

Counter-Example Guided Synthesis of Control Lyapunov Functions for Switched Systems.

Hadi Ravanbakhsh and Sriram Sankaranarayanan
Dept. of Computer Science, University of Colorado, Boulder.
firstname.lastname@colorado.edu

Abstract— We investigate the problem of synthesizing switching controllers for stabilizing continuous-time plants. First, we introduce a class of control Lyapunov functions (CLFs) for switched systems along with a switching strategy that yields a closed loop system with a guaranteed minimum dwell time in each switching mode. However, the challenge lies in automatically synthesizing appropriate CLFs. Assuming a given fixed form for the CLF with unknown coefficients, we derive quantified nonlinear constraints whose feasible solutions (if any) correspond to CLFs for the original system.

However, solving quantified nonlinear constraints pose a challenge to most LMI/BMI-based relaxations. Therefore, we investigate a general approach called Counter-Example Guided Inductive Synthesis (CEGIS), that has been widely used in the emerging area of automatic program synthesis. We show how a LMI-based relaxation can be formulated within the CEGIS framework for synthesizing CLFs. We also evaluate our approach on a number of interesting benchmarks, and compare the performance of the new approach with our previous work that uses off-the-shelf nonlinear constraint solvers instead of the LMI relaxation. The results shows synthesizing CLFs by using LMI solvers inside a CEGIS framework can be a computational feasible approach to synthesizing CLFs.

I. INTRODUCTION

The goal of this article is to automatically synthesize continuous-time switching controllers for guaranteed asymptotic stability of a switched polynomial dynamical system. The plant is defined by a continuous-time switched system with continuous state variables and finitely many control modes. The controller can choose a control mode through state-feedback in order to guarantee closed loop stability w.r.t a specified equilibrium point.

The proposed solution is based on adapting Control Lyapunov Functions (CLFs) to provide a switching strategy that guarantees asymptotic stability. A CLF extends a regular Lyapunov function to the controlled setting, where it requires that for each state, there exists a control that causes an instantaneous decrease in the value of the CLF. However, CLFs for switched systems can be quite tricky: for a controller to be realizable, the CLF must guarantee that the switching signal does not always attempt to change modes infinitely often inside a finite time horizon (zenoness). In this paper, we first provide a sufficient condition on CLFs, along with an associated switching strategy that ensures the switching function respects a *minimum dwell time* for each control mode. In other words, we guarantee a minimal time $\tau > 0$, such that once a control mode is chosen by the controller at time t , it remains chosen during the interval

$[t, t + \tau]$. This requirement is essential for the controller to be implementable.

However, the main challenge is to arrive at such CLFs in the first place. To do so, we use a *template* CLF that is simply a parametric form of the desired CLF with unknown coefficients. We wish to solve for these coefficients to find if a CLF with the given template exists. We find that this process yields nonlinear feasibility problems that have alternating \exists and \forall quantifiers. This is in direct contrast with a standard optimization problems that simply involve \exists quantifiers. The presence of nonlinear (semi-algebraic) constraints is yet another complication.

To get around the quantification problem, we employ a framework called CounterExample Guided Inductive Synthesis (CEGIS) that was originally proposed to “complete” unknown parameters inside partial programs (termed sketches) so that the resulting programs satisfy some correctness properties [19]. In this paper, we adapt CEGIS to the problem of controller synthesis to solve the resulting quantified constraints.

Another challenge lies in dealing with nonlinear (semi-algebraic) constraints. Our previous work used off-the-shelf nonlinear constraint solvers like dReal [18]. However, the resulting procedure is often expensive and fails to complete, even for small systems. In this article, we examine a LMI-based relaxation for the semi-algebraic constraints. We show how the CEGIS-framework can be adapted to use LMI-relaxation for synthesizing CLFs.

We provide an implementation of the CEGIS approach to synthesizing CLFs using the SMT solver Z3 constraint-solver for linear constraints [6] and the CVXOPT [1] solver for LMI constraints. The evaluation suggests our approach can synthesize switching controllers for a number of interesting benchmarks and can solve larger problems in comparison with our previous results. In summary, the contributions of this paper are as follows:

- 1) We present a sufficient condition on CLFs along with a switching strategy which guarantees asymptotic stability as well as non-zeno behavior.
- 2) We adapt the CEGIS algorithm (used to discover CLFs) to use LMI-relaxations, thus significantly improving its performance.
- 3) We provide a detailed experimental evaluation on a set of benchmarks.

A. Related Work

“Correct-by-construction” approaches seek hybrid (switching) controllers from mathematical plant models, wherein the synthesis procedure also guarantees a set of user-defined correctness properties for the closed loop. One approach to the synthesis first constructs a finite abstraction of the system along with a *simulation relation* between the abstract system and the actual system. The simulation relation guarantees that a controller that guarantees a certain class of properties (eg., safety) on the abstract system will also serve to control the original plant model. Then the problem is solved for the abstract system using discrete automata-based synthesis techniques [13], [11], [3]. The problem of zeno-ness is addressed in some of these approaches (eg., [3]) by enforcing a minimum dwell time between mode switches.

The other class of approaches are based on Lyapunov functions. Synthesizing Lyapunov functions is a well-studied problem for polynomial systems. For instance, the conditions on Lyapunov functions have been relaxed using Sum-of-Squares (SOS) programming [14]. However, the problem for synthesizing a CLF is known to be much harder. For control-continuous feedback systems Artstein [2] introduced necessary conditions on CLFs, and then showed that a static feedback law can be extracted from the CLF once it is discovered. However, synthesizing such a CLF is typically formulated as a Bilinear Matrix Inequality (BMI) (e.g. [21]). CLFs have been studied for switched systems, as well, but mainly for switched linear systems. For a survey on these results, we refer the reader to Lin et al. [10].

The problem of zeno behavior roughly corresponds to *chattering*, that is common in approaches such as sliding mode control [9], [5]. However, chattering is dealt with in sliding mode control by providing a smooth feedback control in a small zone surrounding the sliding surface that allows trajectories to approach the sliding mode. It is not entirely clear if the formal properties sought in this paper are necessarily preserved by such a smoothing step.

Recently, we proposed a CLF-based approach to controller synthesis [18] that guarantees a *minimum dwell time* property for *region-stabilization* of switched systems using a counterexample-guided synthesis approach similar (but not identical) to the approach described in this paper. Region stability notions first introduced by Podelski and Wagner, reason about asymptotic convergence of trajectories to a set around the equilibrium rather than the equilibrium itself [16]. In this article, we address asymptotic stability. Furthermore, our previous work used a nonlinear constraint solver (dReal) “out-of-the-box” [7]. Here, we provide substantial performance improvements by formulating a LMI relaxation.

