspaces. In \mathbb{R}^4 , the region of stability determined by $P_1 \sim P_4$ consists of four third-order ellipsoids on the switching surfaces. Usually, their projections on \mathbb{R}^3 are second-order ellipses on the switching surfaces

(Figure 1(a)). However, the projections on the corresponding switching surface are still third-order ellipsoids (Figure 1(b)). Notably, in Figure 1 (b), the axes of x_2 and x_4 are inverted to clarify the result. The

Figure 1
Results of the sample system



(a) Projections on the x_1, x_2, x_3 subspace



(b) Projections on the x_2, x_3, x_4 subspace

system and the resulting region of stability are both symmetric; thus the projections on the x_1, x_2 subspace and x_3, x_4 subspace are similar. The projections not illustrated in Figure 1 could be inferred by symmetry.

Compared with the result of [4], the resulting guaranteed region of stability is much larger.

6. Conclusion

This paper introduces an improved algorithm for computing the regions of stability for limit cycles of PWA systems. With softer constraints developed, multiple feasible solutions are obtained by convex optimization. To overcome the shortcoming of convex optimization, the final solution is obtained by GA. With the proposed method, the resulting guaranteed regions of stability are larger than the results of existing methods. Given the proposed method, when designing a new PWA system, the stability conditions are easier to be satisfied, and the parameters of the system could be less conservative.

Appendix A

Proof of Proposition 1

When $x = \Pi\Delta$, we obtain

$$\begin{aligned} Cx &= R^T Q^T Q L_{n \times (n-1)} \Delta \\ &= R^T L_{n \times (n-1)} \Delta = \mathbf{0}_{1 \times (n-1)} \Delta = \mathbf{0} \end{aligned} \quad (35)$$

and

$$\Pi^T x = \Pi^T \Pi \Delta = L_{n \times (n-1)}^T Q^T Q L_{n \times (n-1)} \Delta = \Delta \quad (36)$$

When $Cx = \mathbf{0}$ and $\Pi^T x = \Delta$, we have

$$\begin{aligned} \begin{bmatrix} \mathbf{0} \\ \Delta \end{bmatrix} &= \begin{bmatrix} Cx \\ \Pi^T x \end{bmatrix} = \begin{bmatrix} R^T Q^T x \\ L_{n \times (n-1)}^T Q^T x \end{bmatrix} = \begin{bmatrix} R^T \\ L_{n \times (n-1)}^T \end{bmatrix} Q^T x \\ \Rightarrow x &= (Q^T)^{-1} \begin{bmatrix} R^T \\ L_{n \times (n-1)}^T \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{0} \\ \Delta \end{bmatrix}. \end{aligned} \quad (37)$$

Q is a unitary matrix; therefore,

$$(Q^T)^{-1} = Q. \quad (38)$$

Assume r is the first element of R . $C \neq \mathbf{0}_{1 \times n}$; hence, we have $r \neq 0$, and

$$\begin{bmatrix} R^T \\ L_{n \times (n-1)}^T \end{bmatrix}^{-1} = \begin{bmatrix} 1/r & \mathbf{0}_{1 \times (n-1)} \\ \mathbf{0}_{(n-1) \times 1} & I_{(n-1) \times (n-1)} \end{bmatrix}. \quad (39)$$

By substituting (41) and (42) into (40), we obtain

$$x = Q \begin{bmatrix} 1/r & \mathbf{0}_{1 \times (n-1)} \\ \mathbf{0}_{(n-1) \times 1} & I_{(n-1) \times (n-1)} \end{bmatrix} \begin{bmatrix} \mathbf{0} \\ \Delta \end{bmatrix} = Q L_{n \times (n-1)} \Delta = \Pi \Delta. \quad (40)$$

Proof of Theorem 2

Before proving Theorem 2, the following lemma is proved as a basis.

Lemma 1: Under Assumption 1 and Condition 1, the function $\tau_i(\Delta)$ is continuous for all $\Delta \in \Omega_i$.

Proof of Lemma 1. Assume $t_0 = \tau_i(\Delta_0)$. If the following condition holds, the function $\tau_i(\Delta)$ is continuous at $\Delta_0 \in \Omega_i$: For all $\epsilon > 0$, an η exists such that if $|\delta| < \eta$ and $(\Delta_0 + \delta) \in \Omega_i$, then $|\tau_i(\Delta_0 + \delta) - t_0| < \epsilon$.

