Hybridization for Stability Verification of Nonlinear Switched Systems

Miriam García Soto  
Henzinger Group  
IST Austria  
Klosterneuburg, Austria  
Email: miriam.garciasoto@ist.ac.at

Pavithra Prabhakar  
Department of Computer Science  
Kansas State University  
Manhattan, KS  
Email: pprabhakar@ksu.edu

Abstract—We propose a novel hybridization method for stability analysis that over-approximates nonlinear dynamical systems by switched systems with linear inclusion dynamics. We observe that existing hybridization techniques for safety analysis that over-approximate nonlinear dynamical systems by switched affine inclusion dynamics and provide fixed approximation error, do not suffice for stability analysis. Hence, we propose a hybridization method that provides a state-dependent error which converges to zero as the state tends to the equilibrium point. The crux of our hybridization computation is an elegant recursive algorithm that uses partial derivatives of a given function to obtain upper and lower bound matrices for the over-approximating linear inclusion. We illustrate our method on some examples to demonstrate the application of the theory for stability analysis. In particular, our method is able to establish stability of a nonlinear system which does not admit a polynomial Lyapunov function.

Keywords—hybridization; non-linear dynamics; stability analysis;

I. INTRODUCTION

Embedded control systems consist of software controlled physical systems that enable sophisticated functionalities such as autonomous driving in vehicles and automated load balancing in smart grids. The safety criticality of these systems demands rigorous analysis methodologies to ensure their adherence to the correct functionalities. Hybrid systems theory provides a mathematical framework for the modelling and analysis of mixed discrete-continuous behaviors exhibited due to the interaction of discrete software components with continuous physical entities in embedded control systems. In this paper, we focus on the analysis of an important correctness specification of embedded control systems, namely, stability.

Stability is a fundamental property in control system design that stipulates that small perturbations to the initial state of the system lead to only small deviations in the resulting behaviors of the system (Lyapunov stability), and that the effect of small perturbations to the initial state eventually vanish (asymptotic stability). Stability analysis has been extensively studied in the domain of control theory [1]; however, stability analysis remains a challenge, especially for nonlinear, switched and hybrid systems. There are broadly two methods for the analysis of nonlinear systems [2], namely, linearization (Lyapunov’s first method) and Lyapunov’s second method. The first method consists of constructing a linear approximation \( \dot{x} = Ax \) of the nonlinear system \( \dot{x} = f(x) \), and determining the stability of the latter by examining the eigenvalues of the matrix \( A \). However, this method in general only applies to deduce asymptotic stability, and Lyapunov stability of, especially marginally stable systems, cannot be concluded using this method, since, small imprecisions in the approximation can destabilize a marginally stable system. In addition, eigenvalue-based analysis does not extend to non-linear, switched or hybrid systems.

Lyapunov’s second method establishes the stability of a system \( \dot{x} = f(x) \) by exhibiting a function, called a Lyapunov function, that serves as a certificate of stability. A Lyapunov function is a continuously differentiable function \( V \) from the state space to non-negative reals, such that \( V(0) = 0 \) only at 0 (assumed to be the equilibrium point) and the value of the function decreases along any trajectory of the system. Automated methods for computing Lyapunov functions essentially consist of a template-based search, wherein, say, a polynomial with coefficients as parameters is chosen as a candidate Lyapunov function, and the conditions of Lyapunov functions are encoded as constraints on the parameters, which are then solved using certain semi-definite programming techniques such as sum-of-squares programming [3]–[5]. However, the challenges with Lyapunov’s second methods are choosing the appropriate templates and the drastic increase in the computation time with the increase in the degree of the polynomials. There has been some work on learning Lyapunov functions [6]. Extensions of Lyapunov functions to hybrid systems stability analysis have been proposed using the notions of common and multiple Lyapunov functions [4], where either a single function that serves as a Lyapunov function for every mode of the system is chosen, or a set of functions, wherein each serves as a Lyapunov function for a particular mode, is chosen along with additional constraints that are required to be satisfied.
at the switching surface. However, Lyapunov function based methods were shown to suffer from numerical issues and to be sensitive to the encoding of the switching constraints [7].

In this paper, we present an alternate stability analysis method for nonlinear switched systems based on abstractions; an abstraction based approach broadly consists of constructing simplified systems and analyzing these simplified systems to infer correctness of the given systems. Switched systems [8] consist of a finite set of continuous systems along with a time-based or state-based switching logic. Here, we consider switched systems that are specified by a partition \( \{P_i\} \) of the state-space and a dynamics \( \dot{x} = F_i(x) \) associated with each of the regions \( P_i \). The system evolves according to \( \dot{x} = F_i(x) \) while in region \( P_i \) and switches to the dynamics \( \dot{x} = F_j(x) \) at the boundary of \( P_i \) and \( P_j \).

The main result of the paper is a novel hybridization technique for stability analysis that over-approximates a switched (nonlinear) system by a switched system with linear inclusion dynamics. Hybridization [9]–[11], refers to an over-approximation method, where the state-space is divided into a finite number of regions and the dynamics restricted to each region is over-approximated by a computationally more tractable dynamics, such as piecewise affine or polyhedral dynamics. For instance, in [11], the nonlinear dynamics \( \dot{x} = f(x) \) is over-approximated by piecewise affine dynamics as follows. Given a partition of the state space into regions \( P_1, \ldots, P_k \), in each region \( P_i \), the dynamics is over-approximated by an affine dynamics \( \dot{x} = Ax + u \), where \( u \in [-\epsilon, \epsilon]^n \) for some \( \epsilon > 0 \). Here, \( \epsilon \) captures the greatest distance between \( f(x) \) and the approximation \( Ax \) over the region \( P_i \).

While existing hybridization techniques are useful for safety analysis, we observe that they do not yield useful results for stability analysis. More precisely, let us consider the stability of \( \dot{x} = f(x) \) with respect to the equilibrium point \( 0 \), that is, \( f(0) = 0 \). Note that \( \dot{x} = Ax + u \), \( u \in [-\epsilon, \epsilon]^n \) is not stable with respect to \( 0 \), for any matrix \( A \); in particular, it does not even have \( 0 \) as an equilibrium point. Our main insight is that the over-approximation cannot have a fixed error \( \epsilon \) for every point in the domain of approximation \( P_i \), but instead the error should reduce as we move closer to the equilibrium point. Hence, we seek an approximation \( \dot{x} = Ax + \gamma x \), \( \gamma \in [-\epsilon, \epsilon] \) or equivalently, \( \dot{x} \in \{(A + \gamma I)x : \gamma \in [-\epsilon, \epsilon]\} \), where the error \( [-\epsilon x, \epsilon x] \) converges to \( 0 \) as \( x \) tends to \( 0 \). In particular, we intend to find two linear functions that bound the nonlinear function \( f(x) \) as in \( A_1x \leq f(x) \leq A_2x \), referred to as a linear inclusion dynamics, where \( \leq \) denotes component-wise comparison between vectors.

Our broad idea for approximating a function \( f(x) \) is based on bounding the partial derivatives of \( f(x) \) and using the bounds to define the linear functions. Traditional hybridization uses bounds of \( f(x) \) instead of the derivatives. More precisely, in the one-dimensional case, we show that if \( a \leq \frac{df}{dx} \leq b \) for \( x \in [0, \infty) \), then \( ax \leq f(x) \leq bx \). We generalize this observation to higher dimensions, however, the extension is non-trivial, since, the component-wise upper and lower bounds on partial derivatives for each dimension do not provide the bounds for \( f \). We propose an alternate representation for polyhedral sets using a finite number of upper and lower bound functions, and present a recursive definition for the computation of upper and lower bounded matrices using those computed for appropriate functions of lower dimension.

The resulting approximate system, which is a switched system with linear inclusion dynamics, can be analyzed for establishing stability of the original nonlinear system. We use the the tool AVERIST [12] for the stability analysis of the hybridized system and illustrate our method on some examples, including proving stability on a system for which no polynomial Lyapunov function exists [13]. To the best of our knowledge, this is the first investigation on the application of abstractions, in particular, hybridization, for the analysis of stability of nonlinear and switched dynamical systems. Abstraction based approaches for stability verification of linear and polyhedral inclusion dynamics have been explored before [7], [14].

II. PRELIMINARIES

In this section, we present some notations and definitions that we will use in the following sections.