II. PRELIMINARIES

A. Notations

Let \mathbb{N} , \mathbb{Z} , \mathbb{R} and \mathbb{R}^+ denote the sets of natural, integer, real and non-negative real numbers, respectively. Let $\mathbb{R}[\mathbf{x}]$ be set of all polynomials involving variables in \mathbf{x} . The 2-norm of a

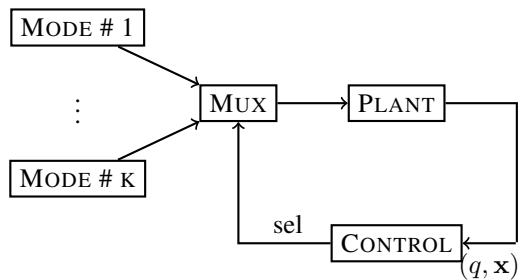


Fig. 1. Model of the closed loop system

vector \mathbf{x} is written $\|\mathbf{x}\|$. The (full dimensional) ball centered around \mathbf{x} with radius r is denoted $\mathcal{B}_r(\mathbf{x})$. Let \mathbf{x}_i is the i^{th} element of vector \mathbf{x} and for a subset $X \subseteq \mathbb{R}^n$, let X_i be it’s projection onto the variable \mathbf{x}_i .

Given a polynomial $p \in \mathbb{R}[\mathbf{x}]$, let $\text{Monos}(p)$ be set of all monomials in p (monomials with non-zero coefficient). Let $\text{Deg}(p)$ be the maximum degree of polynomial p , and $\text{Vars}(p)$ is the set of variables involved in p . For a function $f : \mathbb{R} \rightarrow \mathbb{R}^n$, $f^+(x)$ ($f^-(x)$) denotes its right (resp. left) limit at x . Also $\dot{f}^+(x)$ ($\dot{f}^-(x)$) represents its right (resp. left) derivative at x .

B. System Model

The system of interest consists of a plant model and a continuous feedback switched controller. The plant has a finite number of modes belonging to the set Q , and is modeled with n continuous variables. These variables have different dynamics, depending on the mode of the plant. The controller chooses the mode for the plant, given the current continuous state of the plant and it’s current mode. Fig. 1 shows a schematic view of the closed loop system.

Definition 1 (Plant): A plant is a triple $\Psi(X, Q, f)$ describing the physical environment:

- 1) $X \subseteq \mathbb{R}^n$ is domain of continuous variables (n is the number of continuous state variables).
- 2) Q is a finite set containing (control) modes.
- 3) f is a function, that maps each mode $q \in Q$ to a polynomial vector field $f_q \in \mathbb{R}[\mathbf{x}]^n$

Definition 2 (Controller): Given a plant $\Psi(X, Q, f)$, a controller is a function switch : $Q \times X \rightarrow Q$ that maps current continuous state variable $\mathbf{x} \in X$ and mode $q \in Q$ to next mode \hat{q} .

A trace of such a system is given by functions $\mathbf{x}(\cdot) : \mathbb{R}^+ \rightarrow X$ and $q(\cdot) : \mathbb{R}^+ \rightarrow Q$, which map time to continuous state variables and the discrete mode of the plant, respectively.

$\mathbf{x}(\cdot)$ is a continuous function defined as

$$\mathbf{x}(0) = \mathbf{x}_0 \quad , \quad \dot{\mathbf{x}}^+(t) = f_{q(t)}(\mathbf{x}(t))$$

$q(\cdot)$ is a piecewise constant function with finite or countably infinite set of times $\text{SwitchTimes}(q) = \{t \mid q^-(t) \neq q^+(t)\}$. A trace $(\mathbf{x}(\cdot), q(\cdot))$ is *time-divergent* if for each $\Delta > 0$, the set $[0, \Delta] \cap \text{SwitchTimes}(q)$ is finite.

C. Problem Statement

The goal is to find a switch function that guarantees asymptotic stability of the resulting closed loop around a specified equilibrium \mathbf{x}^* . Since we are considering polynomial dynamical systems, w.l.o.g we assume $\mathbf{x}^* = \mathbf{0}$. In addition to the main specification, we also require the closed-loop system to maintain a minimum dwell time in each mode, as explained earlier.

Given a connected and compact set $P \subseteq X$, the asymptotic stability of the closed loop inside P implies it's (local) Lyapunov stability and the asymptotic convergence of all trajectories starting in P to \mathbf{x}^* [12]. Notice that if $\mathbf{x}(T) = \mathbf{0}$, then $(\forall t > T) \mathbf{x}(t) = \mathbf{0}$ should also hold. Therefore, we assume there is at least one mode $q_0 \in Q$ s.t. $f_{q_0}(\mathbf{0}) = \mathbf{0}$. Also, it is well known that a given set P may not be asymptotically stabilizable but a subset $P^* \subseteq P$ may often be stabilizable.

Problem 1: Given a plant Ψ and region $P \subseteq X$, find a switch function and a region $P^* \subseteq P$ s.t. the closed loop switched system is asymptotically stable w.r.t P^* , while all the traces are time-divergent.

III. CONTROLLER SYNTHESIS

Control Lyapunov functions (CLFs) have been used to stabilize systems with control-continuous feedback [2]. Artstein first showed once a CLF is obtained, how a corresponding feedback law is extracted. First we formally define what is a CLF and which class of CLF can be used for *switched* controllers.

Definition 3: Given a plant Ψ and a region $P \subset X$, a control Lyapunov function (CLF) for the plant w.r.t P is a positive definite function $V : X \rightarrow \mathbb{R}^+$ ($V(\mathbf{0}) = 0$) s.t. $\forall \mathbf{x} \in P \setminus \{\mathbf{0}\}$

$$V(\mathbf{x}) > \alpha(\mathbf{x}) \wedge (\min_{q \in Q} \dot{V}_q(\mathbf{x})) < -\alpha_Q(\mathbf{x}), \quad (1)$$

where α and α_Q are positive definite functions and $\dot{V}_q = (\frac{dV}{d\mathbf{x}})^T f_q$.

Henceforth, we restrict our attention to polynomial CLFs $V(\mathbf{x}) \in \mathbb{R}[\mathbf{x}]$. Given a polynomial CLF V and region P , we associate region P^* to V as $P^* := P \cap \{\mathbf{x} \mid V(\mathbf{x}) < \underline{\beta}(P, V)\}$, where $\underline{\beta}(X, F) := \min_{\mathbf{x} \in \partial X} F(\mathbf{x})$.

Also given a CLF V , the associated set of control functions switch satisfy

$$\text{switch}(q, \mathbf{x}) \in \{\hat{q} \mid \dot{V}_{\hat{q}}(\mathbf{x}) < -\alpha_Q(\mathbf{x})\}$$

In other words, the controller chooses a mode \hat{q} to enforce the decrease of the CLF at all times. However, time-divergence is not guaranteed with this class of functions, and therefore asymptotic stability cannot be guaranteed. To guarantee time-divergent behavior, we can impose a minimal dwell time property. A trace satisfies minimum dwell time property for a dwell time $\delta > 0$ iff

$$(\forall t_1, t_2 \in \text{SwitchTimes}(q)) t_1 \neq t_2 \implies |t_1 - t_2| > \delta$$

How do we find functions switch s.t. all the resulting closed loop behaviors satisfy this property?

A. Non-Zero CLF

In this section, we define a large class of CLFs that can be used to synthesize controllers with guaranteed minimum dwell time. Before introducing this class of CLFs, we need to define another condition.

Definition 4 (ϕ -boundedness): Given functions $p, \phi : X \rightarrow \mathbb{R}$, p is said to be ϕ -bounded iff for every bounded region $S \subset X$ there exists a constant Λ_S s.t. $(\forall \mathbf{x} \in S) p(\mathbf{x}) \leq \Lambda_S \phi(\mathbf{x})$.

