Software Engineering Group
Department of Computer Science
Nanjing University
http://seg.nju.edu.cn

Technical Report No. NJU-SEG-2016-IC-009

2016-IC-009

A Linear Programming Relaxation Based Approach for Generating Barrier Certificates of Hybrid Systems

Zhengfeng Yang, Chao Huang, Xin Chen, Wang Lin, Zhiming Liu

Formal Methods - 21st International Symposium 2016

A Linear Programming Relaxation Based Approach for Generating Barrier Certificates of Hybrid Systems

Zhengfeng Yang[1], Chao Huang[2], Xin Chen[2], Wang Lin[3](✉), and Zhiming Liu[4]

[1] Shanghai Key Lab of Trustworthy Computing, East China Normal University, Shanghai, China
zfyang@sei.ecnu.edu.cn

[2] State Key Lab for Novel Software Technology, Nanjing University, Nanjing, China
{huangchao,chenxin}@nju.edu.cn

[3] Key Lab of Mathematics Mechanization, AMSS, CAS, Beijing, China
linwang@wzu.edu.cn

[4] Center for Research and Innovation in Software Engineering, Southwest University, Chongqing, China
zhimingliu88@swu.edu.cn

Abstract. This paper presents a linear programming (LP) relaxation based approach for generating polynomial barrier certificates for safety verification of semi-algebraic hybrid systems. The key idea is to introduce an LP relaxation to encode the set of nonnegativity constraints derived from the conditions of the associated barrier certificates and then resort to LP solvers to find the solutions. The most important benefit of the LP relaxation based approach is that it possesses a much lower computational complexity and hence can be solved very efficiently, which is demonstrated by the theoretical analysis on complexity as well as the experiment on a set of examples gathered from the literature. As far as we know, it is the first method that enables LP relaxation based polynomial barrier certificate generation.

Keywords: Formal verification · Hybrid systems · Barrier certificates · Linear programming relaxation

1 Introduction

Safety verification of hybrid systems has attracted much research attention in recent years [2]. This is mainly due to the requirement of ensuring the safety of embedded systems whose complex behaviors can be exhibited by hybrid systems via interacting discrete and continuous dynamics [3,12]. In principle, safety verification aims to decide that starting from an initial set, whether a system can evolve to some unsafe region in the state space. A successful verification can give more confidence in the verified systems.

Barrier certificate based methods are developed to handle the safety verification problem [13,16,17,27]. A barrier certificate is a function of state that

J. Fitzgerald et al. (Eds.): FM 2016, LNCS 9995, pp. 721–738, 2016.
DOI: 10.1007/978-3-319-48989-6_44

divides the state space into two parts. All system trajectories starting from a given set of initial conditions fall into one side while the unsafe region locates on the other. Thus, the problem of safety verification is converted to the problem of barrier certificate generation. Compared with reachable set computation, when encountering nonlinear systems, a barrier function is much easier to compute. It also gives more exact result when the safety property refers to infinite horizon [17].

Barrier certificate generation is a computationally intensive task. Usually, a function of a specific form with unknown coefficients is given as the template, and then computational methods based on different principles are used to determine the value of those unknown coefficients so that the conditions of the desired barrier certificate are satisfied. For barrier certificate based verification, its effectiveness and practicality are decided to a large extent by the efficiency of the computational methods, therefore the method for effective computation becomes a key point.

There have been many barrier certificates of different types proposed for hybrid systems with different features [13,15,21,27,31]. Among them, polynomial barrier certificates for semi-algebraic hybrid systems (i.e. those systems whose vector fields are polynomials and whose set descriptions are polynomial equalities or inequalities) receive most attention, as they are more universal. For barrier certificates generation, methods based on sums of squares (SOS) relaxation are quite popular, as the associated semidefinite programming (SDP) has a much lower computational complexity and there are many efficient solvers available.

The paper focuses on introducing linear programming (LP) relaxation to generating polynomial barrier certificates with convex condition for semi-algebraic hybrid systems. Compared with SOS relaxation based approaches, our LP relaxation based method offers three main advantages: First, LP has a much lower computational complexity than SDP does, thus it can be solved more quickly. Second, LP provides a much higher numerical stability and hence can treat many cases where SDP generates invalid polynomials due to numerical errors [15,21]. At last, LP gives a new encoding of polynomial positivity quite different from SDP, and thus has the potential to generate polynomials that SDP is unable to produce. It is a necessary complement to relaxation based methods as it can generate barrier certificates uncovered by existing methods.

The proposed method considers a polynomial barrier certificate whose coefficients must satisfy a set of nonnegativity constraints of multivariate polynomials over semi-algebraic sets. It employs the theory of Krivine-Vasilescu-Handelman's (KVH) Positivstellensatz [14] to construct an LP relaxation of the constraint set and then relies on LP solvers to find the solution for the coefficients of the barrier certificate. The theoretical analysis demonstrates that for a hybrid system, the complexity of finding the solution based on the LP solver is approximately $O(n^{2d+D})$ while that based on the SDP solver is approximately $O(n^{4D})$, where n is the number of system variables, d and D are the degree bounds of the barrier certificate and its nonnegative representation derived from LP relaxation

and SOS relaxation, respectively. Our LP relaxation based method is compared with the SOS relaxation based approach over a set of benchmarks gathered from the literature, which shows that our method provides much better efficiency. To the best of our knowledge, it is the first study that enables LP relaxation based polynomial barrier certificate generation.

