Reachability Analysis of Continuous-Time Piecewise Affine Systems

Abdullah Hamadeh, Jorge Goncalves

Department of Engineering, University of Cambridge, Cambridge CB2 1PZ, UK

Abstract

This paper proposes an algorithm for the characterization of reachable sets of states for continuous-time piecewise affine systems. Given a model of the system and a bounded set of possible initial states, the algorithm employs a linear matrix inequality approach to compute both upper and lower bounds on reachable regions. Rather than performing computations in the state-space, this method uses impact maps to find the reachable sets on the switching surfaces of the system. This tool can then be used to deduce safety and performance results about the system.

Key words: Reachability; Switched Systems.

1 Introduction

Hybrid systems are a class of dynamical systems that feature multiple modes of operation. The dynamics of the system obey a particular set of differential (or difference) equations, depending on which mode the system is in. Often, there exist a set of state-dependent rules for switching between these modes, although the transition may also be event- or time-driven. This paper addresses the reachability problem in a particular type of hybrid system termed continuous-time piecewise affine systems (cPWA). The distinguishing feature of cPWA is that the differential equations in each mode are affine.

The question of the reachability of a hybrid system is of particular interest to the verification engineer seeking to ensure that the system trajectories satisfy certain properties. In addition to yielding information regarding the stability and performance of the system, reachability results can be used to verify whether a system's trajectories remain outside unsafe regions of the state-space.

Although there are several variants of the reachability problem, it essentially involves identifying the regions of the state space that trajectories of the system can reach given an uncertainty in the system, in a finite amount of time. The uncertainty could lie in the initial state, the input, the system dynamics or the switching rules.

This study concerns autonomous cPWA whose discrete modes of operation are state-dependent. We assume that the various regions of the state-space that are associated with particular modes are separated from each other by hyperplanes (termed switching surfaces). Furthermore, we assume that there exists a bounded set of possible initial states, possibly representing an uncertainty in the system. Therefore, a brief problem description is, given a hybrid system and such a bounded set of possible initial states, what bound can one place on the reachable set after a finite amount of time?

Previous methods of computing reach sets were based on face lifting techniques [2],[5], whereby vertices of a polyhedral initial set are expanded at incremental periods of time r in the direction of the system's flow, perpendicularly to the edges of the initial set. The tool d/dt uses this procedure and stores the reach set as a union of orthogonal polyhedra. For linear systems of low dimension this tool keeps the over-approximation error of order $\mathcal{O}(r^2)$ [1]. The tool Checkmate [8] maps the vertices of a polyhedral set to their successors at fixed increments of time into the future and then over-approximates the convex hull of these vertices. Since this method uses unions of convex polyhedral sets, the method is difficult to use in high dimensional cases. In other methods [3],[9], ellipsoids have been used to approximate reach sets. The complication with using ellipsoidal reach sets in the state space is that their unions are non-convex.

In [6] a new approach was introduced that globally analyzed stability in cPWA. This method consisted of finding Lyapunov functions on the switching surfaces to prove that Poincaré-type maps associated with the system were contracting. These generalized Poincaré maps, or *impact maps*, are defined from one switching surface to another. This work introduced a technique that involved expressing the impact map as a linear transformation parameterized by the switching time, that is, the time for a trajectory to cross from one switching surface to another. This led to the ability to nu-

Email addresses: aoh21@cam.ac.uk (Abdullah Hamadeh), jmg77@cam.ac.uk (Jorge Goncalves).

merically solve sets of linear matrix inequalities (LMIs) to find surface Lyapunov functions for the system.

The reachability analysis method proposed herein takes an approach based on the tools in [6]. This is a different approach from those employed in most previous studies in that the core reachability computations take place on the switching surfaces rather than in the state space. In other words, the algorithm begins by identifying an ellipsoidal set of initial states on one switching surface, called the departure switching surface. Each point within this set will map onto a subsequent hyperplane (the arrival switching surface) to form a reachable set *on this latter switching surface* that is generally not convex. LMIs are then used to find two ellipsoids on the arrival switching surface, one of which is an over-approximation of the reach set and the other an under-approximation. These steps are repeated taking the bounds on the reach set as the new initial sets and and using them to compute the reachable sets on the next switching surface. By the end of the algorithm, after a certain number of switches, the reach sets will form a series of upper bound and lower bound ellipsoidal subsets of the switching surfaces indicating what states the trajectories of the system can autonomously reach given the set of possible initial states. The upper bounds represent limits on the states that the system cannot autonomously reach beyond given the initial set. The lower bounds indicate subsets of the switching surfaces that are definitely reachable from the initial set.

The novelty of this method is its use of LMIs in computing reach sets, thus significantly reducing computation times. It also allows us to analyze cPWA models of high dimensionality, a feature which would have been computationally intractable using previous methods. The expensive computations of the reach sets in the state space are avoided by finding the image of the reach set on the switching surfaces of the system in terms of ellipsoids. To verify the safety of the system we then express the unsafe states in terms of new hyperplanes and verify that the upper bound ellipsoids do not reach these switching surfaces.