Example 1: Consider $\phi(x, y) : x^2 + y^2$. Any multivariate polynomial $p(x, y)$ whose lowest degree terms have degree at least 2 is ϕ -bounded. Examples include $x^2 + 2x^3 + 3xy$, xy , and $x^6 - 3y^3$. On the other hand, the function $p(x, y) = x + y$ is not ϕ -bounded since no bound of the form $p(x, y) \leq \Lambda_S(x^2 + y^2)$ when S is taken to be a region containing $(0, 0)$. Similarly, the function $3 + x$ is not ϕ -bounded.

Definition 5 (Non-Zero CLF): A CLF is said to be non-zero iff there exist constants $\epsilon_q > 0$ and positive (definite) functions $\phi_q : X \rightarrow \mathbb{R}$ s.t.

$$\ddot{V}_q(\mathbf{x}) \text{ is } \phi_q\text{-bounded,} \quad (2)$$

$$\dot{\phi}_q(\mathbf{x}) \text{ is } \phi_q\text{-bounded, and} \quad (3)$$

$$(\forall \mathbf{x} \in P \setminus \{\mathbf{0}\}) (\exists q \in Q) \dot{V}_q(\mathbf{x}) < -\epsilon_q \phi_q(\mathbf{x}), \quad (4)$$

where $\ddot{V}_q(\mathbf{x}) = (\frac{d\dot{V}_q}{d\mathbf{x}})^T f_q(\mathbf{x})$ and $\dot{\phi}_q(\mathbf{x}) = (\frac{d\phi_q}{d\mathbf{x}})^T f_q(\mathbf{x})$. Informally, the goal is to make sure not only \dot{V}_q is negative definite, but also is smaller than a class of negative (definite) functions. Now we explain how such property helps to guarantee min-dwell time property.

Assume there exists a non-zero CLF V and let a class of functions switch associated to V be defined as

$$\text{switch}(q, \mathbf{x}) := \begin{cases} \hat{q} & \left(\begin{array}{l} \dot{V}_{\hat{q}}(\mathbf{x}) \geq \eta \wedge \mathbf{x} \in P \\ \wedge \dot{V}_{\hat{q}}(\mathbf{x}) \leq -\epsilon_{\hat{q}} \phi_{\hat{q}}(\mathbf{x}) \end{array} \right) \\ q & \text{otherwise} \end{cases} \quad (5)$$

wherein $\eta := -\frac{\epsilon_q \phi_q(\mathbf{x})}{\lambda}$ for a chosen scale constant $\lambda > 1$. In other words, rather than switch when the CLF $\dot{V}_q(\mathbf{x}) = 0$, we force the system to switch when $\dot{V}_q(\mathbf{x}) \geq \eta$. We also force the system to switch to a mode \hat{q} for which $\dot{V}_{\hat{q}}(\mathbf{x}) \leq -\epsilon_{\hat{q}} \phi_{\hat{q}}(\mathbf{x})$. The definition of a non-zero CLF guarantees that such a mode \hat{q} will exist.

The key observation here is that the constraints on $\ddot{V}_q, \dot{V}_q, \dot{\phi}_q$ altogether guarantee that when the controller switches at time t_1 , the controller need not switch again in interval $[t_1, t_1 + \delta]$ for some fixed $\delta > 0$ (i.e. $\dot{V}_{\hat{q}}(\mathbf{x}(t)) < \eta$ for all $t \in [t_1, t_1 + \delta]$). A bound for δ is given directly in the proof of the following proposition.

Theorem 1: Given regions P , a plant Ψ and a non-zero CLF $V(\mathbf{x})$, let P^* be the associated region for V w.r.t P . Given $\mathbf{x}(0) \in P^*$, a switching function that admit the description of Equation (5) results in a system which satisfies the following properties.

- 1) all the traces of the system are time-divergent

- 2) P^* is a positive invariant.
 3) system is asymptotically stable w.r.t P^*
 A proof is provided in the Appendix.

B. Implementation

Once a non-zero CLF is found, the controller can be implemented in many ways. We can implement an operational amplifier circuit that selects the appropriate mode by computing $\phi(\mathbf{x})$ and $\dot{V}_q(\mathbf{x})$ from the state feedback \mathbf{x} . Such a circuit will not need to know the minimum dwell time: however, the minimum dwell time provides us with a guideline on the maximum delay permissible.

Another approach is to find an under-approximation of min-dwell time $\underline{\delta}$ and use a discrete time controller that change the modes every $\underline{\delta}$ time units. Yet another software-based solution is to use a model predictive control scheme: the controller switches to a mode q at time t_s given $\mathbf{x}(t_s)$ ($\dot{V}_q(\mathbf{x}(t_s)) < -\epsilon_q \phi_q(\mathbf{x}(t_s))$). Also, the controller predicts the first time instance $t_f > t_s$ s.t. $\dot{V}_q(\mathbf{x}(t_f)) \geq -\frac{\epsilon_q \phi_q(\mathbf{x}(t_f))}{\lambda}$. Then the controller sets a wake up timer for time $t = t_f$ and re-evaluates at that point. The minimum dwell time provides a design guideline to the scheduler on the shortest possible wake up time t_f .

IV. DISCOVERING CONTROL LYAPUNOV FUNCTIONS

Thus far, the problem of controller synthesis has been reduced to problem of finding a non-zero CLF. First, a template polynomial is chosen for function V . More precisely $V(\mathbf{c}, \mathbf{x}) = \sum_{i=1}^m \mathbf{c}_i m_i(\mathbf{x})$ is a polynomial with fixed monomials $m_i(\mathbf{x})$ and unknown coefficients $\mathbf{c} \in R^m$. Second, appropriate values for ϵ_q and ϕ_q (for all $q \in Q$) are chosen. In particular, finding positive (definite) functions ϕ_q s.t. Equations (2) and (3) hold is not straightforward. We consider a simple class of positive (definite) functions of the form $\phi_q(\mathbf{x}) = \sum_{i=1}^n \mathbf{x}_i^{2d_{i,q}}$, where $d_{i,q} \in \mathbb{N}$. Then, we use the following theorem to find proper values for $d_{i,q}$ s.t. $\dot{\phi}_q$ and \ddot{V}_q are ϕ_q -bounded.

Theorem 2: Given a function $\phi(\mathbf{x}) = \sum_{i=1}^n \mathbf{x}_i^{2d_i}$ and a function $p : X \rightarrow R$ is ϕ -bounded if

$$(\forall m \in \text{Monos}(p)) (\forall i) 2d_i \leq \text{Deg}(m). \quad (6)$$

A proof is provided in the Appendix. By this theorem, one can find all possible functions ϕ_q s.t. Equations (2) and (3) hold, because the process of finding these functions depends only on the possible monomial terms in V , and not on their coefficients.