**Euclidean space:** Let \( \mathbb{R} \) denote the set of reals, \( \mathbb{R}_{\geq 0} \) the set of non-negative reals and \( \mathbb{R}_{\leq 0} \) the set of non-positive reals. An interval is a closed convex subset of \( \mathbb{R} \). A time interval is an interval \( I \) of the form either \( [0, T] \) for some \( T \in \mathbb{R}_{\geq 0} \) or \( [0, \infty) \). A signed interval is an interval of the form \( [0, T], [0, \infty], [-T, 0] \) or \( (-\infty, 0] \). The \( n \)-dimensional Euclidean space is given by \( \mathbb{R}^n \). Given a point \( x \in \mathbb{R}^n \), we use \( x_i \) to denote the \( i \)-th component of \( x \), that is, \( x = (x_1, \ldots, x_n) \), and we denote \( x_k \) the point \( (x_k, \ldots, x_n) \in \mathbb{R}^{n-k+1} \) for any \( k \in \{1, \ldots, n\} \). Observe that we use bold letters for multidimensional points. We use \( ||x|| \) to denote the infinity norm of \( x \in \mathbb{R}^n \), and \( \langle x, y \rangle \) to denote the dot product of \( x, y \in \mathbb{R}^n \). Given a set \( X \subseteq \mathbb{R} \), the powerset of \( X \), denoted \( \mathcal{P}(X) \), is the set of all subsets of \( X \). We denote a matrix \( A \in \mathbb{R}^{n \times n} \) as \( (a_{ij})_{1 \leq i, j \leq n} \), where \( a_{ij} \in \mathbb{R} \) is the \((i,j)\)-th element of \( A \).

**Polyhedral sets:** A linear constraint is an expression of the form \( \langle a, x \rangle \geq b \), where \( a \in \mathbb{R}^n \) is a tuple of values, \( b \) is a real value, \( x = (x_1, \ldots, x_n) \) is a tuple of variables. A half-space is a set defined by a linear constraint with inequality relation. A polyhedral set is determined by an intersection of finitely many half-spaces. It is said to be pointed if one of its vertices is the origin, \( 0 \).

An orthant in \( \mathbb{R}^n \) is a polyhedral set \( \mathcal{O}_e = \{I(e_1) \times \ldots \times I(e_n) \} \), where \( e \in \{-1, 1\}^n \), \( I(-1) = (-\infty, 0] \) and \( I(1) = [0, \infty) \). An orthant in \( \mathbb{R} \) is known as a ray while in
$\mathbb{R}^2$ is known as a quadrant. We call orthant-polyhedron a polyhedral set $P$ such that there exists $\epsilon \in \{-1, 1\}^n$ with $P \subseteq Q_\epsilon$. A pointed orthant-polyhedron $P$ is said to be a polyhedron. Given $P \subseteq \mathbb{R}^n$ and $1 \leq i_1, \ldots, i_k \leq n$, the projection of $P$ into $\{i_1, \ldots, i_k\}$ is defined as $\text{proj}_{i_1, \ldots, i_k}(P) = \{(y_1, \ldots, y_k) : \exists x \in P \text{ such that } y_1 = x_{i_1}, \ldots, y_k = x_{i_k}\}$. An $n$-dimensional polyhedral partition of $X \subseteq \mathbb{R}^n$ is a finite tuple $\mathcal{P} = (P_1, \ldots, P_k)$ such that $X = \bigcup_{P \in \mathcal{P}} P$, $P_i \neq \emptyset$ for every $i$ and $P_i \cap P_j = \emptyset$ for every $i \neq j$, where $P$ is the interior of $P$. A refinement partition of $\mathcal{P}$ is a polyhedral partition $\mathcal{Q} = (Q_1, \ldots, Q_l)$ of $X$ such that for every $Q_i \in \mathcal{Q}$ there exists a $P_j$ with $Q_i \subseteq P_j$. Such a relation between $Q_i$ and $P_j$ is denoted by $\text{ind}(P_i, Q_j) = j$.

Functions and derivatives: Given a function $F$, we use $\text{dom}(F)$ to denote the domain of $F$, and $\text{img}(F)$ to denote the image of $F$. A function $F$ is said to be an $n$-dimensional function if $\text{dom}(F) \subseteq \mathbb{R}^n$, a real-valued function if $\text{img}(F) \subseteq \mathbb{R}$ and a vector-valued if $\text{img}(F) \subseteq \mathbb{R}^n$. Given a function $F$ from $X$ to $\mathbb{R}^n$, the components of $F$ are the functions $F_i : X \to \mathbb{R}$, where $F_i(x)$ is the $i$-th component of $F(x)$ with $1 \leq i \leq n$. We say a function $F : X \to Y$ is of class $C^k(X)$ if every component of the function is continuously differentiable with respect to $k$ coordinates, which can be different or just the same. A function $F$ is bounded if $\text{img}(F)$ is bounded. Given a one-dimensional function $F : \mathbb{R} \to \mathbb{R}^n$, we use $\dot{F}$ to denote the derivative $\frac{d}{dt}F(t)$. A function $F$ from a polyhedral set $P$ to $\mathbb{R}^n$ is said to be linear if there exist $a \in \mathbb{R}^n$ and $b \in \mathbb{R}$, such that for every $x \in P$, $F(x) = (x, a) + b$; and piecewise-linear if there exists a polyhedral partition $\mathcal{P}$ of $P$ such that for every $Q \in \mathcal{P}$, the function $F$ restricted to $Q$ is linear.

A. Ordered-bounded representation and maximal functions

Recall that given $x \in \mathbb{R}^n$, we denote $x_k$ the point $(x_1, \ldots, x_k) \in \mathbb{R}^{n-k+1}$ for any $k \in \{1, \ldots, n\}$.

Definition 1. Let $P$ be a polyhedron in $\mathbb{R}^n$. The ordered-bounded representation of $P$ is $\{(l_1^k, \ldots, l_n^k), (u_1^k, \ldots, u_n^k)\}$ where $l_k^p$ and $u_k^p$ are piecewise linear functions over $\mathbb{R}^{n-k}$ for $1 \leq k < n$, and $l_n^p$ and $u_n^p$ are constant values in $\mathbb{R}$ such that $P = \{x \in \mathbb{R}^n : l_k^p(x_k) \leq x_k \leq u_k^p(x_k) \text{ for } 1 \leq k < n, l_n^p \leq x_n \leq u_n^p\}$.

Remark 1. We will ignore the superscript $k$, since, it is often clear from the argument of $l_k^p$ and $u_k^p$ functions, and instead will denote them as $l_p$ and $u_p$.

We note that the functions $l_p$ and $u_p$ are piecewise linear functions, which can be computed effectively from a description of $P$ as conjunctions of linear constraints. Figure 1 shows a polyhedron $P$, depicted in blue and described by 3 linear constraints, $P = \{x \in \mathbb{R}^2 : x_1 - x_2 \leq 0, x_1 \geq 0, x_1 + x_2 \geq 1\}$. Observe that this is not a unique description of $P$. The ordered-bounded representation is an alternative description where we first enclose the variable $x_2$ between two values, which in this case are as follows $0.5 \leq x_2 \leq \infty$, so $l_p = 0.5$ and $u_p = \infty$. Second, we enclose the variable $x_1$ between functions of $x_2$. Observe in Figure 1 that the dotted black line $x_2 = 1$ partitions the polyhedron $P$ into a lower and an upper region. The lower region corresponds to the triangular area delimited by the orange, green and black-dotted lines, where $0.5 \leq x_2 \leq 1$ and $x_1$ is lower bounded by the orange line and upper bounded by the green line. The upper region corresponds to the unbounded region with $x_2 \geq 1$ and where $x_1$ is lower bounded by the blue line and upper bounded by the green line. Formally, we define $l_p(x_2) = \begin{cases} 1 - x_2 & \text{if } 0.5 \leq x_2 \leq 1 \\ 0 & \text{if } x_2 \geq 1 \end{cases}$ and $u_p(x_2) = x_2$, and the ordered-bounded representation of $P$ is $\{x \in \mathbb{R}^2 : l_p(x_2) \leq x_1 \leq u_p(x_2), l_p \leq x_2 \leq u_p\}$.

Lemma 1. Given an $n$-dimensional convex and closed polyhedral set $P$, there exist two piecewise linear functions $l_p, u_p : \mathbb{R}^{n-1} \to \mathbb{R}$ such that $P = \{x \in \mathbb{R}^n : l_p(x_2) \leq x_1 \leq u_p(x_2)\} \cap \mathbb{R} \times \text{proj}_{2,\ldots,n}(P)$.

Proposition 1. Given a convex and closed polyhedral set $P$ in $\mathbb{R}^n$, an ordered-bounded representation of $P$ is computable.

Both proof of Lemma 1 and proof of Proposition 1 can be found in the Appendix.

III. Switched systems

In this section, we define a class of switched systems [8], where the continuous state-space is partitioned into a finite number of regions, each of which is associated with a differential inclusion. Each of the regions corresponds to a (discrete) operational mode, and differential inclusion specifies the evolution of the continuous state in that mode. The continuous state does not change (reset) during a mode switch.
Definition 2. An $n$-dimensional switched system (SS) is a tuple $S = (\mathcal{P}, \mathcal{F})$, where $\mathcal{P} = (P_1, \ldots, P_k)$ is a polyhedral partition and $\mathcal{F}$ is a tuple of functions from $\mathbb{R}^n$ to $\mathcal{P}(\mathbb{R}^n)$, $(F_1(x), \ldots, F_k(x))$.

Next, we define an execution that specifies the evolution of the continuous state of the system with time elapse. The execution follows the solution of the differential inclusion $\dot{x} \in F_i(x)$ while in the polyhedral region $P_i$, and switches to a solution of the differential inclusion of an adjacent polyhedral region at the boundary.