Under Condition 1, the only switching surface to consider is S_{i+1} . For $t \in [0, t_0]$, the trajectory of $\Phi_i(x_i^* + \Delta_0, t)$ is at one side of S_{i+1} . Without loss of generality, assume that the trajectory is at the side where $C_{i+1} \Phi_i(x_i^* + \Delta_0, t) \geq d_{i+1}$. The continuity is proven if for all $\epsilon > 0$, an η exists such that if $|\delta| < \eta$, then the trajectory of $\Phi_i(x_i^* + \Delta_0 + \delta, t)$ does not reach S_{i+1} for all $t \in (0, t_0 - \epsilon]$, but reaches and passes S_{i+1} at some $t' \in (t_0 - \epsilon, t_0 + \epsilon)$. The proof consists of three parts: not reaching S_{i+1} for all $t \in (0, t_0 - \epsilon]$, reaching S_{i+1} at some $t' \in (t_0 - \epsilon, t_0 + \epsilon)$, and passing S_{i+1} at t' .

Part 1. Define the following helper function:

$$f(\delta) = \min \{ C_{i+1} \Phi_i(x_i^* + \Delta_0 + \delta, t) - d_{i+1} \mid t \in (0, t_0 - \epsilon] \} \quad (41)$$

It is continuous.

$f(\mathbf{0}) > 0$ can be proven. First, we assume that $f(\mathbf{0}) < 0$, then a $t_1 \in (0, t_0 - \epsilon]$ exists such that the trajectory of $\Phi_i(x_i^* + \Delta_0, t)$ passes S_{i+1} at t_1 . However, $\tau_i(\Delta_0) = t_0 > t_1$; hence, $f(\mathbf{0}) < 0$ does not hold. Second, we assume $f(\mathbf{0}) = 0$, because of the property of minimum, a $t_1 \in (0, t_0 - \epsilon]$ exists such that $C_{i+1} \dot{\Phi}_i(x_i^* + \Delta_0, t) = 0$. This relation contradicts Assumption 1. Thus, $f(\mathbf{0}) > 0$ holds.

Given the continuity of $f(\delta)$, an η_1 exists such that if $|\delta| < \eta_1$, then $f(\delta) > 0$. That is, the trajectory of

$\Phi_i(x_i^* + \Delta_0 + \delta, t)$ does not reach S_{i+1} for all $t \in (0, t_0 - \epsilon]$.

Part 2. Define

$$g(\delta, t) = C_{i+1}\Phi_i(x_i^* + \Delta_0 + \delta, t) - d_{i+1} \tag{42}$$

We have $g(\mathbf{0}, t_0) = 0$. Given Assumption 1 we also have

$$\frac{\partial g(\delta, t)}{\partial t}(\mathbf{0}, t_0) = C_{i+1}\dot{\Phi}_i(x_i^* + \Delta_0, t_0) - d_{i+1} \neq 0 \tag{43}$$

According to the implicit function theorem, the implicit function $\mathbb{R}^{n-1} \rightarrow \mathbb{R}$ determined by $g(\delta, t) = 0$ is continuous at $\delta = \mathbf{0}, t = t_0$. Thus, an η_2 exists such that if $|\delta| < \eta_2$, then $g(\delta, t) = 0$ holds for some $t' \in (t_0 - \epsilon, t_0 + \epsilon)$. That is, the trajectory of $\Phi_i(x_i^* + \Delta_0 + \delta, t)$ reaches S_{i+1} at some $t' \in (t_0 - \epsilon, t_0 + \epsilon)$.

Part 3. Define

$$h(\delta, t) = C_{i+1}\dot{\Phi}_i(x_i^* + \Delta_0 + \delta, t) \tag{44}$$

According to Assumption 1, $h(\mathbf{0}, t_0) \neq 0$. Moreover, the trajectory of $\Phi_i(x_i^* + \Delta_0, t)$ passes S_i at t_0 ; therefore, we have $h(\mathbf{0}, t_0) > 0$. Given the continuity of $h(\delta, t)$, an $\eta_3 > 0$ exists such that if $|\delta| + |t - t_0| < \eta_3$, then $h(\delta, t) > 0$.

At this point, given the results of Parts 1 and 2, for $\epsilon' = \eta_3/2$, η_1' and η_2' exist such that if $|\delta| < \eta_1'$ and $|\delta| < \eta_2'$, then the trajectory of $\Phi_i(x_i^* + \Delta_0 + \delta, t)$ does not reach S_i for all $t \in (t_0, t_0 - \epsilon']$ but reaches S_i at some $t_\delta \in (t_0 - \epsilon', t_0 + \epsilon')$. When the trajectory reaches S_i , due to the result of Step 3, if $|\delta| < \eta_3 - \epsilon' = \eta_3/2$, then $C_{i+1}\dot{\Phi}_i(x_i^* + \Delta_0 + \delta, t) > 0$. That is, the corresponding trajectory passes S_i .