Example 2 (Choosing ϕ_q for a System): Consider a switched system with three continuous variables x, y and z and two modes q_1 and q_2 with dynamics:

$$q_1 \begin{cases} \dot{x} = -y \\ \dot{y} = z \\ \dot{z} = 1 \end{cases} \quad q_2 \begin{cases} \dot{x} = y^2 \\ \dot{y} = -x^3 - y^3 \\ \dot{z} = -z \end{cases}$$

Assuming $V([x \ y \ z]^T) = c_1 x^2 + c_2 y^2 + c_3 z^2$, $\ddot{V}_{q_1}([x \ y \ z]^T) = 2c_1(y^2 - xz) + 2c_2(z^2 + y) + 2c_3$ and $\phi_{q_1}([x \ y \ z]^T) = y^0 + x^0 + z^0$ satisfies both Equations (2) and (3) and it is a

proper ϕ_{q_1} . For mode q_2 , one can choose many ϕ_{q_2} functions as well. For example $\phi_{q_2}([x \ y \ z]^T) = x^4 + y^4 + z^2$ is a possible solution.

In addition to ϵ_q and ϕ_q , we fix a positive definite function $\alpha(\mathbf{x})$. Furthermore, we assume \mathbf{c} belongs to a bounded set $C_0 \subset R^m$ (Often $C_0 : [-1, 1]^m$). Now, the problem is to find unknown coefficients \mathbf{c} s.t. V is a non-zero CLF. In other words, we want to solve problem below

$$(\exists \mathbf{c} \in C_0) (\forall \mathbf{x} \in P \setminus \{0\}) \left(\begin{array}{l} V(\mathbf{x}) > \alpha(\mathbf{x}) \wedge \\ (\exists q \in Q) \dot{V}_q(\mathbf{x}) < -\epsilon_q \phi_q(\mathbf{x}) \end{array} \right) \quad (7)$$

Note that, if the formula above is feasible (satisfiable), then the existential quantifier ($\exists \mathbf{c} \in C_0$) yields us a solution \mathbf{c} that can be used to instantiate the CLF. First, we use an LMI-based relaxation of the relevant polynomial problems. This is done following the standard approach [8], [15]. Briefly, let \mathbf{m} represent a $m \times 1$ vector of monomials. A polynomial $p(\mathbf{x})$ can be written as $\langle Q, \mathbf{m}\mathbf{m}^t \rangle$ where Q is a symmetric $m \times m$ matrix and $\langle A, B \rangle$ denotes $\text{tr}(A \times B)$. Next, we relax $\mathbf{m}\mathbf{m}^t$ by a matrix $Z \succeq 0$ where $\text{rank}(Z) = 1$. The constraint $\mathbf{x} \in P$ is rewritten as the constraint $Z \in \hat{P}$. Typically, P is given as a interval constraint. Therefore \hat{P} is itself an interval over matrices that represent the lower and upper bounds of each monomial in $\mathbf{m}\mathbf{m}^t$. Finally, the rank constraint is thrown out, and often replaced by a ‘‘low-rank promoting’’ constraint or objective.

Therefore, the constraints in (7) are rewritten to yield a (mixed linear + LMI cone) constraint of the following form:

$$(\exists \mathbf{c} \in C_0) (\forall Z) (Z \succeq 0 \wedge Z \in \hat{P} \wedge \langle G, Z \rangle > 0) \Rightarrow \left(\begin{array}{l} \langle F(\mathbf{c}) - G, Z \rangle > 0 \wedge \\ (\exists q \in Q) \langle F_q(\mathbf{c}) - G_Q, Z \rangle > 0 \end{array} \right) \quad (8)$$

Here, $\langle G, Z \rangle$ ($\langle G_Q, Z \rangle$) is the relaxed version of $\alpha(\mathbf{x})$ ($\alpha_Q(\mathbf{x})$) and $F(\mathbf{c}) = \sum_{j=0}^k c_j F_j$ represents a matrix whose entries are linear in \mathbf{c} , and similarly for $F_q(\mathbf{c})$. As such the form above is not easy to solve using existing methods: the constraints are bilinear and contain disjunctions. To solve this $\exists \forall$ formula we employ CEGIS framework [18].

Overview of the CEGIS framework: At a high level, CEGIS focuses on formulae of the form

$$(\exists \mathbf{x} \in A) (\forall \mathbf{y} \in B) \psi(\mathbf{x}, \mathbf{y}).$$

The algorithm is iterative and at any iteration maintains a finite set of witnesses $\hat{B}_{\{i\}} = \{\mathbf{b}_1, \dots, \mathbf{b}_l\}$.

Initially $\hat{B}_{\{0\}}$ is a some finite subset consisting of samples from the set B . At each iteration, we consider the following two steps:

- 1) Choose a candidate $\mathbf{a}_{\{i\}}$ by solving the problem:

$$\mathbf{a}_{\{i\}} := \text{find } \mathbf{x} \in A \text{ s.t. } \bigwedge_{\mathbf{b}_j \in \hat{B}_{\{i\}}} \psi(\mathbf{x}, \mathbf{b}_j). \quad (9)$$

Note that the inner \forall quantifier is replaced by a finite conjunction and we have \mathbf{y} variables in ψ instantiated. If the problem is feasible and $\mathbf{a}_{\{i\}} \in A$ is obtained, we

move to the next step in the iteration. Otherwise, we declare failure of the overall procedure.

- 2) Next, check the candidate $\mathbf{a}_{\{i\}}$ by checking the formula: $(\forall \mathbf{y} \in B)\psi(\mathbf{a}_{\{i\}}, \mathbf{y})$, or equivalently if its negation is feasible:

$$\text{find } \mathbf{y} \text{ s.t. } \neg\psi(\mathbf{a}_{\{i\}}, \mathbf{y}). \quad (10)$$

If the formula above is infeasible, then we have found the required answer $\mathbf{a}_{\{i\}}$ for the original problem. Otherwise, we find a $\mathbf{b}_{\{i+1\}}$ such that $\neg\psi(\mathbf{a}_{\{i\}}, \mathbf{b}_{\{i+1\}})$ succeeds. We now set $\hat{B}_{\{i+1\}} := \hat{B}_{\{i\}} \cup \{\mathbf{b}_{\{i+1\}}\}$.

Note that adding $\mathbf{b}_{\{i+1\}} \in B_{\{i+1\}}$ ensures that $\mathbf{a}_{\{i\}}$ is never chosen again in any future iteration. It can also eliminate all other previously unexamined values of $\mathbf{a} \in X$ that also fail $\psi(\mathbf{a}, \mathbf{b}_{\{i+1\}})$.

Applying CEGIS Procedure Given the disjunctive formula from Eq. (8), which will be written as

$$(\exists \mathbf{c} \in C_0) (\forall Z) (Z \succeq 0 \wedge Z \in \hat{P} \wedge \langle G, Z \rangle > 0) \Rightarrow \Psi(\mathbf{c}, Z).$$

The CEGIS procedure works with a set $\hat{B}_{\{i\}} := \{Z_{\{1\}}, \dots, Z_{\{k_i\}}\}$, wherein each $Z \in \hat{B}_{\{i\}}$ satisfies the constraints $Z \succeq 0$, $Z \in \hat{P}$ and $\langle G, Z \rangle > 0$.

- 1) Find a value $\mathbf{c}_{\{i\}} \in C_0$ that satisfies:

$$\mathbf{c}_{\{i\}} := \text{find } \mathbf{c} \text{ s.t. } \Psi(\mathbf{c}, Z_{\{1\}}) \wedge \dots \wedge \Psi(\mathbf{c}, Z_{\{k_i\}}).$$

Plugging in $Z = Z_{\{1\}}, \dots, Z_{\{k_i\}}$ yields a system of *disjunctive linear constraints* over \mathbf{c} . While solving constraints is NP-hard, recent progress in SAT modulo-theory (SMT) solvers has yielded efficient implementations such as Z3 can handle quite large instances of disjunctive linear constraints [6].