Definition 3. An execution $\sigma$ of an $n$-dimensional SS $S = (\mathcal{P}, \mathcal{F})$ is a continuous function $\sigma : I \rightarrow \mathbb{R}^n$, where $I$ is a time interval, such that there exists a finite or infinite sequence of pairs of polyhedral sets and real values of the form $(P_{i_1}, t_1), \ldots, (P_{i_k}, t_k), \ldots$ satisfying the following for every $i \geq 1$:

- $0 = t_0 < t_1 < \ldots < t_k < \ldots$,
- $P_i \in \mathcal{P}$,
- $\sigma(t) \in P_i$ for every $t \in [t_{i-1}, t_i]$ and
- $\frac{d}{dt} \sigma(t) \in F_i(\sigma(t))$ for every $t \in [t_{i-1}, t_i]$.

An execution $\sigma : I \rightarrow \mathbb{R}^n$ of $S$ is complete if $I = [0, \infty)$; otherwise, it is finite. We denote the set of all executions of the system $S$ by $\text{exec}(S)$, and the set of all complete executions by $\text{exec}_{\text{finite}}(S)$.

Next, we identify certain subclasses of switched systems. Firstly, a differential equation is a special type of differential inclusion $\dot{x} \in F(x)$, where $F(x)$ is a singleton set. Hence, we model a single nonlinear system $\dot{x} = f(x), x \in P$, where $P$ is a polyhedral set and $f$ is a nonlinear function, as a switched system $S = (\mathcal{P}, \mathcal{F})$, with $\mathcal{P} = (P)$ and $\mathcal{F} = (F)$, where $F(x) = \{f(x)\}$. We will also denote the single nonlinear system as $S = (P, f)$. Next, we say that a switched system $S = (\mathcal{P}, \mathcal{F})$ is a nonlinear switched system if $F_i(x) = \{f_i(x)\}$ for every $x \in P_i$, where $f_i$ is a nonlinear function. A switched system $S = (\mathcal{P}, \mathcal{F})$ is a polyhedral inclusion switched system if $F_i(x) = Q$ for every $x \in P_i$, where $Q \subseteq \mathbb{R}^n$ is a polyhedral set. A linear inclusion switched system $S = (\mathcal{P}, \mathcal{F})$ is such that $F_i(x) = \{y \in \mathbb{R}^n : Ax \leq y \leq Bx\}$ where $A, B \in \mathbb{R}^{n \times n}$.

IV. LYAPUNOV STABILITY

Stability is a fundamental property of dynamical systems, that is considered as an important objective in the control system design. It ensures that small perturbations to the initial state of the system result in just small deviations of the nominal behaviour of the system, and also that the effect of the perturbations eventually subsides. Here, we introduce two classical notions of stability in control theory, namely, Lyapunov stability and asymptotic stability. These notions of stability are considered with respect to an equilibrium state, which corresponds to a state where the system does not change with time. We assume that the origin $0$ is an equilibrium state, without loss of generality. For a non-linear system $\dot{x} = f(x)$, this means that $f(0) = 0$. Intuitively, a system is Lyapunov stable if executions that start close to the equilibrium remain close to it.

Definition 4. A set of executions $\Sigma$ of an SS $S$ is Lyapunov stable with respect to the equilibrium state $0$ if for every $\epsilon \in \mathbb{R}_{>0}$ there exists $\delta \in \mathbb{R}_{>0}$ such that every execution $\sigma : I \rightarrow \mathbb{R}^n$ belonging to $\Sigma$, with $\sigma(0) \in B_\delta(0)$, satisfies $\sigma(t) \in B_\epsilon(0)$ for every $t \in I$.

A system is asymptotically stable if, in addition to being Lyapunov stable, satisfies that every execution starting close enough to the equilibrium converges to it.

Definition 5. An execution $\sigma : I \rightarrow \mathbb{R}^n$ is said to converge to $0$ if for every $\epsilon \in \mathbb{R}_{>0}$ there exists a time $T \in I$ such that $\sigma(t) \in B_\epsilon(0)$ for every $t \geq T$.

Definition 6. A set of executions $\Sigma$ of an SS $S$ is asymptotically stable with respect to $0$ if it is Lyapunov stable and there exists $\eta \in \mathbb{R}_{>0}$ such that every execution $\sigma \in \Sigma$ with $\sigma(0) \in B_\eta(0)$ converges to $0$.

We say that a switched system $S$ is Lyapunov stable if $\text{exec}(S)$ is Lyapunov stable with respect to $0$. A switched system $S$ is said to be asymptotically stable if $\text{exec}_{\text{finite}}(S)$ is asymptotically stable with respect to $0$.

V. HYBRIDIZATION

Checking stability of a nonlinear system $\dot{x} = f(x)$ is a challenging problem, more so when it is part of a switched system. In this section, we present a general method to abstract a nonlinear dynamical system by a “switched” linear inclusion system. The latter is a class of hybrid systems which can be analyzed, for instance, using results from [7].

A. Motivation

Our abstraction falls into a general technique called hybridization, wherein the state-space is partitioned into a finite number of regions, and the dynamics restricted to each region is over-approximated by a simpler dynamics [9], [10]. For instance, given a partition of the state-space $\mathcal{P} = (P_1, \ldots, P_k)$ and a nonlinear system $\dot{x} = f(x)$, a simpler affine dynamics $\dot{x} = A_i x + u, u \in B_i(0)$ for some $\epsilon > 0$ is constructed for every $P_i$, where $f(x) \in F_i(x) = \{A_i x + u : u \in B_i(0)\}$ for all $x \in P_i$. Thus, the switched linear system $(\mathcal{P}, (F_1, \ldots, F_k))$ over-approximates the behaviors of $(\cup_i P_i, f)$. While such affine approximations are useful in proving safety [9], [11], they are not conducive for stability analysis. Note that $F_i(0) = B_i(0)$, which does not imply that $0$ is an equilibrium point. More importantly, there is a fixed error of $\epsilon > 0$ between the concrete and the abstract vector fields even at a very small distance from the equilibrium point.
For illustration, consider the nonlinear system \( \dot{x} = -\sin x \) restricted to the interval \([0, 1]\)\. Existing hybridization techniques overapproximate the nonlinear system by \( \dot{x} \in [-0.75x, -0.75x + [-\epsilon, \epsilon]] \) for \( \epsilon \in \mathbb{R}_{>0} \). Figure 2a shows the result of such a hybridization procedure, where the function \(-\sin x\) is depicted in red, and the region where the overapproximated derivative \( \dot{x} \) lies is represented in light red. Observe that for \( x(t) = 0 \), we conclude \( \dot{x}(t) \in [-\epsilon, \epsilon] \). Therefore, there is no equilibrium point but an interval, and stability cannot be analyzed with respect to it.

Since, intuitively, stability requires that small perturbations in the state with respect to 0 lead to small deviations, we need an abstraction in which the error between the state with respect to \( t \) approximations in the state lies is represented in light red. Therefore, there is no equilibrium point but an interval, and stability cannot be analyzed with respect to it.

Definition 7. Given an \( n \)-dimensional function \( f \) and a polyhedral set \( P \subseteq \mathbb{R}^n \), a linear inclusion approximation of \( f \) in \( P \) is a function \( G \) from \( P \) to \( \mathbb{R}^n \), such that there exist \( A, B \in \mathbb{R}^{n \times n} \), where for every \( x \in P \), \( f(x) \in G(x) = \{ y \in \mathbb{R}^n : Ax \leq y \leq Bx \} \).

Definition 8. Given a \( n \)-dimensional nonlinear switched system \( S = (P, F) \) and a polyhedral set \( R \) and a refinement partition \( Q = (Q_1, \ldots, Q_k) \) of \( P \), we define a hybridized switched system \( \mathcal{H}(S, R, Q) \) as the switched system \( (Q, \mathcal{G}) \) with \( \mathcal{G} = (G_1, \ldots, G_k) \), where \( G_i \) is a linear inclusion approximation of \( f_{\text{ind}}(P, Q_i) \) in \( Q_i \cap R \) for every \( 1 \leq i \leq k \).

Note that the nonlinear system and the switched system are defined over the whole state-space \( \mathbb{R}^n \), however, the linear inclusion dynamics over-approximates the actual dynamics only on some polyhedral set \( R \), which is usually a compact set. Restricting to \( R \) is necessary to obtain tight upper and lower bounds for \( f(x) \). On the other hand, the stability of the hybridized system still implies the stability of nonlinear system, since, stability is a property about a small neighborhood around the equilibrium. This is summarized in the following theorem.

**Theorem 1.** Given a \( n \)-dimensional nonlinear switched system \( S = (P, F) \), a polyhedral set \( R \) containing \( B_\gamma(0) \) for some \( \gamma > 0 \) and a polyhedral partition \( P = (P_1, \ldots, P_k) \) of \( \mathbb{R}^n \), if the hybridized switched system \( \mathcal{H}(S, R, P) \) is Lyapunov (asymptotically) stable with respect to 0, then \( S \) is Lyapunov (asymptotically) stable with respect to 0.