In conclusion, $|\delta| < \eta = \min\{\eta_1, \eta_2, \eta_3/2, \eta_1', \eta_2'\}$ is eligible for proving the continuity of $\tau_i(\Delta)$ at Δ_0 .

Proof of Theorem 2. Assume $t_1 = \tau_i(\Delta_1)$. Theorem 2 is proven if the following two statements hold:

(a) If $0 < t_1 < t_{i-}$, then

$$\|\Delta_1\|_{P_i} > \min\{\Gamma_{P_i}(\Lambda_{i,0,t_{i-}}), \Gamma_{P_i}(\Lambda_{i,0,t_{i+}})\}$$

(b) If $t_1 > t_{i+}$, then

$$\|\Delta_1\|_{P_i} > \min\{\Gamma_{P_i}(\Lambda_{i,0,t_{i-}}), \Gamma_{P_i}(\Lambda_{i,0,t_{i+}})\}$$

Only Statement (a) is proven herein. The proof for Statement (b) is similar and thus omitted.

On the basis of the definition of the impact map, we have $\tau_i(\mathbf{0}) = \tau_i^*$. On the line segment from $\mathbf{0}$ to Δ_1 , all points can be written as $\alpha\Delta_1$, where $\alpha \in [0, 1]$. Define

$$l(\alpha) = \tau_i(\alpha\Delta) \tag{45}$$

Given Assumption 1, $l(\alpha)$ is continuous at $\alpha = 0$. About $l(\alpha)$'s continuity, there are two cases. First, $l(\alpha)$ is continuous on $[0, 1]$. Second, a $\alpha_0 \in (0, 1]$ exists such that $l(\alpha)$ is continuous on $[0, \alpha_0)$, and discontinuous at α_0 .

First, if $l(\alpha)$ is continuous on $[0, 1]$, given that $l(0) = t_i^* > t_{i-}$, $l(1) < t_{i-}$, an $\alpha_1 \in (0, 1)$ exists such that $l(\alpha_1) = t_{i-}$. For such an α_1 , we have

$$\|\Delta_1\| > \alpha_1 \|\Delta_1\|_{P_i} = \|\alpha_1\Delta_1\|_{P_i} \geq \Gamma_{P_i}(\Lambda_{i,0,t_{i-}}) \tag{46}$$

Second, if $l(\alpha)$ is continuous on $(0, \alpha_0)$ and discontinuous at α_0 , given Lemma 1, we have

$$l(\alpha_0) \notin [t_{i-}, t_{i+}] \tag{47}$$

Otherwise, $l(\alpha)$ would be continuous at α_0 . Considering (50), $l(\alpha_0) < t_{i-}$ or $l(\alpha_0) > t_{i+}$. $l(0) = t_i^*$; hence, an $\alpha_1 \in (0, \alpha_0)$ exists such that $l(\alpha_1) = t_{i-}$ or $l(\alpha_1) = t_{i+}$.

By arguments similar to (49), if $l(\alpha_1) = t_{i-}$, then

$$\|\Delta_1\| > \Gamma_{P_i}(\Lambda_{i,0,t_{i-}}) \tag{48}$$

If $l(\alpha_1) = t_{i+}$, then

$$\|\Delta_1\| > \Gamma_{P_i}(\Lambda_{i,0,t_{i+}}) \tag{49}$$

In conclusion, Statement (a) holds.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 61573074, 51477020 and by the National High-Tech R & D Program of China under Grant 2015AA010402.

References

1. Hajiahmadi, M., De Schutter, B., Hellendoorn, H. Design of Stabilizing Switching Laws for Mixed Switched Affine Systems. *IEEE Transactions on Automatic Control*, 2015, PP(99), 1-1.
2. Hetel, L., Bernuau, E. Local Stabilization of Switched Affine Systems. *IEEE Transactions on Automatic Control*, 2015, 60(4), 1158-1163. <https://doi.org/10.1109/TAC.2014.2350211>