We start by defining continuous systems and hybrid systems in Sect. 2. We then present our approach and give a complexity analysis on both our LP relaxation based method and the SOS relaxation based method in Sect. 3. We present how to use our approach to generate barrier certificate for several nontrivial examples and compare the efficiency of our method with SOS relaxation based method over a set of benchmarks in Sect. 4. We compare with related works in Sect. 5 before concluding.

2 Continuous and Hybrid Systems

Notations. Let $\mathbb{R}$ and $\mathbb{N}$ be the field of real number and natural number, respectively; $\mathbb{R}[\mathbf{x}]$ denotes the polynomial ring with coefficients in $\mathbb{R}$ over $\mathbf{x} = [x_1, x_2, \cdots, x_n]^T$, and $\mathbb{R}[\mathbf{x}]^n$ denotes the n-dimensional polynomial ring vector.

A continuous dynamical system is modeled by a finite number of first-order ordinary differential equations

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}), \tag{1}$$

where $\dot{\mathbf{x}}$ denotes the derivative of $\mathbf{x}$ with respect to the time variable t, and $\mathbf{f}(\mathbf{x})$ is called vector field $\mathbf{f}(\mathbf{x}) = [f_1(\mathbf{x}), \cdots, f_n(\mathbf{x})]^T$ defined on an open set $\Psi \subseteq \mathbb{R}^n$. We assume that $\mathbf{f}$ satisfies the local Lipschitz condition, which ensures that given $\mathbf{x}(0) = \mathbf{x}_0$, there exists a time $T > 0$ and a unique function $\tau : [0, T) \mapsto \mathbb{R}^n$ such that $\tau(t) = \mathbf{x}(t)$. And $\mathbf{x}(t)$ is called a solution of (1) that starts at a certain initial state $\mathbf{x}_0$. Namely, $\mathbf{x}(t)$ is also called a trajectory of (1) from $\mathbf{x}_0$.

Definition 1 (Continuous System). *A continuous system over* $\mathbf{x}$ *consists of a tuple* $\mathbf{S} : \langle \Theta, \mathbf{f}, \Psi \rangle$*, wherein* $\Theta \subseteq \mathbb{R}^n$ *is a set of initial states,* $\mathbf{f}$ *is a vector field over the domain* $\Psi \subseteq \mathbb{R}^n$*.*

Hybrid systems involve both continuous dynamics as well as discrete transitions. To model hybrid systems, we use the notion of hybrid automata [3].

Definition 2 (Hybrid Automata). *A* hybrid automaton *is a system* $\mathbf{H} : \langle L, X, F, \Psi, E, G, R, \Theta, \ell_0 \rangle$*, where*

- L*, a finite set of locations (or models);*
- $X \subseteq \mathbb{R}^n$ *is the continuous state space. The hybrid state space of the system is defined by* $\mathcal{X} = L \times X$ *and a state is defined by* $(\ell, \mathbf{x}) \in \mathcal{X}$*;*
- $F : L \to (\mathbb{R}^n \to \mathbb{R}^n)$*, assigns to each location* $\ell \in L$ *a locally Lipschitz continuous vector field* $\mathbf{f}_\ell$*;*
- Ψ *assigns to each location* $\ell \in L$ *a* location condition (location invariant) $\Psi(\ell) \subseteq \mathbb{R}^n$*;*

- *$E \subseteq L \times L$ is a finite set of discrete transitions;*
- *G assigns to each transition $e \in E$ a switching guard $G_e \subseteq \mathbb{R}^n$;*
- *R assigns to each transition $e \in E$ a reset function $R_e : \mathbb{R}^n \to \mathbb{R}^n$;*
- *$\Theta \subseteq \mathbb{R}^n$, an initial continuous state set;*
- *$\ell_0 \in L$, the initial location. The initial state space of the system is defined by $\ell_0 \times \Theta$.*

A trajectory [31] of $\mathbf{H}$ is an infinite sequence of states

$$(l_0, \mathbf{x}_0),\ (l_1, \mathbf{x}_1), \cdots, (l_i, \mathbf{x}_i),\ (l_{i+1}, \mathbf{x}_{i+1}), \cdots$$

such that

- **[Initiation]** $(l_0, \mathbf{x}_0) \in \ell_0 \times \Theta$;
  Furthermore, for each consecutive pair $(l_i, \mathbf{x}_i), (l_{i+1}, \mathbf{x}_{i+1})$, one of the two *consecution* conditions holds:
- **[Discrete Consecution]** $e = (l_i, l_{i+1}) \in E$, $\mathbf{x}_i \in G_e$ and $x_{i+1} = R_e(\mathbf{x}_i)$; or
- **[Continuous Consecution]** $l_i = l_{i+1} = \ell$, and there exists a time interval $[0, \delta]$ such that the solution $\mathbf{x}(\mathbf{x}_i; t)$ to $\dot{\mathbf{x}} = \mathbf{f}_\ell$ evolves from $\mathbf{x}_i$ to $\mathbf{x}_{i+1}$, while satisfying the location invariant $\Psi(\ell)$. Formally,
  - $\mathbf{x}(\mathbf{x}_i, \delta) = \mathbf{x}_{i+1}$ and
  - $\forall t \in [0, \delta], \mathbf{x}(\mathbf{x}_i, t) \in \Psi(\ell)$.