**Proof:** Suppose \( \mathcal{H}(S, R, P) \) is Lyapunov stable. We need to prove that \( S \) is Lyapunov stable. Let us fix \( \epsilon > 0 \). Let \( \eta > 0 \) be such that \( \epsilon < \gamma \) and \( B_{2\gamma}(0) \subseteq R \). From Lyapunov stability of \( \mathcal{H}(S, R, P) \), let \( \delta > 0 \) be such that all executions \( \sigma \) of \( \mathcal{H}(S, R, P) \) with \( \sigma(0) \in B_{\delta}(0) \) remain within \( B_{\gamma}(0) \). We claim that all executions \( \sigma' \) of \( S \) which start within \( B_{\delta}(0) \) will also remain within \( B_{\gamma}(0) \). Suppose not, then \( \sigma' \) will at some time go out of \( B_{\gamma}(0) \). Let \( T \) be a time such that \( \sigma' \) is out of \( B_{\gamma}(0) \) but still has not left \( B_{2\gamma}(0) \). Hence, \( \sigma' \) until time \( T \) is still within \( R \). Then \( \sigma' \) until time \( T \) is an execution of \( \mathcal{H}(S, R, P) \) as well, since, the derivative of \( \sigma' \) until time \( T \) will lie in the linear inclusions of the corresponding regions in \( P \). However, \( \sigma' \) started in \( B_{\delta}(0) \), and reached a point outside \( B_{\gamma}(0) \) in \( \mathcal{H}(S, R, P) \), which contradicts the choice of \( \delta \).

Asymptotic stability of \( S \) follows from a similar observation that if executions starting from an \( \eta \) neighborhood of \( \mathcal{H}(S, R, P) \) converge, then executions starting from a small enough neighborhood will remain within \( R \) for all times, and hence, will converge as well.

Next, we propose a technique for abstracting a nonlinear function to a linear inclusion approximation in a particular region of the state-space.

**VI. Linear Inclusion Abstraction**

In this section, we present the crux of the hybridization procedure, that is, a method for over-approximating a nonlinear function by a linear inclusion over a polyhedral set. Consider a single nonlinear system \( S = (\mathbb{R}^n, f) \), and a compact po-polyhedron \( P \), where \( f : \mathbb{R}^n \rightarrow \mathbb{R}^n \) is a function of class \( C^1(\mathbb{R}^n) \), that is, is differentiable. The goal is to find two matrices \( A, B \in \mathbb{R}^{n \times n} \), such that \( Ax \leq f(x) \leq Bx \) for every \( x \in P \). We construct such matrices by iteratively constructing linear bounds on each
component \( f_i : \mathbb{R}^n \to \mathbb{R} \) of \( f \). More precisely, we compute vectors \( a, b \in \mathbb{R}^n \) such that \( (a, x) \leq f_i(x) \leq (b, x) \) for every \( x \in P \). Existence of such vectors is summarized in the following theorem.

**Theorem 2.** Let \( f : \mathbb{R}^n \to \mathbb{R} \) be an \( n \)-dimensional real-valued function of class \( C^1(P) \), where \( f(0) = 0 \) and \( P \) is a compact polyhedron. Then, there exist vectors \( a, b \in \mathbb{R}^n \) such that \( (a, x) \leq f(x) \leq (b, x) \) for every \( x \in P \).

**A. One dimensional nonlinear function approximation**

First, we prove Theorem 2, when \( f \) is a one-dimensional function, that is, \( f : \mathbb{R} \to \mathbb{R} \). The broad idea is to obtain a bound on \( f \) by bounding their derivatives. The following lemma states the existence of two constant values which linearly bound the value of a one-dimensional real-valued function.

**Lemma 2.** Let \( f(x) \) be a one-dimensional real-valued function such that \( f(0) = 0 \) and \( f \in C^1(I) \) where \( I \) is a signed interval. Let \( a \) and \( b \in \mathbb{R} \) be real values such that \( a \leq \frac{d}{dx} f(x) \leq b \) for all \( x \in I \). Then, \( ax \leq f(x) \leq bx \) when \( I \subseteq [0, \infty) \) and \( bx \leq f(x) \leq ax \) when \( I \subseteq (-\infty, 0] \).

**Proof:** We know that \( a \leq \frac{d}{dx} f(x) \leq b \) for all \( x \in I \). Let us consider \( y \in I \). In the case of \( I \subseteq [0, \infty) \), by integrating from \( 0 \) to \( y \) with respect to \( dx \), we obtain the following

\[
\int_0^y a \, dx \leq \int_0^y \frac{d}{dx} f(x) \, dx \leq \int_0^y b \, dx,
\]

which is \( ay - 0 \leq f(y) - f(0) \leq by - 0 \). Hence, \( ay \leq f(y) \leq by \) for every \( y \in I \). In the case of \( I \subseteq (-\infty, 0] \), by integrating from \( y \) to \( 0 \) with respect to \( dx \), we obtain the following

\[
\int_y^0 a \, dx \leq \int_y^0 \frac{d}{dx} f(x) \, dx \leq \int_y^0 b \, dx,
\]

which is \( 0 - ay \leq f(0) - f(y) \leq 0 - by \). Hence, \( by \leq f(y) \leq ay \) for every \( y \in I \).

**Remark 2.** Observe that Lemma 2 is a consequence of the mean-value theorem.

**Example 1.** Consider the dynamical system \( \dot{x} = -\sin x \) restricted to the interval \([0, 1]\). We want to compute \( a, b \in \mathbb{R} \) such that for every \( x \in [0, 1] \), \( ax \leq -\sin x \leq bx \). Figure 2b shows two sample linear functions \( ax \) and \( bx \), which contain the function \(-\sin x \). Hence, the tightest values of \( a \) and \( b \) will be the minimum and maximum slope of the function \(-\sin x \) in the considered interval \([0, 1]\). This is captured by the inequality \( a \leq -\cos x \leq b \), where \(-\cos x \) is obtained by differentiating \(-\sin x \) with respect to \( x \). Therefore, the bounds are obtained by computing the maximum and minimum of \(-\cos x \) in the interval \([0, 1]\). The maximum (minimum) of \(-\cos x \) is obtained by equating its own derivative to zero, that is, by solving \( \sin x = 0 \). This last equation give us the solution \( x^* = 0 \). The other bound will correspond to an endpoint of the interval, that is, \( x^* = 1 \). Then, \( a = -\cos 0 = -1 \) and \( b = -\cos 1 = -0.54 \). Hence, the over approximation is as follows, \(-x \leq -\sin x \leq -0.54x \) for every \( x \in [0, 1] \). Such an approximation is shown in Figure 2b, where the upper bound function, \( y = -0.54x \), is depicted in blue, the lower bound function \( y = -x \), is depicted in green, and the nonlinear function, \( y = -\sin x \), is depicted in red.

**B. Multi-dimensional nonlinear function approximation**

In this section, we describe the idea behind the extension of Lemma 2 to multiple dimensions, and provide the details. Unlike Lemma 2, a multidimensional real-valued function \( f : \mathbb{R}^n \to \mathbb{R} \) satisfying the inequality \( a \leq \nabla f(x) \leq b \), where \( \nabla \) is the gradient, need not satisfy \( (a, x) \leq f(x) \leq (b, x) \). That is, finding bounds on the gradient of \( f \) does not provide coefficients for a sound linear inclusion approximation of \( f \). The computation of coefficients for a sound linear inclusion approximation of \( f \) in a polyhedron \( P \) requires, among other tasks, to construct the ordered-bounded representation of \( P \) and incorporate it in the definition of functions derived from \( f \).

For illustration, let us consider a two dimensional function \( f(x_1, x_2) \) restricted to a compact polyhedron \( P \). Let us assume that \( P \) is contained in the first quadrant. We start by computing the partial derivative of \( f \) with respect to the first variable, \( x_1 \), and optimize over the set \( P \), thus, obtaining

\[
a_1 \leq \frac{\partial}{\partial x_1} f(x_1, x_2) \leq b_1,
\]

for every \((x_1, x_2) \in P\). Following the proof of Lemma 2, we fix a point \((y_1, y_2) \in P\), and we intend to find linear functions over \( y_1 \) and \( y_2 \) which bound the nonlinear function \( f \) from above and below. Note that we cannot integrate the inequality in (1) from 0 to \( y_1 \), because, the inequality need not hold for all points \( x_1 \) in the interval \([0, y_1]\), as the inequality holds for those \( x_1 \) such that \((x_1, x_2) \in P\). Given \( x_2 \), let \( l_P(x_2) \) denote the lowest value of \( x_1 \) such that \((x_1, x_2) \in P\). We can now integrate the inequality in (1) from \( l_P(x_2) \) and \( y_1 \), and obtain the following:

\[
a_1 y_1 - a_1 l_P(x_2) \leq f(y_1, x_2) - f(l_P(x_2), x_2) \leq b_1 y_1 - b_1 l_P(x_2)
\]

(2)

for every \((y_1, x_2) \in P\). Let us consider the lower bound, by reorganizing the above inequality, we obtain

\[
a_1 y_1 + f(l_P(x_2), x_2) - a_1 l_P(x_2) \leq f(y_1, x_2).
\]

Let us define \( g(x_2) \) to be

\[
f(l_P(x_2), x_2) - a_1 l_P(x_2),
\]

then we have

\[
a_1 y_1 + g(x_2) \leq f(y_1, x_2)
\]

(3)

for every \((y_1, x_2) \in P\). Note that \( g \) is a function over the variable \( x_2 \). If we approximate \( g(x_2) \) by linear functions over all values \( x_2 \) such that \((x_1, x_2) \in P\), that is, \( x_2 \in \text{proj}_2(P) \), then, we can use that inequality within (3). However, since \( g \) is a function over just the variable \( x_2 \) and \( \text{proj}_2(P) \) is some interval of the form \([0, a]\), we can use
Lemma 2 to obtain $a_2, b_2$ such that $a_2x_2 \leq g(x_2) \leq b_2x_2$. Putting all together, we obtain, $a_1y_1 + a_2y_2 \leq f(y_1, y_2)$.