This paper begins with a description of cPWA and impact maps. This is followed by a section detailing the problem and tools for computing the upper and lower bound estimates on reach sets. We then present an algorithm that incorporates the tools in the previous section to find the reach sets after a finite number of switches. Following this, we present some technical notes on improving the results of the algorithm. Finally, we conclude with a discussion of the results.

2 Framework

In this section, we begin by describing the framework of the cPWA that will be analyzed in this paper. This approach builds on the tools developed in [6].

2.1 Piecewise Affine Systems

The autonomous, continuous-time, n-dimensional piecewise affine system $\mathcal{H}$ takes the form $\mathcal{H} = [Q, \mathbf{\Sigma}, \mathcal{J}, \mathbf{S}]$. The set $Q = \{1, \cdots, N\}$ is the collection of indices q denoting the discrete mode of the system. The set $\mathbf{\Sigma} = \{\Sigma_q\}_{q \in Q}$ is a set of affine dynamical systems. When the system is in a particular mode q, the active dynamical system is Σ_q. The system Σ_q has the time-dependent, continuous state vector $x(\tau) \in \mathbb{R}^n$ which is the solution to the affine differential equation

$$\dot{x} = A_q x + B_q, \qquad q \in Q \tag{1}$$

at time τ, with initial state $x(0)$. Here, $A_q \in \mathbb{R}^{n \times n}$, $B_q \in \mathbb{R}^n$. We place no restrictions on the eigenvalues of A_q except that they are non-zero, and hence A_q is invertible.

The set $\mathbf{S} = \{S_j\}_{j \in \mathcal{J}}$ is a set of hyperplanes (or, switching surfaces), indexed by $j \in \mathcal{J}$, $\mathcal{J} = \{1, \cdots, M\}$. These hyperplanes divide the state space X into closed polyhedral subsets X_q. In each region X_q the dynamics of the system are given by its respective equation (1). The subsets X_q are polyhedral, with limit points given by the switching surfaces S_j. Since the subsets X_q are closed, the state space X is such that $X = \bigcup_{q \in Q}\{X_q\}$. Which discrete mode q is active at a particular instant of time τ depends on which subset X_q of the state space the trajectory of x lies in at time τ.

Definition 1 *The switching surface S_j, $j \in \mathcal{J}$ is defined as the hyperplane of states x such that $S_j = \{x \in \mathbb{R}^n | C_j x = d_j\}$, where $C_j \in \mathbb{R}^{1 \times n}$ and $d_j \in \mathbb{R}$.*

Assuming no sliding modes exist, then if a trajectory $x(\tau)$ reaches a switching surface it will either cross it into a new mode q or remain in its current mode depending on the direction of the vector field given by (1). Letting t_s be the *switching time* at which the trajectory $x(\tau)$ reaches a switching surface, we impose the constraint that the trajectory is continuous at the switching time t_s:

$$\lim_{\tau \to t_s^-} x(\tau) = \lim_{\tau \to t_s^+} x(\tau) = x(t_s)$$

2.2 Impact Maps

Consider an autonomous cPWA which has a discrete mode q, active when the current state $x \in X_q$. In this mode the continuous-time dynamics are given by the differential equation (1). Let $U_j \subset S_j$ be a nonempty set of states such that any trajectory with an initial condition in U_j will next switch at the switching surface S_{j+1}. Therefore, states $x \in U_j$ are such that $x(0) \in U_j$, $x(t_s) \in S_{j+1}$ and $x(\tau) \in X_q$ for $\tau \in (0, t_s)$ is the solution to the differential equation (1) with initial state $x(0)$.

Definition 2 *Define as $\mathcal{T}(U_j)$ the set of switching times t_s of trajectories with initial states in set $U_j \subset S_j$, all of which next switch at switching surface S_{j+1}.*

Now consider two general states $x_k \in U_j$ and $x_{k+1} \in S_{j+1}$. By fixing a state $x_k^* \in U_j$ we can define any point $x_k \in U_j$ as $x_k = x_k^* + \Delta_j(x_k^*)$ where $\Delta_j(x_k^*) \in U_j - x_k^*$. After finite switching times the trajectories emanating from x_k, x_k^* will reach states $x_{k+1}, x_{k+1}^* \in S_{j+1}$ respectively, and so we can rewrite x_{k+1} as $x_{k+1} = x_{k+1}^* + \Delta_{j+1}(x_{k+1}^*)$, with $\Delta_{j+1}(x_{k+1}^*) \in S_{j+1} - x_{k+1}^*$. This is illustrated in Figure 1.

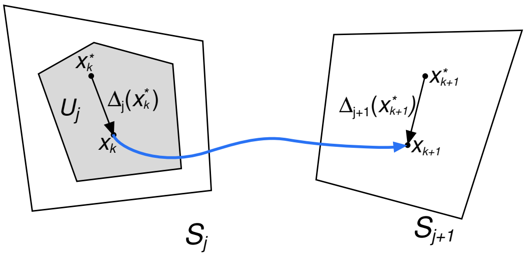


Figure 1. Points in U_j next switch at $U_{j+1} \subset S_{j+1}$.