3. Goncalves, J. M., Megretski, A., Dahleh, M. A. Global Analysis of Piecewise Linear Systems Using Impact Maps and Surface Lyapunov Functions. *IEEE Transactions on Automatic Control*, 2003, 48(12), 2089-2106. <https://doi.org/10.1109/TAC.2003.820061>
4. Goncalves, J. M., Regions of Stability for Limit Cycle Oscillations in Piecewise Linear Systems. *IEEE Transactions on Automatic Control*, 2005, 50(11), 1877-1882. <https://doi.org/10.1109/TAC.2005.858674>
5. Gonçaves, J. M. Regions of Stability for Limit Cycles of Piecewise Linear Systems. In *Proceedings of the 42nd IEEE Conference on Decision and Control*, Maui, Hawaii USA, December, 2003, 1, 651-656. <https://doi.org/10.1109/CDC.2003.1272638>
6. Gonçaves, J. M. *Constructive Global Analysis of Hybrid Systems*. Massachusetts Institute of Technology, 2000.
7. Lin, C., Wang, Q.-G., Lee, T. H. Existence Analysis for Limit Cycles of Relay Feedback Systems. *Asian Journal of Control*, 2004, 6(3), 428-431. <https://doi.org/10.1111/j.1934-6093.2004.tb00219.x>
8. Manchester, I. R. Transverse Dynamics and Regions of Stability for Nonlinear Hybrid Limit Cycles. *ArXiv Prepr. ArXiv10102241*, 2010.
9. Peterchev, A. V., Sanders, S. R. Quantization Resolution and Limit Cycling in Digitally Controlled PWM Converters. In *Power Electronics Specialists Conference, 2001. PESC. 2001 IEEE 32nd Annual*, 2001, 2(2), 465-471. <https://doi.org/10.1109/PESC.2001.954158>
10. Sun, Y., Tang, C., Hu, A. P., Li, H. L., Nguang, S. K. Multiple Soft-Switching Operating Points-Based Power Flow Control of Contactless Power Transfer Systems. *IET Power Electronics*, 2011, 4(6), 725-731. <https://doi.org/10.1049/iet-pel.2009.0318>
11. Tang, J. Z., Manchester, I. R. Transverse Contraction Criteria for Stability of Nonlinear Hybrid Limit Cycles. In *2014 IEEE 53rd Annual Conference on Decision and Control (CDC)*, 2014, 31-36. <https://doi.org/10.1109/CDC.2014.7039355>
12. Tang, C. S., Sun, Y., Su, Y. G., Nguang, S. K., Hu, A. P. Determining Multiple Steady-State ZCS Operating Points of a Switch-Mode Contactless Power Transfer System. *IEEE Transactions on Power Electronics*, 2009, 24(2), 416-425. <https://doi.org/10.1109/TPEL.2008.2007642>

Summary / Santrauka

This paper proposes an improved algorithm to compute the regions of stability for limit cycles of piecewise affine systems. Instead of using convex optimization algorithms, such as solving linear matrix inequalities and sum-of-square programming, the genetic algorithm (GA) is used to obtain the final results. With the help of GA, both the constraints and objective functions can be nonconvex. As a result, larger guaranteed regions of stability are achieved. On the basis of the impact map and Lyapunov stability theory, the stability conditions are analyzed. Algorithm-friendly criteria, both convex and nonconvex, are developed. Randomly generated solutions are usually infeasible; hence, we generate the initial GA population by convex optimization. To improve the initial population's diversity, we use multiple convex objective functions to produce different initial solutions. Other application-specific parts of GA, such as computing the fitness of solutions, are also introduced in detail. A sample system is analyzed to illustrate the effectiveness of the proposed method.

Straipsnyje pateikiamas patobulintas algoritmas, skirtas hibridinių afininių sistemų ribinių ciklų stabilumo regionams apskaičiuoti. Galutiniams rezultatams išgauti vietoj tokių išgaubto optimizavimo algoritmų, kaip tiesinių matricių nelygybių sprendimas ar kvadratų sumos programavimas, pasitelkiamas genetinis algoritmas (GA). Pastarojo algoritmo pagalba tiek apribojimai, tiek tikslo funkcijos gali būti neišgaubtos. Dėl to galima garantuotai pasiekti didesnius stabilumo regionus. Stabilumo sąlygos straipsnyje analizuojamos remiantis poveikio žemėlapiu ir Liapunovo stabilumo teorija. Sukurti algoritmams palankūs tiek išgaubti, tiek neišgaubti kriterijai. Atsitiktinai sugeneruoti sprendimai įprastai yra neįmanomi, taigi, autoriai generuoja pradinę GA populiaciją optimizuodami išgaubtumą. Siekiant pagerinti pradinę populiacijos įvairovę, autoriai naudoja keletą išgaubtų tikslo funkcijų skirtingiems pradiniais sprendimams išgauti. Detaliai pristatomos ir kitos specifinės GA taikymo detalės, tokios, kaip sprendimų tinkamumo skaičiavimas. Siūlomo metodo veiksmingumas iliustruojamas analizuojant pavyzdinę sistemą.