- 2) If $\mathbf{c}_{\{i\}}$ is found, we next check the feasibility of $\mathbf{c}_{\{i\}}$ by successively solving, separately, a series of mixed cone constraints:

$$(1) \quad \langle F(\mathbf{c}_i) - G, Z \rangle \leq 0, \langle G, Z \rangle > 0, Z \in \hat{P}, Z \succeq 0$$

$$(2) \quad \bigwedge_{q \in Q} \langle F_q(\mathbf{c}_i) - gcq, Z \rangle \leq 0$$

$$\langle G, Z \rangle > 0, Z \in \hat{P}, Z \succeq 0$$

With $\mathbf{c} = \mathbf{c}_{\{i\}}$, the bilinearity is now avoided. If any of these constraints are feasible, we obtain a new witness $Z_{\{i+1\}}$ that is added to $B_{\{i+1\}}$. Otherwise, the constraints are infeasible and we have found our required $\mathbf{c}^* = \mathbf{c}_{\{i\}}$.

The process is iterated until we find parameters $\mathbf{c} = \mathbf{c}^*$ or fail to find a candidate.

A. Extension to Control-Affine Systems

The CEGIS framework as mentioned can be used to find non-zero CLF for switched systems. However, it is not restricted to this class of CLFs. In this section, we discuss how such framework can be used to discover CLF for control-affine systems as well. Assume we have a nonlinear control-affine dynamical system as below

$$\dot{\mathbf{x}} = f(\mathbf{x}) + g(\mathbf{x})\mathbf{u},$$

where $f(\mathbf{x}) : \mathbb{R}[\mathbf{x}]^n$ is a homogeneous polynomial vector ($f(\mathbf{0}) = \mathbf{0}$), $\mathbf{u} : \mathbb{R}^p$ is the input vector. Also $\mathbf{u} \in U$ and

$U (\ni \mathbf{0})$ is a closed bounded polyhedra. $g(\mathbf{x}) : \mathbb{R}[\mathbf{x}]^{n \times p}$ is a polynomial matrix. From definition of CLF [2], we know that $V (V(\mathbf{0}) = 0)$ is a CLF if

$$(\forall \mathbf{x} \in P \setminus \{\mathbf{0}\})$$

$$V(\mathbf{x}) > \alpha(\mathbf{x})$$

$$\min_{\mathbf{u} \in U} \dot{V}(\mathbf{x}, \mathbf{u}) < -\alpha_Q(\mathbf{x})$$

Let U^v be set of vertices of polyhedron U . Each $\mathbf{u}^* \in U$ can be written as a convex combination of elements of U^v . Also, because of linearity of \mathbf{u} in $\dot{V}_q(\mathbf{x}, \mathbf{u})$, if $\dot{V}(\mathbf{x}, \mathbf{u}^*) < -\alpha_Q(\mathbf{x})$ for some $\mathbf{u}^* \in U$:

$$(\exists \lambda) \lambda_{\mathbf{u}} \geq 0 \wedge \sum_{\mathbf{u} \in U^v} \lambda_{\mathbf{u}} = 1 \wedge \mathbf{u}^* = \sum_{\mathbf{u} \in U^v} \lambda_{\mathbf{u}} \mathbf{u}$$

$$\Rightarrow \dot{V}(\mathbf{x}, \left(\sum_{\mathbf{u} \in U^v} \lambda_{\mathbf{u}} \mathbf{u} \right)) < -\alpha_Q(\mathbf{x})$$

$$\Rightarrow \sum_{\mathbf{u} \in U^v} \lambda_{\mathbf{u}} \dot{V}(\mathbf{x}, \mathbf{u}) < -\alpha_Q(\mathbf{x})$$

$$\Rightarrow (\exists \mathbf{u} \in U^v) \dot{V}(\mathbf{x}, \mathbf{u}) < -\alpha_Q(\mathbf{x})$$

One can define a switched system with modes $Q = \{q_{\mathbf{u}} | \mathbf{u} \in U^v\}$ and dynamics for each mode $q_{\mathbf{u}}$ as $f_{q_{\mathbf{u}}} = f(\mathbf{x}) + g(\mathbf{x})\mathbf{u}$ and claim that $V (V(\mathbf{0}) = 0)$ is a CLF iff

$$(\forall \mathbf{x} \in P \setminus \{\mathbf{0}\})$$

$$V(\mathbf{x}) > \alpha(\mathbf{x})$$

$$(\exists q \in Q) \dot{V}_q(\mathbf{x}) = \dot{V}(\mathbf{x}, \mathbf{u}_q) < -\alpha_Q(\mathbf{x})$$

Then, CEGIS framework can be employed to find a CLF to solve this problem, given α and α_Q . Once a CLF is found, we can also synthesize the appropriate controller as discussed in Section III-B or known methods from control theory [20] can be applied to find a feedback law.

V. EVALUATION

In this section, we demonstrate the effectiveness of the LMI-based CEGIS framework on some benchmark nonlinear problems. Our implementation consists of a python script which interacts with two other parts: (a) The Z3 SMT solver used for finding CLF candidates by solving linear arithmetic formulae over the reals [6], and (b) The CVXOPT [1] solver which is used to solve mixed cone constraints. While Z3 is an ‘‘exact arithmetic’’ solver, CVXOPT relies on numerical calculations that are susceptible to error.

The inputs for our implementation are: (i) continuous variables, (ii) ODEs for each control mode, (iii) region P (assumed to be a box), (iv) a template for the CLF and (v) ϵ_q for each mode. The vertices of region P are used as initial witness points X_0 , and we also fixed $\alpha(\mathbf{x}) = \sum_{i=1}^n x_i^2$ and chose $\phi_q(\mathbf{x}) = \sum_{i=1}^n x_i^2$ for all modes in all problems. We use a generic quadratic form for the CLF (i.e. all the monomials with degree 2), unless otherwise mentioned.

We collected a set of 21 benchmarks to evaluate the proposed approach. The instances of these benchmarks are taken from the literature including control-affine feedback systems and switched systems. A full description of the

TABLE I
RESULTS OF RUNNING OUR IMPLEMENTATION ON THE SWITCHED SYSTEMS BENCHMARK SUITE

Problem				Previous Results					New Results				
ID	n	α	Spec	itr	z3 T	SMT T	Tot. T	Stat	itr	z3 T	SDP T	Tot. T	Stat
1	2	0.01 $\ \mathbf{x}\ ^2$	AS	1	0.0	0.8	0.8	✓	18	3.9	1.7	5.9	✓
2	2	0.01	RS	3	0.0	3.4	3.6	✓	30	0.5	2.0	2.8	✗
3	2	0.0001	RS	6	0.1	1.6	2.0	✓	10	0.1	0.8	1.0	✓
4	2	0.1	RS	6	0.1	3.6	4.0	✓	12	0.2	1.5	2.1	✓
5	3	0.1 $\ \mathbf{x}\ ^2$	AS	13	2.2	352	355.2	✓	4	0.1	0.7	1.3	✓
6	3	0.1 $\ \mathbf{x}\ ^2$	AS	TO				✗	1	0.0	0.2	0.7	✓
7	3	0.05	RS	8	4.4	80.8	86.2	✓	1	0.0	0.3	0.7	✓
8	3	1.0	RS	36	48.1	57.3	108.4	✓	13	0.4	1.7	2.5	✓
9	3	0.001	RS	1	0.0	2.1	2.2	✓	1	0.0	0.1	0.5	✓
10	4	0.001	RS	TO				✗	1	0.0	0.4	2.0	✓
11	4	0.001	RS	1	0.0	14.9	14.9	✓	1	0.0	0.3	1.5	✓
12	5	0.001	RS	1	0.0	596.5	596.5	✓	1	0.0	0.3	3.7	✓
13	6	0.001	RS	2	0.5	2994.0	2995.6	✓	1	0.4	0.5	9.2	✓
14	9	0.001	RS	TO				✗	2	0.0	0.3	202.3	✓