Definition 3 $\Pi_j \in \mathbb{R}^{n \times n-1}$ *is the matrix of column vectors in the orthogonal complement of* C_j*, forming an orthonormal basis on the switching surface* S_j*.*

Given the states $x_k, x_k^* \in U_j$, the vector $\Delta_j(x_k^*) = x_k - x_k^* \in U_j$ can be written, for any x_k, as

$$\Delta_j(x_k^*) = \Pi_j \delta_j(x_k^*) \tag{2}$$

where $\delta_j(x_k^*) \in \mathbb{R}^{n-1}$. Since $\Pi_j^T \Pi_j = I_{n-1}$, the $n-1 \times n-1$ identity matrix, we then have:

$$\delta_j(x_k^*) = \Pi_j^T (x_k - x_k^*) \tag{3}$$

Using (3) we define the vectors $\delta_j(x_k^*) = \Pi_j^T(x_k - x_k^*)$ and $\delta_{j+1}(x_{k+1}^*) = \Pi_{j+1}^T(x_{k+1} - x_{k+1}^*)$ which respectively lie in S_j and S_{j+1}. The mapping from the vector $\delta_j(x_k^*)$ to $\delta_{j+1}(x_{k+1}^*)$ is given by a generally nonlinear map, called the *impact map*, which is defined in [6]:

Definition 4 (Gonçalves et al.) *The impact map is the matrix* $\bar{H}_{j,k}(\tau) \in \mathbb{R}^{n-1 \times n-1}$ *given by*

$$\bar{H}_{j,k}(\tau) = \Pi_{j+1}^T \left(I_n - \frac{(x_k^*(\tau) - x_{k+1}^*)C_{j+1}}{C_{j+1}(x_k^*(\tau) - x_{k+1}^*)} \right) e^{A_q \tau} \Pi_j \tag{4}$$

where $x_k^*(\tau)$ *is the development with time of a trajectory with initial state* x_k^* *and dynamics given by (1). This map is such that*

$$\delta_{j+1}(x_{k+1}^*) = \bar{H}_{j,k}(t_s)\delta_j(x_k^*) \tag{5}$$

where t_s *is the switching time associated with a trajectory obeying (1) with initial state* x_k^*.

Following a derivation completely analogous to that of $\bar{H}_{j,k}(\tau)$ in [6], it is straightforward to construct a 'reverse' impact map, $\bar{J}_{j,k}(\tau)$ which maps $\delta_j(x_k^*)$ back onto $\delta_{j+1}(x_{k+1}^*)$.

Definition 5 *The reverse impact map is the matrix* $\bar{J}_{j,k}(\tau) \in \mathbb{R}^{n-1 \times n-1}$ *given by*

$$\bar{J}_{j,k}(\tau) = \Pi_j^T \left(I_n - \frac{(x_{k+1}^*(\tau) - x_k^*)C_j}{C_j(x_{k+1}^*(\tau) - x_k^*)} \right) e^{-A_q \tau} \Pi_{j+1} \tag{6}$$

This map is such that

$$\delta_j(x_k^*) = \bar{J}_{j,k}(t_s)\delta_{j+1}(x_{k+1}^*) \tag{7}$$

where t_s *is the switching time associated with a trajectory obeying (1) with initial state* x_k^*.

3 Reach Set Computations

3.1 Problem Formulation

In a system with unsafe states, ensuring emptiness of the the intersection of such regions with an over-approximation (or, upper bound) of the reach set would imply safety. Similarly, if there are regions of the state space where one would like the system to reach, results on an under-approximation (or, lower bound) on the reach set can be used to measure the performance of the system. In this section we show how such bounds may be computed.

First, consider an initial ellipsoidal set that is a subset of U_j. The impact map $\bar{H}_{j,k}(\tau)$ maps points within this set onto S_{j+1}. The smallest upper bound is the smallest ellipsoidal set on S_{j+1} which contains all the points $\delta_{j+1}(x_{k+1}^*) = \bar{H}_{j,k}(t_s)\delta_j(x_k^*)$ for all switching times t_s in that ellipsoidal set. It may also contain points that are not reachable from the initial set on S_j. This idea is illustrated in Figure 2.

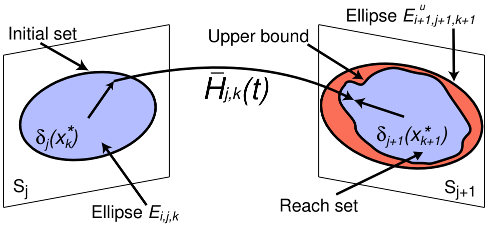


Figure 2. Initial set, exact reach set and upper bound.

The reverse map $\bar{J}_{j,k}(\tau)$ maps points in the reach set back onto the initial set. Define the largest lower bound on the reach set to be the largest subset of the actual reach set such that each point in this lower bound is mapped by $\bar{J}_{j,k}(\tau)$ back onto a point in the initial set on S_j. The lower bound may not cover the entire reach set, but it contains points that can definitely be reached from the initial set. This is illustrated in Figure 3.