A state $(\ell, \mathbf{x})$ is called a *reachable state* of a hybrid system $\mathbf{H}$ from the initial state set $\ell_0 \times \Theta$ if it appears in some trajectory of $\mathbf{H}$. During a continuous flow, the discrete location ℓ_i is maintained and the continuous state variables $\mathbf{x}$ evolve according to the differential equations $\dot{\mathbf{x}} = \mathbf{f}_{\ell_i}(\mathbf{x})$, with $\mathbf{x}$ satisfying the location invariant $\Psi(\ell_i)$. At the state $(\ell_i, \mathbf{x})$, if there is a discrete transition $e = (\ell_i, \ell_j) \in E$ such that $\mathbf{x} \in G_e$, the system may undergo a transition to location ℓ_j, and $\mathbf{x}$ will take the new value $\mathbf{x}'$, which is determined by the reset function R_e.

In this paper, we focus on continuous systems and hybrid systems whose elements are represented as polynomial relations (equalities and inequalities) over the system variables. In what follows, the definition of semi-algebraic hybrid system is provided. The definition of semi-algebraic continuous system is similar.

Definition 3 *(Semi-algebraic Hybrid System). A semi-algebraic hybrid system is a hybrid system: $\mathbf{H} : \langle L, X, F, \Psi, E, G, R, \Theta, \ell_0 \rangle$, where*

- *the continuous vector field $F(\ell)$ for each $\ell \in L$ is of the form $\dot{\mathbf{x}} = \mathbf{f}_\ell(\mathbf{x})$, where $\mathbf{f}_\ell(\mathbf{x}) \in \mathbb{R}[\mathbf{x}]^n$;*
- *the initial condition Θ, the location invariant $\Psi(\ell)$ for each $\ell \in L$, and the guard condition G_e for each $e \in E$ are semi-algebraic sets defined by polynomial inequalities with variables $\mathbf{x}$; $R_e \in \mathbb{R}[\mathbf{x}]^n$ is the reset function for each $e \in E$.*

For ease of presentation, the semi-algebraic sets Θ, $\Psi(\ell)$ and G_e in Definition 3 are represented as follows:

$$\begin{aligned}\Theta :&= \{\mathbf{x} \in \mathbb{R}^n \mid \theta_1(\mathbf{x}) \geq 0, \ldots, \theta_q(\mathbf{x}) \geq 0\},\\ \Psi(\ell) :&= \{\mathbf{x} \in \mathbb{R}^n \mid \psi_{\ell,1}(\mathbf{x}) \geq 0, \ldots, \psi_{\ell,r}(\mathbf{x}) \geq 0\},\\ G_e :&= \{\mathbf{x} \in \mathbb{R}^n \mid g_{e,1}(\mathbf{x}) \geq 0, \ldots, g_{e,s}(\mathbf{x}) \geq 0\},\end{aligned}$$

where $\ell \in L$, $e \in E$, and $\theta_i(\mathbf{x})$, $\psi_{\ell,j}(\mathbf{x})$, $g_{e,k}(\mathbf{x})$ are polynomials. In addition, hereafter we assume that the above semi-algebraic sets are compact.

Given a semi-algebraic hybrid system $\mathbf{H}$ with prespecified unsafe state set $\mathcal{X}_u = \ell \times X_u$, we say that the system $\mathbf{H}$ is *safe* if all trajectories of $\mathbf{H}$ starting from the initial state set $\ell_0 \times \Theta$, can not evolve to any state specified by $\mathcal{X}_u$. Given a semi-algebraic hybrid system $\mathbf{H}$, the problem of verifying the safety property is to decide that whether $\mathbf{H}$ is safe, or, any state specified by $\mathcal{X}_u$ is not reachable. Here we also assume that $\mathcal{X}_u$ is a compact semi-algebraic set, defined by

$$X_u(\ell) := \{\mathbf{x} \in \mathbb{R}^n \mid \zeta_{\ell,1}(\mathbf{x}) \geq 0, \ldots, \zeta_{\ell,p}(\mathbf{x}) \geq 0, \},$$

where $\zeta_{\ell,i} \in \mathbb{R}[\mathbf{x}], 1 \leq i \leq p$.

3 Computational Method for Barrier Certificates

For safety verification of (continuous or hybrid) dynamical systems, the notion of barrier certificates [16] plays an important role. A barrier certificate maps all the states in the reachable set to non-negative reals and all the states in the unsafe set to negative reals, thus can be employed to prove safety of dynamical systems. Utilizing barrier certificates has the benefit of avoiding explicit computation of the exact reachable set which is usually not tractable for nonlinear continuous and hybrid systems. In other words, a barrier certificate can be regarded as the over-approximation of the reachable set, and most importantly, is a boundary between the reachable set and the given unsafe state set. In the sequel, we propose a new computational method for generating the barrier certificates for safety verification of dynamical systems.

3.1 Barriers Certificates

As stated in [13], the key point in generating barrier certificates is how to establish verification conditions that are as less conservative as possible and how to efficiently compute the barrier certificates satisfying these verification conditions. Taking them into account, the idea that introduces auxiliary polynomials to offer relaxed verification conditions for barrier certificates of continuous and hybrid systems can be applied.