Legend: n : # state variables, AS: Asymptotic Stability, RS: Region Stability, itr : # iterations, Tot. T: total computation time, Z3 T: time taken by Z3, SMT T: time taken by the SMT solver for finding counter-examples, SDP T: time taken by CVXOPT, TO: timed out, NA: not applicable, ✓: Success, ✗: Failed. All timings are in seconds.

TABLE II
RESULTS OF RUNNING OUR IMPLEMENTATION ON THE CONTROL-AFFINE SYSTEMS BENCHMARK SUITE

Problem			Results				
ID	n	ϵ_q	itr	z3 T	SDP T	Tot. T	Stat
15	2	0.1	34	2.7	2.3	5.2	✓
16	2	0.0	1	0.0	0.1	0.3	✓
17	2	0.05	38	1.7	4.1	6.2	✓
18	2	0.0	20	0.4	1.9	2.6	✓
19	3	1.0	1	0.0	1.0	3.0	✓
20*	4	0.0	46	164.4	30.3	202.2	✓
21	6	0.0	TO				✗

Legend: See Legend of Table I.

* After failure with a quadratic template, a template with 9 monomial selected carefully according to the dynamics $(V(x, y, z, w) = c_1x^2 + c_2y^2 + c_3z^2 + c_4w^2 + c_5yz + c_6xz + c_7xz^3 + c_8z^4 + c_9z^6)$

benchmarks is available in the extension version of this paper [17].

In first phase of the evaluation, we considered a set of switched system problems with multiple control modes. For some of these problems, the origin is not an equilibrium for any of the modes: therefore, stabilization is not possible with finite dwell time. Therefore, we considered the problem of stabilizing to a small neighborhood of the origin. To do so, the CLF conditions are relaxed to eliminate the small region around the equilibrium [18]. The rest of our framework applies directly. The results are shown in Table I.

Next, we considered a set of problems with control-affine feedback systems. For these systems we solve the problem by finding a CLF (not necessarily non-zero) and for all the problems we chose $\alpha_Q(\mathbf{x}) = \sum_{i=1}^n x_i^2$. If such CLF does not exist, then we try to find a CLF with $\epsilon_q = 0$. The results are shown in Table II.

In summary, from the given 14 problem instances in Table I, we find that the LMI relaxation introduced here, fails to solve one problem instance due to the LMI relaxation.

One solution to address the loss in precision is to decompose the state space for getting more precise abstraction of the state space. On the other hand, the proposed technique can solve three previously unsolved instances that are among the larger ones in our benchmarks. The timings for our LMI-based approach are nearly an order of magnitude faster than our earlier approach, especially for larger examples.

As results suggest, the problem of finding a CLF can be solved in few iterations. Finding witnesses using LMI-relaxation are significantly faster compared to the previous approach using non-linear solvers (Z3 or dReal [7]). As currently, problems with as many as 9 variables are solvable. However, the framework fails to terminate for the problem with 6 variables, due to high complexity of finding a CLF candidate. The problem of finding a CLF candidate using linear real arithmetic is the bottleneck of the computations in our new framework, whereas the nonlinear solver is the bottleneck for the older framework. The size of the related problem depends on the size of the template and the number of witness points. Therefore, one challenging problem is to carefully choose a small template (as in System 21) in order to manage the complexity of these problems.

VI. CONCLUSION

In this work we introduced a class of CLFs, namely non-zero CLFs which guarantee the existence of a switching strategy for asymptotic stability of switched system. We also proposed a LMI-based CEGIS framework for finding CLFs for switched systems as well as control-affine systems and we evaluated the proposed approach on a set of benchmark from the literature. The main shortcoming of this framework comes from hardness of solving formulae in linear arithmetics and as SMT solvers improve, we hope this approach can solve bigger problems. Going forward, we are investigating extension of this framework for finding control barrier certificates to solve safety problems.

Acknowledgments: This work was supported by the US National Science Foundation (NSF) under CAREER award # 0953941 and CCF-1527075. All opinions expressed are those of the authors and not necessarily of the NSF.

REFERENCES

- [1] M. S. Andersen, J. Dahl, and L. Vandenberghe. Cvxopt: A python package for convex optimization, version 1.1. 6. Available at cvxopt.org, 2013.
- [2] Z. Artstein. Stabilization with relaxed controls. *Nonlinear Analysis: Theory, Methods & Applications*, 7(11):1163–1173, 1983.
- [3] E. Asarin, O. Bournez, T. Dang, O. Maler, and A. Pnueli. Effective synthesis of switching controllers for linear systems. *Proceedings of the IEEE*, 88(7):1011–1025, 2000.
- [4] R. Bartle and D. Sherbert. *Introduction to Real Analysis, 4th Edition*. John Wiley & Sons, 2011.
- [5] P. Colaneri, J. C. Geromel, and A. Astolfi. Stabilization of continuous-time switched nonlinear systems. *Systems & Control Letters*, 57(1):95–103, 2008.
- [6] L. De Moura and N. Björner. Z3: An efficient smt solver. In *Tools and Algorithms for the Construction and Analysis of Systems*, pages 337–340. Springer, 2008.
- [7] S. Gao, S. Kong, and E. M. Clarke. dreal: An SMT solver for nonlinear theories over the reals. In *Intl. Conference on Automated Deduction (CADE)*, pages 208–214, 2013.
- [8] J. B. Lasserre. Semidefinite programming vs. lp relaxations for polynomial programming. *Mathematics of operations research*, 27(2):347–360, 2002.
- [9] H. Lin and P. J. Antsaklis. Switching stabilizability for continuous-time uncertain switched linear systems. *Automatic Control, IEEE Transactions on*, 52(4):633–646, 2007.
- [10] H. Lin and P. J. Antsaklis. Hybrid dynamical systems: An introduction to control and verification. *Found. Trends Syst. Control*, 1(1):1–172, 2014.
- [11] J. Mazo, Manuel, A. Davitian, and P. Tabuada. PESSOA: a tool for embedded controller synthesis. In *CAV*, volume 6174 of *LNCS*, pages 566–569. Springer, 2010.
- [12] J. D. Meiss. *Differential Dynamical Systems*. SIAM, 2007.
- [13] P. Nilsson and N. Ozay. Incremental synthesis of switching protocols via abstraction refinement. In *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*. IEEE, 2014.
- [14] A. Papachristodoulou and S. Prajna. On the construction of lyapunov functions using the sum of squares decomposition. In *Decision and Control, 2002. Proceedings of the 41st IEEE Conference on*, volume 3, pages 3482–3487. IEEE, 2002.
- [15] P. A. Parillo. Semidefinite programming relaxation for semialgebraic problems. *Mathematical Programming Ser. B*, 96(2):293–320, 2003.
- [16] A. Podolski and S. Wagner. A sound and complete proof rule for region stability of hybrid systems. In *HSCC*, volume 4416 of *LNCS*, pages 750–753. Springer, 2007.
- [17] H. Ravanbakhsh and S. Sankaranarayanan. Counter-example guided synthesis of control lyapunov functions for switched systems. *arXiv preprint arXiv:1509.03688*, 2015. (Extended Version).
- [18] H. Ravanbakhsh and S. Sankaranarayanan. Counterexample guided synthesis of switched controllers for reach-while-stay properties. *arXiv preprint arXiv:1505.01180*, 2015.
- [19] A. Solar-Lezama, L. Tancau, R. Bodik, S. Seshia, and V. Saraswat. Combinatorial sketching for finite programs. In *ACM Sigplan Notices*, volume 41, pages 404–415. ACM, 2006.
- [20] J. Solis-Daun and H. Leyva. On the global clf stabilization of systems with polytopic control value sets. In *Proc. 18th IFAC World Congress*, pages 11042–11047, 2011.
- [21] W. Tan and A. Packard. Searching for control lyapunov functions using sums of squares programming. In *Allerton Conference*, pages 210–219, 2004.