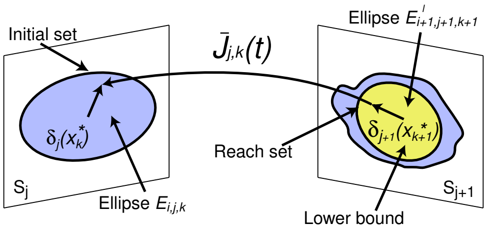


Figure 3. Initial set, exact reach set and lower bound.

The remainder of this section shows how to compute these upper and lower bounds. Prior to that, however, we first give the definition of the standard ellipsoidal set on a switching surface that will be used to characterize initial sets and reach sets.

Definition 6 *The set $E_{i,j,k} = E_{i,j,k}(P_i, S_j, x_k^*) \subset S_j$, where $P_i > 0$, $P_i \in \mathbb{R}^{n-1 \times n-1}$ and $x_k^* \in S_j$ is defined as $E_{i,j,k}(P_i, S_j, x_k^*) = \{x_k \in S_j | \delta_j(x_k^*)^T P_i \delta_j(x_k^*) \leq 1\}$.*

The initial set of states is taken to be an ellipsoidal set $E_{i,j,k} \subset U_j$, as per Definition 6. The upper bound and lower bound on the reach set are respectively over-approximations and under-approximations on the reach set, both also expressed as ellipsoidal subsets of S_{j+1} as per Definition 6.

3.2 Upper Bound Computation

Before defining the upper bound of a reach set we make the following assumption which will be relaxed in the sequel.

Assumption 7 *Given a set $E_{i,j,k} \subset U_j$ of initial states on the departure switching surface S_j, we assume that all trajectories with initial states $x_k(0) \in E_{i,j,k}$ next switch at the same switching surface S_{j+1}. Each trajectory switches at S_{j+1} after a finite switching time in the set $\mathcal{T}(E_{i,j,k})$.*

Definition 8 [Upper Bound Reach Set] *Given a set $E_{i,j,k} = E_{i,j,k}(P_i, U_j, x_k^*)$ of initial states on the departure switching surface S_j and under Assumption 7, an upper bound on the reach set is defined as the set $E^u_{i+1,j+1,k+1} = E_{i+1,j+1,k+1}(P^u_{i+1}, U_{j+1}, x^*_{k+1})$ which is such that if $x(0) \in E_{i,j,k}$ then $x(t_s) \in E^u_{i+1,j+1,k+1}$ for some $t_s \in \mathcal{T}(E_{i,j,k})$.*

Theorem 9 *Under Assumption 7, given a set $E_{i,j,k} \subset U_j$ of initial states on the departure switching surface S_j as defined in Definition 6, an upper bound on the reach set (in the sense of Definition 8) for this set of initial states is given by the set $E^u_{i+1,j+1,k+1} \subset S_{j+1}$ where*

$$P_i - \bar{H}_{j,k}(\tau)^T P_{i+1} \bar{H}_{j,k}(\tau) \geq 0, \quad \forall \tau \in \mathcal{T}(E_{i,j,k}) \tag{8}$$

Proof From Definition 6, states x_k within the initial set $E_{i,j,k} \subset U_j$, which are represented on the switching surface S_j by the vector $\delta_j(x_k^*)$, are such that:

$$F_i^u = 1 - \delta_j(x_k^*)^T P_i \delta_j(x_k^*) \geq 0 \tag{9}$$

The set $E_{i,j,k}$ is parameterized by the state $x_k^* \in U_j$ and the trajectory with initial state x_k^* switches at state $x^*_{k+1} \in S_{j+1}$. States x_{k+1} in the upper bound reach set $E^u_{i+1,j+1,k+1}$, represented on the switching surface S_{j+1} by the vector $\delta_{j+1}(x^*_{k+1})$, are such that:

$$F^u_{i+1} = 1 - \delta_{j+1}(x^*_{k+1})^T P^u_{i+1} \delta_{j+1}(x^*_{k+1}) \geq 0 \tag{10}$$

By Assumption 7 and Definition 8, for $E^u_{i+1,j+1,k+1}$ to be an upper bound (9) must imply (10), since any state x_k lying in the start set $E_{i,j,k}$ must have its image x_{k+1} in the upper bound on the reach set. Applying the S-procedure [4] gives a single relation that says that non-negativity of F_i^u implies non-negativity of F^u_{i+1} when ϵ_u is positive:

$$F^u_{i+1} - \epsilon_u F_i^u \geq 0 \tag{11}$$

Now as $\delta_j(x_k^*) \to 0$, (5) implies that $\delta_{j+1}(x^*_{k+1}) \to 0$. Furthermore, $F_i \to 1$ and $F_{i+1} \to 1$, which implies that $\epsilon_u \leq 1$. Since $F_i^u, F^u_{i+1} \geq 0$, setting $\epsilon_u = 1$ (the supremum over its allowable range) gives the tightest condition on (11). Inequality (11) now becomes

$$\delta_j(x_k^*)^T P_i \delta_j(x_k^*) - \delta_{j+1}(x^*_{k+1})^T P_{i+1} \delta_{j+1}(x^*_{k+1}) \geq 0$$

Substituting (5) into the above inequality then yields the series of LMIs, parameterized by the switching times $\tau \in \mathcal{T}(E_{i,j,k})$

$$P_i - \bar{H}_{j,k}(\tau)^T P^u_{i+1} \bar{H}_{j,k}(\tau) \geq 0, \quad \forall \tau \in \mathcal{T}(E_{i,j,k}) \tag{12}$$

We therefore need to solve for P^u_{i+1} to obtain the upper bound $E^u_{i+1,j+1,k+1}$. ■

What remains is to optimize the upper bound so that it is as ‘small’ as possible in some sense. Maximizing the trace of the matrix P^u_{i+1} is one convex optimization that is linear in the elements of P^u_{i+1} that could be performed to do this.