If $P$ were in the second quadrant, then we would use the upper bound of $P$ $u_P(x_2)$ instead of the lower bound $l_P(x_2)$.

The broad idea for the approximation of $f(x)$ follows the above description. The bounding linear functions are constructed by using the bounds on a partial derivative of $f$, and recursively, bounding linear functions on another function that has one less variable. We summarize the recursive definitions of the functions and the bounds. Consider the po-polyhedron $P$ in ordered-bounded form, we proceed to construct a series of functions and points which depend on three elements: the function $f$, the lower and upper bound functions $l_P$ and $u_P$, and the orthant containing $P$. These functions are, in total, $2n$ functions constructed using an iterative method.

**Definition 9.** Let $f$ be an $n$-dimensional function and $P$ a compact $n$-dimensional polyhedron. The maximal functions on $f$ and $P$, denoted by $g_k$ and $h_k$ for $k \in \{1, \ldots, n\}$ are defined as follows:

$$g_1(x) = h_1(x) = f(x) \text{ for } x \in P \text{ and for } k \in \{2, \ldots, n\}$$

$$C1 \equiv \text{proj}_{k-1}(P) \subseteq \mathbb{R}_{\geq 0}, \text{ and } C2 \equiv \text{proj}_{k-1}(P) \subseteq \mathbb{R}_{\leq 0}.$$

$$g_k(x_k) = \begin{cases} g_{k-1}(l_P(x_k), x_k) - a_{g_{k-1}}l_P(x_k) & \text{if } C1 \\ g_{k-1}(u_P(x_k), x_k) - b_{g_{k-1}}u_P(x_k) & \text{if } C2 \end{cases}$$

$$h_k(x_k) = \begin{cases} h_{k-1}(l_P(x_k), x_k) - b_{h_{k-1}}l_P(x_k) & \text{if } C1 \\ h_{k-1}(u_P(x_k), x_k) - a_{h_{k-1}}u_P(x_k) & \text{if } C2 \end{cases}$$

for $x_k \in \text{proj}_{k-1}(P)$, where $a_{g_k}, b_{g_k}, a_{h_k}$ and $b_{h_k}$ are such that $a_{g_k} \leq \frac{\partial g_k}{\partial x_k}(x_k) \leq b_{g_k}$ and $a_{h_k} \leq \frac{\partial h_k}{\partial x_k}(x_k) \leq b_{h_k}$ for every $x_k \in \text{proj}_{k-1}(P)$ with $k \in \{1, \ldots, n\}$.

**Definition 10.** The maximal bounds on $f$ and $P$ are denoted by $a = (a_1, \ldots, a_n)$ and $b = (b_1, \ldots, b_n)$, where the coordinates $a_k$ and $b_k$, for every $k \in \{1, \ldots, n\}$, are defined as follows:

$$a_k = \begin{cases} a_{g_k} & \text{if } \text{proj}_k(P) \subseteq \mathbb{R}_{\geq 0} \\ b_{g_k} & \text{if } \text{proj}_k(P) \subseteq \mathbb{R}_{\leq 0} \end{cases}$$

$$b_k = \begin{cases} b_{h_k} & \text{if } \text{proj}_k(P) \subseteq \mathbb{R}_{\geq 0} \\ a_{h_k} & \text{if } \text{proj}_k(P) \subseteq \mathbb{R}_{\leq 0} \end{cases}$$

Next, we present the theoretical result that states the relation between the maximal functions and maximal bounds.

**Lemma 3.** Let $f$ be an $n$-dimensional real-valued function of class $C^0(P)$ where $0$ is an equilibrium point and $P$ is a compact po-polyhedron. Let $g_k, h_k$ be maximal functions on $f$ and $P$ for every $k \in \{1, \ldots, n\}$, and $a, b \in \mathbb{R}^n$ be maximal bounds on $f$ and $P$. Then, $(a_k, x_k) \leq g_k(x_k)$ and $h_k(x_k) \leq (b_k, x_k)$ for every $x_k \in \text{proj}_{k-1}(P)$.

We use Lemma 3 to provide a proof of Theorem 2.

C. Abstraction algorithm

In this section, we present the main algorithm for the computation of the matrices which define the abstract linear inclusion of a single nonlinear system. Algorithm 1 takes as input an $n$-dimensional vector-valued function $f$ and a po-polyhedron $P \subset \mathbb{R}^n$, and outputs two $n$-dimensional real matrices $A$ and $B$ such that $Ax \leq f(x) \leq Bx$ for every $x \in P$. The function $\text{BOUNDFUNCTIONS}$ on $P$ computes the functions involved in the ordered-bounded representation of $P$ in Definition 1. The function $\text{DIFFERENTIAL}(g, k)$ computes the partial derivative of the function $g$ with respect to the variable $x_k$. The function $\text{PROJECTION}(P, k)$ computes the projection of polyhedron $P$ over $x_k$, while $\text{POSTPROJECTION}(P, k)$ computes the projection of $P$ over the variables $x_k, \ldots, x_n$.

VII. Computational details

In this section, we discuss some computational aspects that arise in the implementation of the hybridization method in Algorithm 1. The linear inclusion hybridization procedure has been implemented in SageMath 7.6 [15] which uses a Python-based language. We use Parma Polyhedra Library (PPL) [16] for representation and manipulation of convex polyhedra and the Python package Scipy for solving optimization problems over multidimensional functions. NetworkX is a Python package intended to create and manipulate graphs, which is required in the construction of the nonlinear and linear inclusion switched systems.

Algorithm 1 requires the construction of upper and lower bound functions in the construction of the ordered-bounded representation of a polyhedral set, that is specified using a conjunction of linear constraints. This is achieved by considering the facets bounding the polyhedra and determining which of them correspond to upper/lower bounds. This construction requires the transformation of polyhedral constraints and can be accomplished using the Parma Polyhedra Library (PPL). The projections of polyhedral sets required by the algorithm can also be computed using PPL.

The maximal functions and their partial derivatives can be computed, for instance, using a library for symbolic calculus integrated into the sagemath software. A critical step of the algorithm is to solve the optimization problems over the maximal functions, which requires, in general, the use of nonlinear optimization algorithms. We have used a minimization routine included in the Python library Scipy for scientific computing. This function supports several minimization algorithms and uses, by default, when provided
Algorithm 1 Over-approximation of a nonlinear function

Require: An n-dimensional vector-valued function $f$ and a polyhedral set $P$
Ensure: Two n-dimensional real matrices
1: $l_P, u_P := $ BOUNDFUNCTIONS($P$)
2: INITIALIZE($A, B$)
3: for $f_i$ in $f$ do
4:   $g := f_i; h := f_i$
5:   INITIALIZE($a, b$)
6:   for $k$ in $(1, \ldots, n)$ do
7:     $g' := \text{DIFFERENTIAL}(g, k)$
8:     $h' := \text{DIFFERENTIAL}(h, k)$
9:     $Q := $ POSTPROJECTION($P, k$) \{Projection of $P$ over $x_k, \ldots, x_n$\}
10:   if $\text{PROJECTION}(P, k) \subseteq \mathbb{R}_{\geq 0}$ then
11:     $a_k := \text{MINIMIZE}(g', Q)$
12:     $b_k := \text{MAXIMIZE}(h', Q)$
13:   else
14:     $a_k := \text{MAXIMIZE}(g', Q)$
15:     $b_k := \text{MINIMIZE}(h', Q)$
16:   end if
17:   $a := \text{APPEND}(a_k)$
18:   $b := \text{APPEND}(b_k)$
20: end for
21: $A.$APPEND($a$)
22: $B.$APPEND($b$)
26: end for
27: return $A, B$

Constraints, the Sequential Least SQares Programming (SLSQP). The input to the function is the objective function, an initial point and a set of constraints. We note that the optimal values returned are sensitive to the initial point provided, and hence, will play a crucial role in the precision of the overall approximation methods.

Finally, the linear inclusion switched system constructed is abstracted into a switched system with polyhedral inclusion dynamics by using the results in [7], which are integrated into the AVERIST [12] software tool. Moreover, AVERIST is used for stability analysis of the switched system with polyhedral inclusions.