Theorem 1. *Let* $\mathbf{S} : \langle \Theta, \mathbf{f}, \Psi \rangle$ *be a semi-algebraic continuous system, and* X_u *be the given unsafe state set. Let* $\lambda(\mathbf{x})$ *be a given polynomial. If there exists a polynomial* $B(\mathbf{x}) \in \mathbb{R}[\mathbf{x}]$*, which satisfies the following conditions:*

(i) $B(\mathbf{x}) \geq 0 \, \forall \mathbf{x} \in \Theta$,
(ii) $\dot{B}(\mathbf{x}) - \lambda(\mathbf{x})B(\mathbf{x}) > 0 \, \forall \mathbf{x} \in \Psi$, *here* $\dot{B}(\mathbf{x})$ *denotes the Lie-derivative of* $B(\mathbf{x})$ *along the vector field* $\mathbf{f}$, *i.e.,* $\dot{B}(\mathbf{x}) = \sum_{i=1}^{n} \frac{\partial B}{\partial x_i} \cdot f_i(\mathbf{x})$,
(iii) $B(\mathbf{x}) < 0 \, \forall \mathbf{x} \in X_u$,

then $B(\mathbf{x})$ *is a barrier certificate of system* $\mathbf{S}$, *and the safety of* $\mathbf{S}$ *is guaranteed.*

Proof. Condition (ii) indicates that $\dot{B}(\mathbf{x}) > 0$ if $B(\mathbf{x}) = 0$. Therefore, by condition (i) and (ii), $B(\mathbf{x})$ cannot become negative during the continuous evolution of $\mathbf{S}$. Condition (iii) implies that all trajectories starting from Θ can not enter X_u. We can conclude $B(\mathbf{x})$ is a barrier certificate of $\mathbf{S}$, which can guarantee the safety of the system. □

Clearly, the existence of such a barrier certificate in Theorem 1 suffices to guarantee the safety property of the given semi-algebraic continuous system. Likewise, Theorem 1 can be generalized to attack safety verification of semi-algebraic hybrid systems.

Theorem 2. *Let* $\mathbf{H} : \langle L, X, F, \Psi, E, G, R, \Theta, \ell_0 \rangle$ *be a semi-algebraic hybrid system,* $\mathcal{X}_u$ *be the unsafe assertion. Let* $\lambda_\ell(\mathbf{x})$ *be given polynomials for all* $\ell \in L$, *and* $\gamma_e(\mathbf{x})$ *be given nonnegative polynomials for all* $e \in E$. *If there exists a polynomial* $B_\ell(\mathbf{x}) \in \mathbb{R}[\mathbf{x}]$ *for each location* $\ell \in L$, *which satisfies the following conditions:*

(i) $B_{\ell_0}(\mathbf{x}) \geq 0 \, \forall \mathbf{x} \in \Theta$,
(ii) $\dot{B}_\ell(\mathbf{x}) - \lambda_\ell(\mathbf{x})B_\ell(\mathbf{x}) > 0 \, \forall \mathbf{x} \in \Psi(\ell)$, *here* $\dot{B}_\ell(\mathbf{x})$ *denotes the Lie-derivative of* $B_\ell(\mathbf{x})$ *along the vector field* $\mathbf{f}_\ell$, *i.e.,* $\dot{B}_\ell(\mathbf{x}) = \sum_{i=1}^{n} \frac{\partial B_\ell}{\partial x_i} \cdot f_{\ell,i}(\mathbf{x})$,
(iii) $B_{\ell'}(\mathbf{x}') - \gamma_e(\mathbf{x})B_\ell(\mathbf{x}) \geq 0 \, \forall \mathbf{x}' = R_e(\mathbf{x}) \;\; \forall \mathbf{x} \in G_e, \; \forall e = (\ell, \ell') \in E$,
(iv) $B_\ell(\mathbf{x}) < 0 \, \forall \mathbf{x} \in X_u(\ell)$,

then $B_\ell(\mathbf{x})$ *is a barrier certificate at the location* ℓ, *and the safety of the system* $\mathbf{H}$ *is guaranteed.*

Proof. By condition (i), $B_{\ell_0}(\mathbf{x})$ is nonnegative on Θ. Condition (ii) indicates that $\dot{B}_\ell(\mathbf{x}) > 0$ if $B_\ell(\mathbf{x}) > 0$, thus yielding that $B_\ell(\mathbf{x})$ is always nonnegative during the continuous flow. Since γ_e is nonnegative, condition (iii) guarantees that $B_\ell(\mathbf{x})$ cannot become negative during every discrete transition. Moreover, condition (iv) shows that all reachable states of $\mathbf{H}$ cannot intersect with the unsafe region X_u. □

Remark 1. Our verification conditions of barrier certificates in Theorems 1 and 2 are also called as the polynomial-scale consecution of the inductive invariants defined in [23], which is less conservative than the constant-scale consecution given in [13].

3.2 Computation of Barrier Certificates

In this section, we consider how to construct barrier certificates given in Theorems 1 and 2 for semi-algebraic dynamical systems. Investigating Theorems 1 and 2, it turns out that all verification conditions can be encoded as nonnegativity constraints for polynomials over the corresponding semi-algebraic sets. For the given degree bound, one may construct the template of the barrier polynomial $B_\ell(\mathbf{x})$ whose coefficients are parameters. In this case, our objective is to find real-valued coefficients of $B_\ell(\mathbf{x})$, satisfying the verification conditions, which is a typical quantifier elimination with polynomial equalities and inequalities constraints. Some symbolic methods, such as QEPCAD [7] and REDLOG [10] are available to offer mathematical proofs of the existence of the barrier certificate, at the cost of high computational complexity. To alleviate this computational intractability, we can apply SOS relaxation based approach [16] to compute $B_\ell(\mathbf{x})$, which starts with sufficient verification conditions by means of SOS representations, proceeds by dealing with SDP. Remark that SDP primarily relies on numerical interior-point SDP solvers running in fixed precision.