So far the functions used to approximate initial and reach sets are quadratic forms. It is also possible to use quadratic functions and higher order polynomial sets as bounds on start and reach sets, giving even less conservative results. Using the techniques in [10], higher order polynomials can be recast as a sum of squares of polynomials, and these can then be used to form LMIs similar to those described above.

3.3 Lower Bound Computation

Definition 10 [Lower Bound Reach Set] *Given a set $E_{i,j,k} \subset U_j$ of initial states on the departure switching surface S_j and under Assumption 7, a lower bound on the reach set is defined as the set $E^l_{i+1,j+1,k+1} = E_{i+1,j+1,k+1}(P^l_{i+1}, U_{j+1}, x^*_{k+1})$ which is such that if $x(t_s) \in E^l_{i+1,j+1,k+1}$ for some $t_s \in \mathcal{T}(E_{i,j,k})$ then $x(0) \in E_{i,j,k}$.*

With this definition, a point in ellipsoid $E^l_{i+1,j+1,k+1}$ can be reached from a point in $E_{i,j,k}$, though $E_{i,j,k}$ will also contain points that can reach beyond the limits of $E^l_{i+1,j+1,k+1}$.

Theorem 11 *Given a set $E_{i,j,k} \subset U_j$ of initial states on the departure switching surface S_j as defined in Definition 6, a lower bound on the reach set (in the sense of Definition 10) for this set of initial states is given by the set $E^l_{i+1,j+1,k+1} \subset S_{j+1}$ where*

$$P_{i+1} - \bar{J}_{j,k}(\tau)^T P_i \bar{J}_{j,k}(\tau) \geq 0, \quad \forall \tau \in \mathcal{T}(E_{i,j,k}) \tag{13}$$

Proof From Definition 6, states x_{k+1} within the lower bound on the reach set $E^l_{i+1,j+1,k+1} \subset U_{j+1}$, which are represented on the switching surface S_{j+1} by the vector $\delta_{j+1}(x^*_{k+1})$, are such that:

$$F^l_i = 1 - \delta_{j+1}(x^*_{k+1})^T P^l_{i+1} \delta_{j+1}(x^*_{k+1}) \geq 0 \tag{14}$$

States x_k in the initial set $E_{i,j,k}$, represented on the switching surface S_j by the vector $\delta_j(x^*_k)$, are such that:

$$F^l_{i+1} = 1 - \delta_j(x^*_k)^T P_i \delta_j(x^*_k) \geq 0 \tag{15}$$

Under Assumption 7 and Definition 10, for $E^l_{i+1,j+1,k+1}$ to be a lower bound (14) must imply (15), since any state x_{k+1} lying in the lower bound must be the image of a state x_k in the initial set. Applying the S-procedure as in the proof of Theorem 9 gives a single relation that says that non-negativity of F^l_{i+1} implies non-negativity of F_i when ϵ_l is positive:

$$F^l_i - \epsilon_l F^l_{i+1} \geq 0 \tag{16}$$

As $\delta_{j+1}(x^*_{k+1}) \rightarrow 0$, (7) implies that $\delta_{j+1}(x^*_{k+1}) \rightarrow 0$. Furthermore, $F^l_{i+1} \rightarrow 1$ and $F^l_{i+1} \rightarrow 1$, which implies that $\epsilon_l \leq 1$. Since $F^l_i, F^l_{i+1} \geq 0$, setting $\epsilon_l = 1$ (the supremum over its allowable range) gives the tightest condition on (16). Inequality (16) now becomes

$$\delta_{j+1}(x^*_{k+1})^T P^l_{i+1} \delta_{j+1}(x^*_{k+1}) - \delta_j(x^k_*)^T P_i \delta_j(x^*_k) \geq 0$$

Substituting (7) into the above inequality then yields the series of LMIs

$$P^l_{i+1} - \bar{J}_{j,k}(\tau)^T P_i \bar{J}_{j,k}(\tau) \geq 0, \quad \forall \tau \in \mathcal{T}(E_{i,j,k}) \tag{17}$$

We therefore need to solve for P^l_{i+1} to obtain the upper bound $E^l_{i+1,j+1,k+1}$. ■

Minimizing the trace of the matrix P^l_{i+1} is a convex optimization that can be performed to maximize the size of the lower bound.