VIII. EXAMPLES

In this section, we present examples of single and switched nonlinear dynamical systems and illustrate the construction of linear inclusion dynamics which overapproximates such nonlinear dynamics. We summarize computational times in seconds.

Example 2. Consider a normalized pendulum model, which corresponds to a 2-dimensional nonlinear dynamical system of the form:

$$\begin{align*}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= -\sin x_1 - x_2
\end{align*}$$

Figure 3a shows the phase portrait for this system, where executions of the system are depicted in red. We will focus on the dynamical system restricted to the region $R = [-1,1] \times [-1,1]$. The goal is to define a linear inclusion dynamical system whose set of executions contains the executions of the nonlinear system in the region $R$. First, consider a planar partition which consists of the four quadrants, $P = \{Q_1, Q_2, Q_3, Q_4\}$. The quadrants intersected with the region $R$ result in a set of polyhedral sets, denoted by $P_1, P_2, P_3, P_4$, respectively. These polyhedral sets are depicted in Figure 4a. Figure 4b shows executions of the restricted dynamical system to the first quadrant. Next, consider the following 2-dimensional functions, $f_1(x_1, x_2) = x_2$ and $f_2(x_1, x_2) = -\sin x_1 - x_2$, which correspond to the right hand side of Equation 4. We denote $f(x) = (f_1(x_1, x_2), f_2(x_1, x_2))$. Then, we compute the matrices $A$ and $B$ for each $P_i$ with $i \in \{1, \ldots, 4\}$ such that $Ax \leq f(x) \leq Bx$ for all $x \in [0,1] \times [0,1] \cap P_i$. Function $f_1$ is already linear; so no further computation is required for it. Next, we consider the function $f_2$. After computation, we obtain the $-x_1 - x_2 \leq f_2(x) \leq -0.54x_1 - x_2$ for $(x_1, x_2) \in P_1$, $-0.54x_1 - x_2 \leq f_2(x) \leq -x_1 - x_2$ for $(x_1, x_2) \in P_2$, $-0.54x_1 - x_2 \leq f_2(x) \leq -x_1 - x_2$ for $(x_1, x_2) \in P_3$ and $-x_1 - x_2 \leq f_2(x) \leq -0.54x_1 - x_2$ for $(x_1, x_2) \in P_4$. An hybridized switched system is defined by the partition $P$ and the functions $F_i(x)$ obtained from the previous inequalities, that is $F_1(x) = \{y \in \mathbb{R}^2 : -x_1 - x_2 \leq -0.54x_1 - x_2 \}, F_2(x) = \{y \in \mathbb{R}^2 : -0.54x_1 - x_2 \leq f_2(x) \leq -x_1 - x_2 \}, F_3(x) = \{y \in \mathbb{R}^2 : -0.54x_1 - x_2 \leq f_2(x) \leq -x_1 - x_2 \}$ and $F_4(x) = \{y \in \mathbb{R}^2 : -0.54x_1 - x_2 \leq f_2(x) \leq -x_1 - x_2 \}$. Then, the hybridized switched system is analyzed by AVERIST and stability cannot be established. A finer polyhedral partition is used for constructing a new hybridized switched system. This polyhedral partition is obtained by partitioning the quadrants with the
linear equalities \( x_1 - x_2 = 0 \) and \( x_1 + x_2 = 0 \). For the new hybridized switched system AVERIST establishes stability.

Example 3. Next, we consider the following stable nonlinear system \( S \),
\[
\begin{align*}
    \dot{x}_1 &= -x_1 + x_1 x_2 \\
    \dot{x}_2 &= -x_2
\end{align*}
\]

This system does not admit a polynomial Lyapunov function [17]. Figure 3b shows the phase portrait of system \( S \). We have considered the polyhedral partition \( \mathcal{P} = \{Q_1, Q_2, Q_3, Q_4\} \), where \( Q_i \) correspond to the \( i \)-th planar quadrant, and the region for approximation to be \( R = [-1,1] \times [-1,1] \). We have abstracted the nonlinear dynamics at each planar quadrant, and constructed the linear inclusion switched system \( \mathcal{H}(S, R, \mathcal{P}) \), which has been over-approximated by a polyhedral switched system and evaluated by the stability analysis tool AVERIST, which has established stability of the polyhedral switched system despite the inexistence of a polynomial Lyapunov function. Therefore, we conclude stability of the initial system with respect to the origin.

Example 4. Consider a system \( S \) switching between the nonlinear dynamics described in Example 2 and Example 3. The system evolves in the upper quadrants by following the dynamics in Example 3 and in the lower quadrants by following the dynamics in Example 2. We restrict the nonlinear hybridization procedure to the polyhedral region \( R = [-1,1] \times [-1,1] \) as in previous examples, and consider the polyhedral partition \( \mathcal{P} = \{Q_1, Q_2, Q_3, Q_4, Q_5, Q_6\} \) where \( Q_1 \) and \( Q_2 \) correspond to the first and second quadrants, respectively. The sets \( Q_3 \) and \( Q_4 \) are the polyhedral sets obtained by partitioning the third quadrant with the linear equality \( x_1 - x_2 = 0 \), while the the sets \( Q_5 \) and \( Q_6 \) are obtained by partitioning the fourth quadrant with the linear equality \( x_1 + x_2 = 0 \). The nonlinear switched system is abstracted to the linear inclusion switched system \( \mathcal{H}(S, R, \mathcal{P}) = (\mathcal{F}, \mathcal{J}) \), where \( \mathcal{F} = \{F_1, F_2, F_3, F_4, F_5, F_6\} \) such that
\[
\begin{align*}
    F_1(x_1, x_2) &= \{(y_1, y_2) \in \mathbb{R}^2 : -x_1 \leq y_1 \leq 0, y_2 = -x_2\}, \\
    F_2(x_1, x_2) &= \{(y_1, y_2) \in \mathbb{R}^2 : 0 \leq y_1 \leq -x_1, y_2 = -x_2\}, \\
    F_3(x_1, x_2) &= \{(y_1, y_2) \in \mathbb{R}^2 : y_1 = x_2, -0.54x_1 - x_2 \leq y_2 \leq -x_1 - x_2\}, \\
    F_4(x_1, x_2) &= \{(y_1, y_2) \in \mathbb{R}^2 : y_1 = x_2, -x_1 - x_2 \leq y_2 \leq -x_1 - x_2\}, \\
    F_5(x_1, x_2) &= \{(y_1, y_2) \in \mathbb{R}^2 : y_1 = x_2, -x_1 - x_2 \leq y_2 \leq -x_1 - x_2\}, \\
    F_6(x_1, x_2) &= \{(y_1, y_2) \in \mathbb{R}^2 : y_1 = x_2, -x_1 - x_2 \leq y_2 \leq -x_1 - x_2\}.
\end{align*}
\]

One of the popular design techniques is to design PID

\[
(I + ml^2)\ddot{\theta} - gml\sin \theta - mlu \cos \theta = 0,
\]

where \( m \) is the mass of the rod, \( l \) is the length of the pendulum, \( g \) is the gravitational acceleration constant, \( \theta \) is the angular displacement of the rod from the vertical position and \( I \) is the moment of inertia. The input \( u \) is the cart acceleration and needs to be set for stabilization.

Let us fix the parameter values as \( m = 0.5 \text{ kg} \), \( l = 0.2 \text{ m} \) and \( g = 9.8 \text{ m/s}^2 \), and denote \( x_1, x_2 \) and \( \theta \). Then, we describe the evolution of the system by the following nonlinear dynamics in a state-space form (where the derivatives appear only on the left hand side):
\[
\begin{align*}
    \dot{x}_1 &= x_2 \\
    \dot{x}_2 &= -42.61 \sin x_1 + 4.35w \cos x_1
\end{align*}
\]
controllers based on a linearization of the non-linear system. P is the simplest of them which uses proportional values of the difference between the state and the equilibrium point, while D includes a derivative term. P is more efficient in terms of real-time implementation since it does not require the computation of the derivatives, and is easy to adjust for good system performance. However, it is not always possible to construct a P controller that stabilizes a non-linear system. For instance, we cannot find a stabilizing P controller for the inverted pendulum, but we were able to find a stabilizing PD controller. To enhance the performance, we can design a switching controller that corresponds to a P controller in the first and third planar quadrants and to the previously designed PD controller in the other two quadrants. Observe that the second and fourth quadrants correspond to the rod rotating counterclockwise whereas the first and third quadrants correspond to the rod rotating clockwise. We design the P controller and the PD controller by choosing appropriate gains. More precisely, we design a PD controller with gains $k_p = 5$ and $k_d = -1.839$ that renders the system of Equations 5 to be:

$$
\begin{align*}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= -42.61 \sin x_1 + 21.75 x_1 \cos x_1 - 8 x_2 \cos x_1
\end{align*}
$$

We design the P controller for the case of clockwise rotation with gain $k_p = 5$, and it results in the following unstable dynamical system:

$$
\begin{align*}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= -42.61 \sin x_1 + 21.75 x_1 \cos x_1
\end{align*}
$$

Our approach can be applied to algorithmically evaluate stability not only for the case of a single PD controller (Equation 6) but for the switching controller (Equation 6 and Equation 7). Stability is deduced in both cases.