These limits may prevent the SOS relaxation based method from yielding valid $B_\ell(\mathbf{x})$. This paper follows another route: rather than applying SOS representations, we offer an alternative one for the nonnegativity of polynomials over compact semi-algebraic sets, and take advantage of this representation to propose new sufficient verification conditions for building the barrier certificates of dynamical systems. Notably, benefited from the above strategy, safety verification of dynamical systems can be converted into a tractable linear programming.

Let $\mathbb{K}$ be a compact semi-algebraic set defined by:

$$\mathbb{K} = \{\mathbf{x} \in \mathbb{R}^n \mid g_1(\mathbf{x}) \geq 0, \ldots, g_m(\mathbf{x}) \geq 0\}, \tag{2}$$

where $g_j(\mathbf{x}) \in \mathbb{R}[\mathbf{x}]$ for $j = 1, \cdots, m$. Since $\mathbb{K}$ is compact, one may compute $g^* := \max_{\mathbf{x}\in\mathbb{K}} g_j(\mathbf{x})$ for every $j = 1, \ldots, m$. Let $\tilde{g}_j(\mathbf{x})$ be the normalized polynomial of $g_j(\mathbf{x})$ with respect to $\mathbb{K}$, namely,

$$\tilde{g}_j(\mathbf{x}) = \begin{cases} g_j(\mathbf{x})/g^*, & \text{if} \quad g^* > 0, \\ g_j(\mathbf{x}), & \text{if} \quad g^* = 0. \end{cases} \tag{3}$$

For convenience, we introduce the following polynomial vector notation. Given a compact semi-algebraic set (2) with polynomials $g_1, \ldots, g_m$, denote by $\tilde{\boldsymbol{g}}$ the polynomial vector:

$$\tilde{\boldsymbol{g}} = [\tilde{g}_1, \ldots, \tilde{g}_m, 1 - \tilde{g}_1, \ldots, 1 - \tilde{g}_m]^T, \tag{4}$$

and $\tilde{\boldsymbol{g}}^\alpha$ stands for the polynomial product of the form:

$$\tilde{\boldsymbol{g}}^\alpha = \prod_{j=1}^{m} \tilde{g}_j^{\alpha_j} (1 - \tilde{g}_j)^{\alpha_{m+j}}, \tag{5}$$

where $\alpha \in \mathbb{N}^{2m}$.

Now we recap an alternative representation of a nonnegative polynomial on the compact semi-algebraic set.

Theorem 3 *(Krivine-Vasilescu-Handelman's(KVH) Positivstellensatz)[14]. Let $\mathbb{K}$ be a compact semi-algebraic set as in (2), and let $\tilde{g}_j(\mathbf{x})$ be the normalized polynomial $g_j(\mathbf{x})$ as in (3) for each j. Suppose the family $\{g_j, (1-g_j)\}_{j=0}^m$ generate $\mathbb{R}[\mathbf{x}]$ where $g_0 \equiv 1$. If $f(\mathbf{x}) \in \mathbb{R}[\mathbf{x}]$ is strictly positive on $\mathbb{K}$, then $f(\mathbf{x})$ can be represented as*

$$f(\mathbf{x}) = \sum_{\alpha \in \mathbb{N}^{2m}} c_\alpha \tilde{\boldsymbol{g}}^\alpha \tag{6}$$

where $c_\alpha \in \mathbb{R}_{\geq 0}$.

Remark 2. Following [14], if the polynomials $\{g_j, 1-g_j\}_{j=0}^m$ cannot generate $\mathbb{R}[\mathbf{x}]$, one can augment some linear functions such that the updated set of polynomials can generate $\mathbb{R}[\mathbf{x}]$. To be more precise, let $\underline{x}_k \leq \min\{x_k | \mathbf{x} \in \mathbb{K}\}$ for all $k = 1, \cdots, n$. Then, with $\mathbf{x} \mapsto g_{m+k}(\mathbf{x}) := x_k - \underline{x}_k$, the updated $\mathbb{K}$ can generate $\mathbb{R}[\mathbf{x}]$ by plugging the (redundant) constraints $g_{m+k} \geq 0$, $k = 1, \cdots, n$. Consider $\mathbb{K}$ is compact, lower bounds $\{\underline{x}_k\}$ on x_k can be obtained or are known. For more details, the reader refers to [14].

Assumption 1. *For every compact semi-algebraic set $\mathbb{K}$ in this paper, the polynomials $\{g_j\}_{j=0}^m$ can generate $\mathbb{R}[\mathbf{x}]$, where $g_1 \geq 0, \ldots, g_m \geq 0$ are the inequalities of $\mathbb{K}$ as in (2).*

From Theorem 3, the existence of the representation as in (6) provides a sufficient and necessary condition for the strict positiveness of $f(\mathbf{x})$ on the compact set $\mathbb{K}$. However, the number of the polynomial products in (6) is infinite, which means that generating its representation is computationally hard. To illustrate the computational applicability, we turn to selecting partial polynomial products in the representation (6) by fixing a priori (much smaller) degree bound D, in the following way. For the given positive integer $D \in \mathbb{Z}_{>0}$, we pick $\alpha \in \mathbb{N}^{2m}$ such that $\deg(\tilde{g}^\alpha) \leq D$. This strategy gives a sufficient condition for the nonnegativity of the given polynomial on the compact semi-algebraic set.