Remark 12 *It would be sufficient to have the series of LMIs (17) hold true for the range of switching times $\mathcal{T}(E^l_{i+1,j+1,k+1})$, but knowing this range would require previous knowledge of P^l_{i+1}. However, since $\mathcal{T}(E^l_{i+1,j+1,k+1}) \subset \mathcal{T}(E_{i,j,k})$, if the LMIs hold true for the latter range of switching times, they necessarily hold true for the former.*

3.4 Bounds On Switching Times

The LMIs (8) and (13) are both parameterized by the switching time of the trajectories in their respective initial sets. To solve these LMIs for the matrices P^u_{i+1} and P^l_{i+1}, we need to have a bound on this range of times. In [6] we saw that the set of points on S_j having the same switching time is always a convex subset of a linear manifold of dimension $n-2$. This follows from the fact that any point on S_j must satisfy two linear equations on $\Delta_j(x^*_k)$ (see the relevant paper for more details). This idea is illustrated in Figure 4.

Therefore, given the initial set $E_{i,j,k}$, finding the subsets of states with the same switching time that are tangent to this ellipsoid yields the set of switching times $\mathcal{T}(E_{i,j,k})$ of points within this ellipsoid.

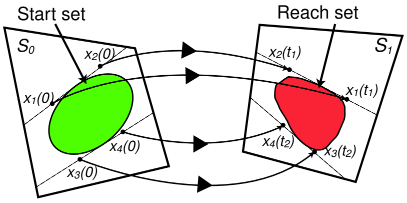


Figure 4. Lines are subsets of the hyperplanes containing states with the same switching time. The range of switching times $\mathcal{T}_k$ is $[t_1,t_2]$.

4 Implementation

Using Theorems 9 and 11 we can now propose an algorithm that systematically finds the upper and lower bound reach sets for multiple switches, terminating after a finite amount of time, a finite number of switches or after a certain switching surface is reached. We assume that the initial set is given as an elliptical set $E_{i,j,k} \subset S_j$.

Algorithm 1 (Computing Reach Sets) *Initialize with set $E_{i,j,k} \subset S_j$*

Step 1 *Find range of switching times $\mathcal{T}(E_{i,j,k})$ to switching surface S_{j+1} for points in $E_{i,j,k}$.*

Step 2 *Find upper bound ellipsoid $E^u_{i+1,j+1,k+1}$ by solving (8) for $\tau \in \mathcal{T}(E_{i,j,k})$. Optimal upper bound is found by maximizing the trace of P^u_{i+1}.*

Step 3 *Find lower bound ellipsoid $E^l_{i+1,j+1,k+1}$ by solving (13) for $\tau \in \mathcal{T}(E_{i,j,k})$. Optimal lower bound is found by minimizing the trace of P^l_{i+1}.*

Step 4 *To find the next upper bound $E^u_{i+2,j+2,k+2}$ repeat steps 1 and 2 using as the new initial set the previous upper bound $E^u_{i+1,j+1,k+1}$.*

Step 5 *To find the next lower bound $E^l_{i+2,j+2,k+2}$ repeat steps 1 and 3 using as the new initial set the previous lower bound $E^l_{i+1,j+1,k+1}$.*

5 Continuity of Impact Maps

Algorithm 1 requires that the all states in $E_{i,j,k}$ will next switch at switching surface S_{j+1}. In addition to requiring the impact map to be continuous, the algorithm would need to also consider situations where points in the initial set would not map onto S_{j+1}. We consider some of these cases here.

5.1 Tangential trajectories

Consider a discrete mode q which is defined on the subset X_q of the state space, and assume that three of the boundaries of X_q are formed by the switching surfaces S_j, S_{j+1} and S_T. Let $E_{i,j,k} \subset S_j$ be a set of initial states which, in the absence of switching surface S_T would all switch at S_{j+1}. However, if, with S_T present, there exists at least one trajectory $x(\tau)$, starting from $E_{i,j,k}$, with a switching time to S_T smaller than the switching time to S_{j+1}, then that trajectory will switch at S_T first. This scenario is illustrated in Figure 5. To find the reach set we need to divide the initial set into those states that switch at S_T and those that switch at S_{j+1} and apply Algorithm 1 to each subset of $E_{i,j,k}$. To find which states switch at S_T we use the result in [7, Lemma 3.1] which says that trajectories tangential to S_T are those where

$$\left.\frac{d(C_{j+1}x(t) - d_{j+1})}{dt}\right|_{t=t_T} = 0 \qquad (18)$$

where t_T is the switching time to S_T. Any trajectories with a switching time to S_T less than t_T next switch at S_T.

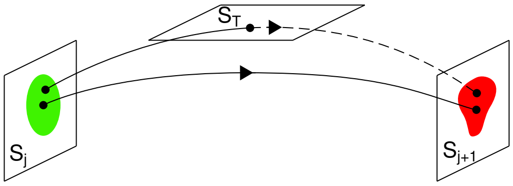


Figure 5. Trajectories may switch at S_T before reaching S_{j+1}.