APPENDIX

Proof of Theorem 1 Given regions P , a plant Ψ and a non-zero CLF $V(\mathbf{x})$, let P^* be the associated region for V w.r.t P . Given $\mathbf{x}(0) \in P^*$, a switching function that admit

the description of Equation (5) results in a system which satisfies the following properties.

- 1) P^* is a positive invariant.
- 2) all the traces of the system are time-divergent
- 3) system is asymptotically stable w.r.t P^*

Proof: We first prove that P^* is a positive invariant. Recall that P is a compact set containing $\mathbf{0}$ and let ∂P denote it's boundary. Also, recall that $\beta(P, V) := \min_{\mathbf{x} \in \partial P} V(\mathbf{x})$.

Consider a class of switch functions defined below.

$$\text{switch}(q, \mathbf{x}) := \begin{cases} \hat{q} & \left(\begin{array}{l} \dot{V}_q(\mathbf{x}) \geq -\frac{\epsilon_q \phi_q(\mathbf{x})}{\lambda} \wedge \\ \dot{V}_{\hat{q}}(\mathbf{x}) \leq -\epsilon_{\hat{q}} \phi_{\hat{q}}(\mathbf{x}) \wedge \mathbf{x} \in P \end{array} \right) \\ q & \text{otherwise} \end{cases}$$

We note that $\text{switch}(q, \mathbf{x})$ is defined over all $\mathbf{x} \in P$ and $q \in Q$ by construction of the CLF V . Assume $\mathbf{x}(0) \in P^*$ and $q \in Q$ such that $\dot{V}_q(\mathbf{x}(0)) \leq -\epsilon_q \phi_q(\mathbf{x})$. We obtain $V(\mathbf{x}(0)) < \beta(P, V)$. Also, the switch function ensures that as long as $\mathbf{x}(t) \in P$, $\dot{V}_{q(t)}(\mathbf{x}(t)) \leq -\frac{\epsilon_{q(t)} \phi_{q(t)}(\mathbf{x}(t))}{\lambda} < 0$. Therefore

$$V(\mathbf{x}(t_b)) = V(\mathbf{x}(0)) + \int_0^{t_b} \dot{V}_{q(t)}(\mathbf{x}(t)) dt \leq V(\mathbf{x}(0))$$

Since $V(\mathbf{x}(0)) < \beta(P, V)$, we have $V(\mathbf{x}(t)) < \beta(P, V)$. Therefore, by definition $\mathbf{x}(t) \in P^*$.

Next, we show there exists a min dwell time between two switching times. Assume there is a switch time t_1 s.t. $\mathbf{x}(t_1) \in P^*$ and mode switches to q . Thus,

$$\dot{V}_q(\mathbf{x}(t_1^+)) \leq -\epsilon_q \phi_q(\mathbf{x}(t_1^+)) \quad (11)$$

Let t_2 be the next time instance when the controller switches to mode \hat{q} . By definition of the controller we can conclude

$$\dot{V}_{\hat{q}}(\mathbf{x}(t_2^-)) = -\frac{\epsilon_{\hat{q}} \phi_{\hat{q}}(\mathbf{x}(t_2^-))}{\lambda} \quad (12)$$

It is sufficient to show $\delta = t_2 - t_1$ has a lower bound and it can not be arbitrarily small.

From Equation (2) and (3) and boundedness of P there are constants Λ_1 and Λ_2 s.t. for all $\mathbf{x} \in P$

$$\ddot{V}_q(\mathbf{x}) \leq \Lambda_1 \phi_q(\mathbf{x}) \quad (13)$$

$$\dot{\phi}_q(\mathbf{x}) \leq \Lambda_2 \phi_q(\mathbf{x}) \quad (14)$$

From Equation (14), we get

$$(\forall t \in [t_1, t_2])$$

$$\begin{aligned} \phi_q(\mathbf{x}(t)) &= \phi_q(\mathbf{x}(t_1)) + \int_{t_1}^t \dot{\phi}_q(\mathbf{x}(\tau)) d\tau \\ &\leq \phi_q(\mathbf{x}(t_1)) + \int_{t_1}^t \Lambda_2 \phi_q(\mathbf{x}(\tau)) d\tau \end{aligned}$$

and therefore

$$\phi_q(\mathbf{x}(t)) \leq e^{\Lambda_2 \delta} \phi_q(\mathbf{x}(t_1)) \quad (15)$$

A lower bound on $\dot{V}_q(\mathbf{x}(t_2^-))$ by Equation (12)

$$\begin{aligned} \dot{V}_q(\mathbf{x}(t_2^-)) &= -\frac{\epsilon_q \phi_q(\mathbf{x}(t_2^-))}{\lambda} \\ \stackrel{\text{Equation (15)}}{\implies} &\geq -\frac{e^{\Lambda_2 \delta} \epsilon_q \phi_q(\mathbf{x}(t_1))}{\lambda} \end{aligned} \quad (16)$$

Also

$$\begin{aligned} (\forall t \in (t_1, t_2)) \\ \dot{V}_q(\mathbf{x}(t)) &= \dot{V}_q(\mathbf{x}(t_1^+)) + \int_{t_1}^t \ddot{V}_q(\mathbf{x}(\tau)) d\tau \\ \stackrel{\text{Equation (13)}}{\implies} &\leq \dot{V}_q(\mathbf{x}(t_1)) + \Lambda_1 \int_{t_1}^t \phi_q(\mathbf{x}(\tau)) d\tau \\ \stackrel{\text{Equation (15)}}{\implies} &\leq \dot{V}_q(\mathbf{x}(t_1)) + \Lambda_1 \int_{t_1}^t e^{\Lambda_2 \delta} \phi_q(\mathbf{x}(t_1)) d\tau \end{aligned}$$

and therefore an upper bound on $\dot{V}_q(\mathbf{x}(t_2))$ is

$$\begin{aligned} \dot{V}_q(\mathbf{x}(t_2)) &\leq \dot{V}_q(\mathbf{x}(t_1)) + \Lambda_1 e^{\Lambda_2 \delta} \phi_q(\mathbf{x}(t_1)) \delta \\ \stackrel{\text{Equation (11)}}{\implies} &\leq -\epsilon_q \phi_q(\mathbf{x}(t_1)) + \Lambda_1 e^{\Lambda_2 \delta} \phi_q(\mathbf{x}(t_1)) \delta \end{aligned} \quad (17)$$