We provide technical details for the analysis of the switched system $S = (P, F)$ where $P = (P_1, P_2, P_3, P_4)$ is a polyhedral partition with $P_i$ corresponding to the $i$-th planar quadrant; and $F = \{F_1(x), F_2(x), F_3(x), F_4(x)\}$ with $F_1(x, x_2) = (x_2, -42.61 \sin x_1 + 21.75 x_1 \cos x_1)$ and $F_2(x_1, x_2) = (x_2, -42.61 \sin x_1 + 21.75 x_1 \cos x_1 - 8 x_2 \cos x_1)$. We have considered the polyhedral refinement partition $Q = (Q_1, Q_2, Q_3, Q_4, Q_5, Q_6, Q_7, Q_8)$, where $Q_i$ correspond to the regions delimited by the hyperplanes $\{(x_1, x_2) : x_1 = 0\}$, $\{(x_1, x_2) : x_2 = 0\}$, $\{(x_1, x_2) : 3x_1 + x_2 = 0\}$ and $\{(x_1, x_2) : 10x_1 - x_2 = 0\}$; and the region for approximation $R = [-1, 1] \times [-1, 1]$. We have abstracted the nonlinear dynamics in each polyhedral region $Q_i \in Q$, and constructed the linear inclusion switched system $H(S, R, Q) = (Q, G)$, where $G = \{G_1, G_2, G_3, G_4, G_5, G_6, G_7, G_8\}$ such that $G_1(x_1, x_2) = \{(y_1, y_2) \in \mathbb{R}^2 : y_1 \geq x_2, y_1 \leq x_2, y_2 \geq -20.9x_1 - 5y_2 \leq -104.3x_1\}$, $G_2(x_1, x_2) = \{(y_1, y_2) \in \mathbb{R}^2 : y_1 \geq x_2, y_1 \leq x_2, 5y_2 \geq -1043x_1\}$, $G_3(x_1, x_2) = \{(y_1, y_2) \in \mathbb{R}^2 : y_1 \geq x_2, y_1 \leq x_2, 5y_2 \geq -1043x_1\}$, $G_4(x_1, x_2) = \{(y_1, y_2) \in \mathbb{R}^2 : y_1 \geq x_2, y_1 \leq x_2, 5y_2 \geq -1043x_1\}$, $G_5(x_1, x_2) = \{(y_1, y_2) \in \mathbb{R}^2 : y_1 \geq x_2, y_1 \leq x_2, 5y_2 \geq -1043x_1\}$, $G_6(x_1, x_2) = \{(y_1, y_2) \in \mathbb{R}^2 : y_1 \geq x_2, y_1 \leq x_2, 5y_2 \geq -1043x_1\}$, $G_7(x_1, x_2) = \{(y_1, y_2) \in \mathbb{R}^2 : y_1 \geq x_2, y_1 \leq x_2, 5y_2 \geq -1043x_1\}$, $G_8(x_1, x_2) = \{(y_1, y_2) \in \mathbb{R}^2 : y_1 \geq x_2, y_1 \leq x_2, 5y_2 \geq -1043x_1\}$.

<table>
<thead>
<tr>
<th>Examples</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nonlinear hybridization</td>
<td>0.047</td>
<td>0.092</td>
<td>0.167</td>
<td>0.088</td>
<td>0.258</td>
<td>0.125</td>
</tr>
<tr>
<td>Linear hybridization</td>
<td>0.001</td>
<td>0.002</td>
<td>0.003</td>
<td>0.002</td>
<td>0.004</td>
<td>0.004</td>
</tr>
<tr>
<td>Stability verification</td>
<td>–</td>
<td>–</td>
<td>0.430</td>
<td>0.461</td>
<td>0.645</td>
<td>0.415</td>
</tr>
</tbody>
</table>

Table I: Computational times

A summary of computational times for the considered examples is presented in Table I. The first row refers to the example number. Observe that Example 2 has two instances, each of them with a different polyhedral partition, as previously explained. Also Example 5 has two instances, the system with the PD controller (5.1) and the controller switching between the PD and the P controllers (5.2). The second row shows the time for abstracting the nonlinear dynamical systems to linear inclusion dynamical systems. The third row reports the time taken for abstracting the linear inclusion switched systems to polyhedral inclusion switched systems. The last row includes the time spent by VERIST in order to prove stability. In the case of not time appearing, stability was not proven. The experimental results in Table I demonstrate the feasibility of our approach for stability analysis of nonlinear switched systems.

IX. CONCLUSION

In this paper, we presented an abstraction based analysis method for stability that constructs a hybridized linear inclusion switched system from a given nonlinear system. The latter system can be analyzed efficiently using existing techniques, which might require further abstraction [7]. Our framework applies to the general class of non-linear systems $\dot{x} = f(x)$, such that $f$ is differentiable. The computational challenge is in computing the upper and lower bounds on certain partial derivatives, which involve nonlinear functions. Further work will involve building a prototype tool that integrates our hybridization technique with methods for analyzing linear inclusion switched systems, and performing experimental comparison with existing tools. Our examples
already indicate that there are benefits to our approach, since, our method can successfully deduce stability of a non-linear system for which no polynomial Lyapunov function exists. In addition, it will be interesting to investigate heuristics for choosing the partition for hybridization, including counterexample guided abstraction refinement techniques [18], [19].

**APPENDIX**

**A. Proof of Lemma 1**

Given \( P \) convex and closed, we know by definition that there exist \( a_1 = (a_{11}, \ldots, a_{1n}), \ldots, a_m = (a_{m1}, \ldots, a_{mn}) \in \mathbb{R}^n \) and \( b_1, \ldots, b_n \in \mathbb{R} \) such that \( P = \{ x \in \mathbb{R}^n : (a_1, x) + b_1 \geq 0, \ldots, (a_m, x) + b_m \geq 0 \} \). The linear constraints defining \( P \) can be rearranged in order to obtain inequalities with respect to \( x_1 \) of the form \( a_{i1}x_1 \geq -a_{i2}x_2 - \ldots - a_{in}x_n - b_i \) for every \( 1 \leq i \leq m \). We classify them in those with \( a_{i1} \geq 0 \), \( a_{i2} < 0 \) and \( a_{i1} = 0 \). Let us call \( I_{L} \) the index set with \( a_{i1} > 0 \) and \( I_u \) the index set with \( a_{i1} < 0 \). Consider the linear inequalities \( (a_i, x) + b_i \geq 0 \) where \( i \in I_{L} \). We let us denote such linear inequalities after rearrangement as \( f_i(x_2) \leq x_1 \) where \( i \in I_{L} \). We know that every point \( x \in P \) is such that \( \max_{i \in I_{L}} f_i(x_2) \leq x_1 \). Next, we prove that \( \max_{i \in I_{L}} f_i(x_2) \) is a linear piecewise function. Since \( f_i(x_2) \leq x_1 \) with \( i \in I_{L} \) are linear constraints defining the convex polyhedral set \( P \), we know that \( f_k(x_2) \) is smaller or equal to \( f_i(x_2) \) for \( i \in I_{L} \) if there exists \( x_1 \) such that \( (x_1, x_2, \ldots, x_n) \in P \) and \( f_k(x_2) = x_1 \). Therefore, \( \max_{i \in I_{L}} f_i(x_2) = f_k(x_2) \) if \( x_2 \) satisfies \( \bigcap_{1 \leq i \leq m} \{ (a_i, (f_k(x_2), x_2, \ldots, x_n)) + b_i \geq 0 \} \). We denote \( l_p(x_2) \) as \( \max_{i \in I_{L}} f_i(x_2) \). Analogously, we define the piecewise linear function \( u_p(x_2) \) to be \( \min_{i \in I_{L}} f_i(x_2) \). Since there can be linear constraints with \( a_{i1} = 0 \), we ensure that \( P \) is well defined by equating to \( \{ x \in \mathbb{R}^n : l_p(x_2) \leq x_1 \leq u_p(x_2) \} \cap \mathbb{R} \times \text{proj}_{2,\ldots,n}(P) \).

**B. Proof of Proposition 1**

Given a polyhedral set \( P \), the construction of an ordered-bounded representation is inductively achieved. The base case considers \( P_1 = P \), which is a convex and closed polyhedral set. Then, by Lemma 1, there exist \( l_{P}^1 \) and \( u_{P}^1 \), such that \( P_1 = \{ x \in \mathbb{R}^n : l_{P}^1(x_2) \leq x_1 \leq u_{P}^1(x_2) \} \cap \mathbb{R} \times \text{proj}_{2,\ldots,n}(P_1) \). For the case \( i \), with \( 2 \leq i \leq n \), we define \( P_i = \text{proj}_{j,\ldots,n}(P_{i-1}) \). By definition of projection, we know that \( P_i \) is convex and closed. Therefore, by Lemma 1, there exist the functions \( l_{P}^i \) and \( u_{P}^i \) such that \( P_i = \{ x_i \in \mathbb{R}^{n-i+1} : l_{P}^i(x_{i+1}) \leq x_i \leq u_{P}^i(x_{i+1}) \} \cap \mathbb{R} \times \text{proj}_{j,\ldots,n}(P_i) \). Hence, the ordered-bounded representation of the polyhedral set \( P \) is \( \{ l_{P}^1, \ldots, l_{P}^n, u_{P}^1, \ldots, u_{P}^n \} \). Observe that all these operations can be performed algorithmically.