5.2 *Intersecting switching surfaces*

Suppose the system is in mode q, with initial set $E_{i,j,k} \subset S_j$ and that the hyperplanes S_{j+1} and S_{j+2} now intersect. We consider the situation where the reach set includes a subset of the $n-2$-dimensional set of intersection between two switching surfaces S_{j+1} and S_{j+2}. To find the reach set on each of these hyperplanes, we apply Algorithm 1 from S_j to S_{j+1}, assuming the absence of S_{j+2}, giving the reach sets $E^u_{i+1,j+1,k+1}$ and $E^l_{i+1,j+1,k+1}$. Following this, the algorithm is applied from S_j to S_{j+2}, assuming the absence of S_{j+1}, giving the reach sets $E^u_{i+2,j+2,k+2}$ and $E^l_{i+2,j+2,k+2}$. Since the subset X_q is closed,the upper bound reach set is then given by $(E^u_{i+1,j+1,k+1} \cap X_q) \cup (E^u_{i+2,j+2,k+2} \cap X_q)$ and the lower bound is $(E^l_{i+1,j+1,k+1} \cap X_q) \cup (E^l_{i+2,j+2,k+2} \cap X_q)$. Since the reach set now lies on different switching surfaces, the reach set of the following iteration of the algorithm will be split into two. For one branch of the following reach set the algorithm uses as the new initial sets $E^l_{i+1,j+1,k+1}$ and $E^u_{i+1,j+1,k+1}$. For the other branch it uses $E^l_{i+2,j+2,k+2}$ and $E^u_{i+2,j+2,k+2}$.

5.3 *Non-switching sets*

In the cases where there exists an equilibrium point in the set X_q, we must isolate any points in the initial set $E_{i,j,k}$ which reach the equilibrium without switching. In certain cases, these points can be separated from points in the initial set that do switch by the hyperplanes of tangential trajectories given in (18). In other cases, a stable eigenvector passing through an initial set will cause any point in the initial set and on the eigenvector to go to the equilibrium without switching.

6 Example - Batch reactor

Controlled batch reactors are typical applications of cPWA. The processing of the reactants usually consists of several stages and the reaction environment (such as temperature, pressure, concentration of reactants) often has to be controlled. The method of control usually involves initiating a process in the reactor that will restore the system to its normal operating conditions when a certain threshold, typically a safety constraint, is breached.

In this example, a batch reactor tries to maintain the system temperature (x_3) above $10°$C by activating its heating element when the temperature falls below $30°$C. The system is modelled as a cPWA with three states, $x = [\, x_1 \;\; x_2 \;\; x_3 \,]^T$, the states being, respectively, the product yield, the amount of unused reactant, and the average temperature of the reactor and its contents. The reaction produces several species and it is required to estimate the minimum amount of the product x_1 that may be produced given an initial uncertainty in the amount of reactant and in the temperature, whilst maintaining the safety constraint. The reaction begins with no product present and ends when there is no more reactant.

The constraints are such that the system exhibits a total of four switching surfaces and three different modes of operation. The details of the dynamics of this system and of the switching surfaces are given in Appendix 7. The uncertainty is modelled in the state-space by a circular set of possible initial states on the switching surface $C_1 x = 0$, centered on the nominal initial state $x = [\, 0 \;\; 40 \;\; 40 \,]^T$. A nominal trajectory emanates from the nominal initial state.

The nominal trajectory of the system and the upper bounds and lower bounds on the reach sets, given the initial uncertainty, are shown in Figure 6. The individual reach sets on each switching surface, centered around the nominal trajectory's intersection with the switching surfaces, are shown in Figure 7. These results now give us a safety certificate and a measure of the performance of the system. To ensure that the $10°$C minimum temperature condition is not violated, we can express this constraint as a new switching surface, namely a hyperplane with equation $x_3 = 10$, and examine whether this hyperplane is reachable. The upper bound ellipse for the second switch shows that the system just satisfies temperature constraint (see Figure 7(b)). In addition, the upper bound ellipse on the fourth switch guarantees a minimum product yield of 13.3 units by the end of the reaction (see Figure 7(d)).

This computation took a total of 64.3720 seconds on a 1400 MHz Pentium M PC. This compares favorably with the similar three dimensional example in [11] which took 40 minutes on a 600 MHz Pentium III PC, even after allowing for the difference in processor speeds.

7 Discussion

We have presented a new method for placing bounds on reach sets in a cPWA given a set of possible initial states.

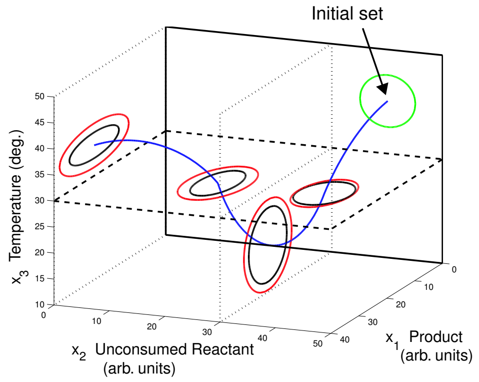
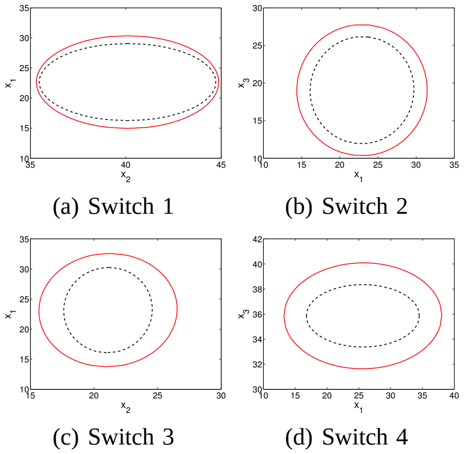


Figure 6. Upper bounds and lower bounds of reach sets on switching surfaces for a batch reactor system.