From Equations (16), (17)

$$\begin{aligned} -\frac{e^{\Lambda_2 \delta} \epsilon_q \phi_q(\mathbf{x}(t_1))}{\lambda} &\leq \dot{V}_q(\mathbf{x}(t_2)) \\ &\leq -\epsilon_q \phi_q(\mathbf{x}(t_1)) + \Lambda_1 e^{\Lambda_2 \delta} \phi_q(\mathbf{x}(t_1)) \delta \end{aligned}$$

and finally assuming $\mathbf{x}(t_1) \neq 0$, we have $\phi(\mathbf{x}(t_1)) > 0$:

$$\begin{aligned} -\frac{e^{\Lambda_2 \delta} \epsilon_q}{\lambda} &\leq -\epsilon_q + \Lambda_1 e^{\Lambda_2 \delta} \delta \\ \implies \epsilon_q &\leq \frac{\lambda \Lambda_1 e^{\Lambda_2 \delta} \delta}{(\lambda - e^{\Lambda_2 \delta})} = h(\delta) \end{aligned} \quad (18)$$

Notice that

- 1) $0 \leq e^{\Lambda_2 \delta} < \lambda \iff h(\delta) > 0$. Since λ is a chosen parameter, it can always be chosen sufficiently large to ensure this inequality.
- 2) h is a monotone function of δ in domain $0 \leq e^{\Lambda_2 \delta} < \lambda$ by showing that $\frac{dh}{d\delta}$ is positive.
- 3) $h(0) = 0$ and $\lim_{\delta \rightarrow \frac{\log(\lambda)}{\Lambda_2}} h(\delta) = +\infty$.

$h^{-1} : \mathbb{R}^+ \rightarrow \mathbb{R}^+$ is defined and $h^{-1}(\epsilon_q) \leq \delta$. Therefore, $h^{-1}(\epsilon_q)$ is a lower bound on δ , and all traces of the system are time-divergent.

In the next step of the proof, we want to show the system is asymptotically stable. Since P^* is a compact set, $(\forall t > 0) \mathbf{x}(t) \in P^*$ and time diverges, by Bolzano-Weierstrass Theorem [4], $\mathbf{x}(t)$ converges to some $\mathbf{x}^* \in P^*$. Assume $\mathbf{x}^* \neq \mathbf{0}$ and therefore $\min_q(\epsilon_q \phi_q(\mathbf{x}^*)) = R > 0$. By continuity of ϕ_q and divergence of time, one can find $\epsilon > 0$ s.t.

$$\begin{aligned} (\exists T > 0) (\forall t \geq T) \mathbf{x}(t) &\in \mathcal{B}_\epsilon(\mathbf{x}^*) \subseteq P^* \\ (\forall q \in Q) (\forall \mathbf{x} \in \mathcal{B}_\epsilon(\mathbf{x}^*)) &\epsilon_q \phi_q(\mathbf{x}) \geq \frac{R}{2} \end{aligned}$$

Also V is bounded in $\mathcal{B}_\epsilon(\mathbf{x}^*)$ and decreases through time. Formally,

$$(\forall t \geq T) \dot{V}_{q(t)}(\mathbf{x}(t)) \leq -\frac{\epsilon_{q(t)} \phi_{q(t)}(\mathbf{x}(t))}{\lambda} \leq -\frac{R}{2\lambda}$$

As a result

$$\begin{aligned} V(\mathbf{x}(T+t)) &= V(\mathbf{x}(T)) + \int_T^{T+t} \dot{V}_{q(\tau)}(\mathbf{x}(\tau)) d\tau \\ &\leq V(\mathbf{x}(T)) - \frac{R}{2\lambda} t \end{aligned}$$

which means eventually V becomes negative as time goes to infinity and that is a contradiction. Therefore $\mathbf{x}^* = \mathbf{0}$ and the system is asymptotically stable. ■

Proof of Theorem 2 Given a function $\phi(\mathbf{x}) = \sum_{i=1}^n \mathbf{x}_i^{2d_i}$ and a function $p : X \rightarrow R$ p is ϕ -bounded if

$$(\forall m \in \text{Monos}(p)) (\forall i) 2d_i \leq \text{Deg}(m).$$

Proof: Assume Equation (6) holds and S is a bounded region. We want to show there exists a Λ s.t. $(\forall \mathbf{x} \in S) p(\mathbf{x}) \leq \Lambda \phi(\mathbf{x})$.

For a monomial $m \in \text{Monos}(p)$ and i s.t. $\mathbf{x}_i \in \text{Vars}(m)$, let $R(i, m)$ be the following region

$$R(i, m) = \{\mathbf{x} \in [-1, 1]^n \mid (\forall j \in \text{Vars}(m)) |\mathbf{x}_j| \geq |\mathbf{x}_i|\}$$

Notice that $[-1, 1]^n = \bigcup_i R(i, m)$. Also

$$(\forall \mathbf{x} \in R(i, m)) m(\mathbf{x}) \leq |\mathbf{x}_i|^{\text{Deg}(m)} \leq \mathbf{x}_i^{2d_i} \leq \phi(\mathbf{x})$$

and therefore

$$(\forall m \in \text{Monos}(p)) (\forall \mathbf{x} \in [-1, 1]^n) m(\mathbf{x}) \leq \phi(\mathbf{x})$$

Since S is bounded, there is a constant Λ_0 s.t. $S \subseteq [-\Lambda_0, \Lambda_0]^n$. Then

$$(\forall m \in \text{Monos}(p)) (\forall \mathbf{x} \in [-\Lambda_0, \Lambda_0]^n) m(\mathbf{x}) \leq \Lambda_0^{\text{Deg}(p)} \phi(\mathbf{x})$$

Now let $p(\mathbf{x}) = \sum_i c_i m_i(\mathbf{x})$ where $m_i(\mathbf{x}) \in \text{Monos}(p)$ and c_i is its coefficient in p . Therefore $\forall \mathbf{x} \in S$

$$\begin{aligned} p(\mathbf{x}) &= \sum_i c_i m_i(\mathbf{x}) \\ &\leq \sum_i |c_i| \Lambda_0^{\text{Deg}(m_i)} \phi(\mathbf{x}) \\ &= \Lambda_0^{\text{Deg}(p)} \left(\sum_i |c_i| \right) \phi(\mathbf{x}) \end{aligned}$$

Thus there exists a $\Lambda = \Lambda_0^{\text{Deg}(p)} \left(\sum_i |c_i| \right)$ s.t.

$$(\forall \mathbf{x} \in S) p(\mathbf{x}) \leq \Lambda \phi(\mathbf{x})$$

■