Theorem 4. *Let $\mathbb{K}$ be a compact semi-algebraic set as in (2), and let D be a positive integer. Let $\tilde{g}_j(\mathbf{x})$ be the normalized polynomial $g_j(\mathbf{x})$ as in (3) for each j. If $f(\mathbf{x}) \in \mathbb{R}[\mathbf{x}]$ can be written as*

$$f(\mathbf{x}) = \sum_{\deg(\tilde{\boldsymbol{g}}^\alpha) \leq D} c_\alpha \, \tilde{\boldsymbol{g}}^\alpha \text{ with } c_\alpha \geq 0, \tag{7}$$

then $f(\mathbf{x})$ is nonnegative on $\mathbb{K}$.

Proof. $\tilde{g}_j(\mathbf{x})$ is the normalized polynomial with respect to $\mathbb{K}$, which follows that $\tilde{g}_j(\mathbf{x})$ and $1 - \tilde{g}_j(\mathbf{x})$ are nonnegative on $\mathbb{K}$ for each j. The desired result can be easily obtained from $c_\alpha \geq 0$. □

The representation (7) ensures that $f(\mathbf{x})$ is nonnegative on $\mathbb{K}$. Observing the verification conditions in Theorems 1 and 2, we can see that all conditions can be rewritten as a unified type, namely, the nonnegativity of polynomials on the

compact semi-algebraic set. From Theorem 4, the original verification conditions can be relaxed as more tractable ones by the representations as (7). Let us now demonstrate by an example on how to convert the verification condition into the associated nonnegative representation.

Example 1. Consider the first verification condition in Theorem 1, $B(\mathbf{x}) \geq 0\,\forall \mathbf{x} \in \Theta$. Let $\tilde{\theta}_j(\mathbf{x})$ be the normalized polynomial of $\theta_j(\mathbf{x})$ with respect to Θ for each $j, 1 \leq j \leq q$. Let $\tilde{\boldsymbol{\theta}}$ be the normalized polynomial vector

$$\tilde{\boldsymbol{\theta}} = [\tilde{\theta}_1, \cdots, \tilde{\theta}_q, 1-\tilde{\theta}_1, \ldots, 1-\tilde{\theta}_q]^T.$$

Following Theorem 4, $B(\mathbf{x}) \geq 0\,\forall \mathbf{x} \in \Theta$ can be converted into the conservative one with the given degree bound $D \in \mathbb{Z}_{>0}$, namely,

$$B(\mathbf{x}) = \sum_{\deg(\tilde{\boldsymbol{\theta}}^{\alpha}) \leq D} c_\alpha \tilde{\boldsymbol{\theta}}^{\alpha},\ c_\alpha \in \mathbb{R}_{\geq 0} \quad \Longrightarrow \quad B(\mathbf{x}) \geq 0\,\forall \mathbf{x} \in \Theta. \qquad \square$$

As demonstrated in Example 1, we next provide a more tractable verification condition for the barrier certificates of continuous systems and hybrid systems. For notational convenience, throughout the rest of this paper, we will use $\tilde{\boldsymbol{\theta}}$, $\tilde{\boldsymbol{\psi}}$, $\tilde{\boldsymbol{\zeta}}$ to denote the normalized polynomial vectors with respect to Θ, Ψ and X_u, respectively.

Theorem 5. *Let* $\mathbf{S} : \langle \Theta, \mathbf{f}, \Psi \rangle$ *be a semi-algebraic continuous system, and* X_u *be the given unsafe state set. Let* D *be a positive integer. If there exist* $B(\mathbf{x}), \lambda(\mathbf{x}) \in \mathbb{R}[\mathbf{x}]$*, which satisfy the following conditions:*

1. $B(\mathbf{x}) = \sum_{\deg(\tilde{\boldsymbol{\theta}}^{\alpha}) \leq D} c_\alpha \tilde{\boldsymbol{\theta}}^{\alpha},\ c_\alpha \geq 0,$
2. $\dot{B}(\mathbf{x}) - \lambda(\mathbf{x}) B(\mathbf{x}) - \epsilon_1 = \sum_{\deg(\tilde{\boldsymbol{\psi}}^{\beta}) \leq D} c_\beta \tilde{\boldsymbol{\psi}}^{\beta},\ c_\beta \geq 0,\ \epsilon_1 > 0,$
3. $-B(\mathbf{x}) - \epsilon_2 = \sum_{\deg(\tilde{\boldsymbol{\zeta}}^{\omega}) \leq D} c_\omega \tilde{\boldsymbol{\zeta}}^{\omega},\ c_\omega \geq 0,\ \epsilon_2 > 0,$

then the safety of the system $\mathbf{S}$ *is guaranteed.*

Proof. Theorem 4 indicates that the conditions (1–3) can imply the conditions (i–iii) in Theorem 1, respectively. Thus, the safety of the system $\mathbf{S}$ is proved. $\square$

Theorem 6. *Let* $\mathbf{H} : \langle L, X, F, \Psi, E, G, R, \Theta, \ell_0 \rangle$ *be a semi-algebraic hybrid system,* $\mathcal{X}_u$ *be the unsafe assertion. Let* D *be a positive integer. If there exist* $B_\ell(\mathbf{x}), \lambda_\ell(\mathbf{x}) \in \mathbb{R}[\mathbf{x}]$ *for each* $\ell \in L$*, and nonnegative polynomial* $\gamma_e(\mathbf{x}) \in \mathbb{R}[\mathbf{x}]$ *for each* $e \in E$*, which satisfy*