(a) Switch 1

(b) Switch 2

(c) Switch 3

(d) Switch 4

Figure 7. Upper bounds (solid ellipses) and lower bounds (dashed ellipses) on different switching surfaces.

This method relies on results in [6], which analyze cPWA by considering their behavior on the switching surfaces. Using this method, upper and lower bounds are placed on the image of a set of states that is mapped from one switching surface to another. By successively repeating this method for multiple switches, we obtain a set of ellipsoids on the switching surfaces that indicate where states in the initial set may reach after a finite amount of time.

As this method relies on solving an optimization of the solution of a set of LMIs, these reachability results can be obtained quicker than via older algorithms based on techniques such as face-lifting. Furthermore, this method can be applied to systems of high dimensions without the need to partition the state-space and discretize the simulation time. Moreover, by expressing constraints on the operation of the system in terms of new switching surfaces and assessing the reachability of those hyperplanes, we obtain results on such issues as the safety and performance of the system, as illustrated in the batch reactor example given in this paper.

A Example Details

The switching surfaces are given by

$$\begin{aligned} C_1x &= [1\ 0\ 0]x &= 0 \\ C_2x &= [0\ 0\ 1]x &= 30 \\ C_3x &= [0\ 1\ 0]x &= 30 \\ C_4x &= [0\ 1\ 0]x &= 0 \end{aligned} \tag{A.1}$$

The differential equations of each cell of the system in the positive orthant, as in (1) are as follows:

Region 1: $\{x : C_2x > 30\} \cap \{x : C_3x > 30\}$

$$\dot{x} = \begin{bmatrix} -0.1 & 0 & 1.0 \\ 0 & -0.1 & 0 \\ -1.0 & 0 & -0.1 \end{bmatrix} x + \begin{bmatrix} 0 \\ 40 \\ 0 \end{bmatrix} \tag{A.2}$$

Region 2: $\{x : C_2x < 30\}$

$$\dot{x} = \begin{bmatrix} 0.2 & 0 & 0 \\ 0 & -0.1 & 1.0 \\ 0 & -3.0 & -0.1 \end{bmatrix} x + \begin{bmatrix} 20 \\ 30 \\ 45 \end{bmatrix} \tag{A.3}$$

Region 3: $\{x : C_2x > 30\} \cap \{x : C_2x < 30\}$

$$\dot{x} = \begin{bmatrix} 0.2 & 0 & 0 \\ 0 & -0.1 & -1.0 \\ 0 & 1.0 & -0.1 \end{bmatrix} x + \begin{bmatrix} 15 \\ 5 \\ 20 \end{bmatrix} \tag{A.4}$$

References

[1] Eugene Asarin, Thao Dang, and Oded Maler. **d/dt**: A tool for reachability analysis of continuous and hybrid systems. In *Proc. IFAC Nonlinear Control Systems*, 2001.

[2] Eugene Asarin and Theo Dang Oded Maler. The d/dt tool for verification of hybrid systems. In *CAV'2002, Copenhagen, Denmark, 365-370, LNCS 2404 Springer-Verlag Postscript*, 2002.

[3] Oleg Botchkarev and Stavros Tripakis. Verification of hybrid systems with linear differential inclusions using ellipsoidal approximations. In *HSCC*, pages 73–88, 2000.

[4] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*. SIAM, Philadelphia, 1994.

[5] Thao Dang and Oded Maler. *Reachability Analysis via Face Lifting*, pages 96–109. Hybrid Systems: Computation and Control, First International Workshop, HSCC'98, Berkeley, California, USA, Proceedings. Lecture Notes in Computer Science, Springer, 1998.

[6] J. M. Goncalves, A. Megretski, and M. A. Dahleh. Global analysis of piecewise linear systems using impact maps and surface lyapunov functions. *IEEE Transactions on Automatic Control*, 48(12):2089–2106, December 2003.

[7] J.M. Goncalves. Regions of stability for limit cycles of piecewise linear systems. In *CDC, Maui, Hawaii*, December 2003.

[8] James Kapinski and Bruce H. Krogh. Verifying switched-mode computer controlled systems. In *IEEE International Symposium on Computer Aided Control System Design*, 2002.

[9] A. Kurzhanski and P. Varaiya. On ellipsoidal techniques for reachability analysis. *Optimization Methods and Software*, 17:177237, 2000.

[10] P. A. Parrilo. *Structured Semidefinite Programs and Semialgebraic Geometry Methods in Robustness and Optimization*. PhD thesis, California Institute of Technology, 2000.

[11] B. Izaias Silva, Olaf Stursberg, Bruce Krogh, and Sebastian Engell. An assessment of the current status of algorithmic approaches to the verification of hybrid systems. In *IEEE Conference on Decision and Control*, 2001.