**C. Proof of Lemma 3**

We prove the inequality \((a_k, x_k) \leq g_k(x_k) \) and \( h_k(x_k) \leq (b_k, x_k) \) for every \( x_k \in \text{proj}_{k,\ldots,n}(P) \) by reverse induction on \( k \).

**Base case:** For the case of \( k = n \), we need to show that \((a_n, x_n) \leq g_n(x_n) \) and \( h_n(x_n) \leq (b_n, x_n) \) for every \( x_n \in \text{proj}(P) \). But \( a_n = a_n \) and \( x_n = x_n \), hence, we need to show that \( a_n x_n \leq g_n(x_n) \) and \( h_n(x_n) \leq b_n x_n \) for every \( x_n \in \text{proj}(P) \). The result then follows from Lemma 2.

**Induction step:** Here, we prove that if \((a_{k+1}, x_{k+1}) \leq g_{k+1}(x_{k+1}) \) and \( h_{k+1}(x_{k+1}) \leq (b_{k+1}, x_{k+1}) \) for every \( x_{k+1} \in \text{proj}_{k,\ldots,n}(P) \), then \((a_k, x_k) \leq g_k(x_k) \) and \( h_k(x_k) \leq (b_k, x_k) \) for every \( x_k \in \text{proj}_{k,\ldots,n}(P) \). Consider the ordered-bounded representation of \( P \). We will consider the two different cases, namely, \( \text{proj}_k(P) \subseteq \mathbb{R}_{\geq 0} \) or \( \text{proj}_k(P) \subseteq \mathbb{R}_{\leq 0} \).

**Case \( \text{proj}_k(P) \subseteq \mathbb{R}_{\geq 0} \):** We know, by Definition 9 that
\[
\frac{\partial}{\partial x_k} g_k(x_k) \leq b_{h_k}
\]
\[
\frac{\partial}{\partial x_k} h_k(x_k) \leq b_{h_k}
\]
Let us integrate both inequalities with respect to \( x_k \) from \( l_p(x_{k+1}) \) to \( y_k \), where \( (y_k, x_{k+1}, \ldots, x_n) \in \text{proj}_{k,\ldots,n}(P) \). For Equation 8, we compute
\[
\begin{align*}
\int_{l_p(x_{k+1})}^{y_k} a_{g_k} dx_k \leq \int_{l_p(x_{k+1})}^{y_k} \frac{\partial}{\partial x_k} g_k(x_k) dx_k
\end{align*}
\]
and obtain the following
\[
\begin{align*}
a_{g_k} y_k - a_{g_k} l_p(x_{k+1}) \leq g_k(y_k, x_{k+1}) - g_k(l_p(x_{k+1}), x_{k+1})
\end{align*}
\]
which can be reorganized as
\[
\begin{align*}
a_{g_k} y_k + g_k(l_p(x_{k+1}), x_{k+1}) - a_{g_k} l_p(x_{k+1}) \leq g_k(y_k, x_{k+1})
\end{align*}
\]
By definition of \( g_{k+1}(x_{k+1}) \), the inequality corresponds to
\[
\begin{align*}
a_{g_k} y_k + g_{k+1}(x_{k+1}) \leq g_k(y_k, x_{k+1})
\end{align*}
\]
Then, by considering the induction hypothesis, we obtain
\[
\begin{align*}
a_{g_k} y_k + (a_{k+1}, x_{k+1}) \leq g_k(y_k, x_{k+1})
\end{align*}
\]
Since this inequality holds for any fixed \( y \in P \) and \( a_k = (a_{g_k}, x_{k+1}) \), we have
\[
\begin{align*}
(a_k, x_k) \leq g_k(x_k) \text{ for every } (x_k) \in \text{proj}_{k,\ldots,n}(P).
\end{align*}
\]
Next, for Equation 9, we compute
\[
\begin{align*}
\int_{l_p(x_{k+1})}^{y_k} \frac{\partial}{\partial x_k} h_k(x_k) dx_k \leq \int_{l_p(x_{k+1})}^{y_k} b_{h_k} dx_k
\end{align*}
\]
and obtain the following,
\[ h_k(y_k, x_{k+1}) - h_k(l_P(x_{k+1}), x_{k+1}) \leq b_{hk} y_k - b_{hk} l_P(x_{k+1}), \]
which we reorganize as
\[ h_k(y_k, x_{k+1}) \leq b_{hk} y_k + h_k(l_P(x_{k+1}), x_{k+1}) - b_{hk} l_P(x_{k+1}). \]
By definition of \( h_{k+1}(x_{k+1}) \), the inequality corresponds to
\[ h_k(y_k, x_{k+1}) \leq b_{hk} y_k + h_k(x_{k+1}). \]
Then, by considering the induction hypothesis, we obtain
\[ h_k(y_k, x_{k+1}) \leq b_{hk} y_k + b_{hk} + h_k(x_{k+1}). \]
Since this inequality holds for any fixed \( y \in P \), and \( b_k = (b_{hk}, b_{hk+1}) \), we have
\[ h_k(x_k) \leq \langle b_k, x_k \rangle \text{ for every } x_k \in \text{proj}_{k,...,n}(P). \]

**Case** \( \text{proj}_k(P) \subseteq \mathbb{R}^n \): Analogously, we know, by Definition 9, that
\[
\frac{\partial}{\partial x_k} g_k(x_k) \leq b_{g_k} a_{hk} \leq \frac{\partial}{\partial x_k} h_k(x_k). 
\]
Note that since \( \text{proj}_k(P) \subseteq \mathbb{R}^n \), we need to integrate the inequalities with respect to \( x_k \) from \( y_k \) to \( u_P(x_{k+1}) \), where \( (y_{k}, x_{k+1}, \ldots, x_n) \in \text{proj}_{k,...,n}(P) \). For Equation 10a, we compute
\[
\int_{y_{k}}^{u_P(x_{k+1})} \frac{\partial}{\partial x_k} g_k(x_k) dx_k \leq 
\int_{y_{k}}^{u_P(x_{k+1})} b_{g_k} dx_k
\]
and obtain the following,
\[ g_k(u_P(x_{k+1}), x_{k+1}) - g_k(y_k, x_{k+1}) \leq b_{g_k} u_P(x_{k+1}) - b_{g_k} y_k, \]
which we reorganize as
\[ g_k(u_P(x_{k+1}), x_{k+1}) - b_{g_k} u_P(x_{k+1}) + b_{g_k} y_k \leq g_k(y_k, x_{k+1}). \]
By definition of \( g_{k+1}(x_{k+1}) \), the inequality corresponds to
\[ g_{k+1}(x_{k+1}) + b_{g_k} y_k \leq g_k(y_k, x_{k+1}). \]
Then, by considering the induction hypothesis, we obtain
\[ \langle a_{k+1}, x_{k+1} \rangle + b_{g_k} y_k \leq g_k(y_k, x_{k+1}). \]
Since this inequality holds for any fixed \( y \in P \), and \( a_k = (b_{g_k}, a_{k+1}) \), we have
\[ \langle a_k, x_k \rangle \leq g_k(x_k) \text{ for every } x_k \in \text{proj}_{k,...,n}(P). \]

Next, for Equation 10b, we compute the integral between \( y_k \) and \( u_P(x_{k+1}) \) and obtain the following,
\[ a_{hk} u_P(x_{k+1}) - a_{hk} y_k \leq h_k(u_P(x_{k+1}), x_{k+1}) - h_k(y_k, x_{k+1}), \]
which we reorganize as
\[ h_k(y_k, x_{k+1}) \leq h_k(u_P(x_{k+1}), x_{k+1}) - a_{hk} u_P(x_{k+1}) + a_{hk} y_k. \]
By definition of \( h_{k+1}(x_{k+1}) \), the inequality corresponds to
\[ h_k(y_k, x_{k+1}) \leq h_{k+1}(x_{k+1}) + a_{hk} y_k. \]
Then, by considering the induction hypothesis, we obtain
\[ h_k(y_k, x_{k+1}) \leq \langle b_{k+1}, x_{k+1} \rangle + a_{hk} y_k. \]
Since this inequality holds for any fixed \( y \in P \), and \( b_k = (a_{hk}, b_{k+1}) \), we have
\[ h_k(x_k) \leq \langle b_k, x_k \rangle \text{ for every } x_k \in \text{proj}_{k,...,n}(P). \]
Hence, every \( a_k, b_k \in \mathbb{R}^{n-k+1} \) from Definition 9 are such that \( \langle a_k, x_k \rangle \leq g_k(x_k) \) and \( h_k(x_k) \leq \langle b_k, x_k \rangle \) for every \( x_k \in \text{proj}_{k,...,n}(P) \).

**REFERENCES**