1. $B_{\ell_0}(\mathbf{x}) = \sum_{\deg(\tilde{\boldsymbol{\theta}}^{\alpha_{\ell_0}}) \leq D} c_{\alpha_{\ell_0}} \tilde{\boldsymbol{\theta}}^{\alpha_{\ell_0}},\ c_{\alpha_{\ell_0}} \geq 0\ ,$
2. $\dot{B}_\ell(\mathbf{x}) - \lambda_\ell(\mathbf{x}) B_\ell(\mathbf{x}) - \epsilon_{\ell,1} = \sum_{\deg(\tilde{\boldsymbol{\psi}}^{\beta_\ell}) \leq D} c_{\beta_\ell} \tilde{\boldsymbol{\psi}}_\ell^{\beta_\ell},\ c_{\beta_\ell} \geq 0,\ \epsilon_{\ell,1} > 0,$

3. $B_{\ell'}(R_e(\mathbf{x})) - \gamma_e(\mathbf{x})B_\ell(\mathbf{x}) = \sum_{\deg(\tilde{\boldsymbol{g}}^{\mu_e})\le D} c_{\mu_e}\tilde{\boldsymbol{g}}_e^{\mu_e},\ c_{\mu_e} \ge 0,$

4. $-B_\ell(\mathbf{x}) - \epsilon_{\ell,2} = \sum_{\deg(\tilde{\boldsymbol{\zeta}}^{\omega_\ell})\le D} c_{\omega_\ell}\,\tilde{\boldsymbol{\zeta}}_\ell^{\omega_\ell},\ c_{\omega_\ell} \ge 0,\ \epsilon_{\ell,2} > 0,$

then the safety of the system $\mathbf{H}$ *is guaranteed.*

Proof. Simiar to the proof of Theorem 5. □

Theorems 5 and 6 produce the sufficient conditions for generating the barrier certificates of continuous and hybrid systems, respectively. With unknown multipliers $\lambda(\mathbf{x})$, $\lambda_\ell(\mathbf{x})$, $\gamma_e(\mathbf{x})$ and unknown barrier certificates $B(\mathbf{x})$, $B_\ell(\mathbf{x})$, some nonlinear terms that are products of the coefficients of unknown polynomials will occur in the constraints in Theorems 5 and 6, which yields a non-convex bilinear matrix inequalities (BMI) problem. To alleviate this computational intractability, provided that the multipliers $\lambda_\ell(\mathbf{x})$ and $\gamma_e(\mathbf{x})$ are given in advance, the problem of generating the above barrier certificates can be transformed into the linear programming problem. To keep it concise, we only sketch the case of continuous systems, but the transformation procedure extends to the case of hybrid systems without much difficulty.

To start with, a key step is to parameterize $B(\mathbf{x})$ and the power products associated to each expression in the conditions (1–3) of Theorem 5. For the given degree d of $B(\mathbf{x})$, we first predetermine a template of $B(\mathbf{x})$ by setting its coefficients as parameters, i.e., $B(\mathbf{x}) = \sum_\alpha b_\iota \mathbf{x}^\iota$, where $\mathbf{x}^\iota = x_1^{\iota_1}\cdots x_n^{\iota_n}$, $\iota = (\iota_1,\ldots,\iota_n) \in \mathbb{Z}_{\ge 0}^n$ with $\sum_{i=1}^n \iota_i \le d$, and b_ι's are unknown coefficients. Let $\mathbf{b}$ be the coefficient vector of $B(\mathbf{x})$. In the sequel, we write $B(\mathbf{x})$ as $B(\mathbf{x},\mathbf{b})$ for clarity. Denote by $\mathbf{c}_\alpha, \mathbf{c}_\beta, \mathbf{c}_\omega$ the parameter vectors appearing in the conditions (1-3) of Theorem 5, respectively, and let $\mathbf{c} = [\mathbf{c}_\alpha^T, \mathbf{c}_\beta^T, \mathbf{c}_\omega^T]^T$. For the given degree bound D, it follows from Theorem 5 that generating a barrier certificate can be transformed into the following optimization problem:

$$\left.\begin{aligned}
&\text{find}\ \ \mathbf{b}\\
&\text{s.t.}\ \ B(\mathbf{x},\mathbf{b}) = \sum_{\deg(\tilde{\theta}^\alpha)\le D} c_\alpha\,\tilde{\boldsymbol{\theta}}^\alpha,\\
&\qquad \dot{B}(\mathbf{x},\mathbf{b}) - \lambda(\mathbf{x})B(\mathbf{x},\mathbf{b}) - \epsilon_1 = \sum_{\deg(\tilde{\boldsymbol{\psi}}^\beta)\le D} c_\beta\,\tilde{\boldsymbol{\psi}}^\beta,\\
&\qquad -B(\mathbf{x},\mathbf{b}) - \epsilon_2 = \sum_{\deg(\tilde{\boldsymbol{\zeta}}^\omega)\le D} c_\omega\,\tilde{\boldsymbol{\zeta}}^\omega,\\
&\qquad c_\alpha, c_\beta, c_\omega \ge 0,
\end{aligned}\right\} \tag{8}$$

where $\lambda(\mathbf{x})$ is a prespecified polynomial, and $\epsilon_1, \epsilon_2 \in \mathbb{R}_{>0}$ are prespecified small positive numbers. We can rewrite the equality constraints in (8) as a linear system with the variables $\mathbf{b}, \mathbf{c}$ by sorting the coefficients with respect to the variables $\mathbf{x}$. By doing so, (8) is equivalent to the following linear programming problem:

$$\left.\begin{aligned}
&\text{find}\ \ \mathbf{y}\\
&\quad \text{s.t.}\ A\cdot\mathbf{y} \ge 0,
\end{aligned}\right\} \tag{9}$$

where $\mathbf{y} = (\mathbf{b}^T, \mathbf{c}^T)^T$ and A is a numerical matrix. Problem (9) can be solved by using conventional algorithms such as the interior-point method [6]. If (9) is feasible, the result yields a barrier certificate $B(\mathbf{x})$, which suffices to verify the safety of the continuous system $\mathbf{S}$. Our LP relaxation is based on the predetermined degree bound D. Once (9) is infeasible, one may improve the relaxation precision and then increase the possibility to find the barrier certificate by increasing the degree bound D. Detailed procedures are summarized in Algorithm 1.

Remark 3. Theorem 6 guarantees that λ_ℓ can be any constants or polynomials, and γ_e can be any nonnegative constants or polynomials. To ease computation, one prefers to set them as simple as possible. Here we choose λ_ℓ from $0, \pm 1, \pm(1+x_1^2+\cdots+x_n^2)$, and γ_e from $0, 1, 1+x_1^2+\cdots+x_n^2$, respectively. Like computing the fractional SOS representations of nonnegative polynomials, one may also choose the denominator as $(1+x_1^2+\cdots+x_n^2)^k$ for some integer exponent k.

Algorithm 1. Search for polynomial barrier certificates

Input: Semi-algebraic continuous system $\mathbf{S}$, or hybrid system $\mathbf{H}$; the degree bound d of the barrier certificate; the degree bound D of the representation (7).

Output: The barrier certificate $\{B_\ell(\mathbf{x})\}$.

```
forall the ℓ ∈ L do
    Parameterized B_ℓ(x) by polynomials of degree d
Construct the power-products with degree D of the polynomials defining the
semi-algebraic sets in Theorem 2.
Set up the linear programming of the form (9) and apply an LP solver to
compute its solutions.
if the problem (9) is feasible then
    return {B_ℓ(x)}.
else
    return "we cannot find the barrier certificates with the degree bound d."
```

Remark 4. Like SOS relaxation method, our method cannot guarantee that the polynomial barrier certificates will always be found due to the limitation on presetting the degree bounds d and D. It is also difficult to predetermine whether such polynomial barrier certificates exist. Therefore, if our algorithm fails to yield any barrier certificate, it does not mean that the given hybrid system has no polynomial barrier certificates with the given degree bound, or that the given system is unsafe.

3.3 Complexity Analysis

In the section, we analyze the complexity of Algorithm 1, and further compare it with SOS relaxation method. Let n, $|L|$ and $|E|$ be the numbers of

system variables, locations and discrete transitions in the given hybrid system $\mathbf{H}$, respectively. And let $d_{\mathbf{f}}$ be the maximal degree of the polynomial vector fields of $\mathbf{H}$, and let d_v be the maximal degree among the polynomial lists, which are used to define the compact semi-algebraic sets appearing in $\mathbf{H}$. The linear programming problem (9) implies the predetermined degree bound D must satisfy $D \geq d + d_{\mathbf{f}}$. Suppose that the numbers of the polynomials defining the compact semi-algebraic sets in $\mathbf{H}$ are bounded by s_p. Therefore, the number of the decision variables, denoted by $\mathcal{V}_l$, of the linear programming problem (9) is

$$\mathcal{V}_l = \binom{n+d}{d} + (1 + 2|L| + |E|)\binom{2s_p + D}{D}, \tag{10}$$

where the first term is the number of coefficients $\mathbf{b}$, and the second one is the number of coefficients $\mathbf{c}$. Meanwhile, the number of constraints, denoted by $\mathcal{C}_l$, in (9) is

$$\mathcal{C}_l = \binom{n+D}{D} + (1 + 2|L| + |E|)\binom{2s_p + D}{D}, \tag{11}$$

where the first term is the number of equality constraints associated with coefficients $\mathbf{b}$ and $\mathbf{c}$, and the second one is the number of nonnegative constraints of coefficients $\mathbf{c}$.

As is well known, the complexity of an LP using interior-point algorithms is approximately $O(\mathcal{V}_l^2\,\mathcal{C}_l)$ [6]. Taking this together with (10) and (11), we get the complexity of Algorithm 1 based on the LP solver is approximately $O(n^{2d+D})$.

We also called the SOS relaxation (cf. (36)–(39) in [16]) to search for the barrier certificate of the hybrid system $\mathbf{H}$. Similarly, let D be the predetermined degree bound for all involved SOS polynomial multipliers. Then, the number of decision variables, denoted by $\mathcal{V}_s$, in the SDP associated with the SOS relaxation is

$$\mathcal{V}_s = (s_p + 1)\,(1 + 2|L| + |E|)\,\frac{N(N+1)}{2}, \tag{12}$$

where $N = \binom{n+D/2}{D/2}$ is the number of monomials in a polynomial of degree $D/2$. Meanwhile, the number of constraints, denoted by $\mathcal{C}s$, in the SDP associated with the SOS relaxation is

$$\mathcal{C}_s = (1 + 2|L| + |E|)\binom{n+D}{D}. \tag{13}$$

It is known that the complexity of SDP-solving via interior-point algorithms is approximately $O(\mathcal{C}_s^3 + \mathcal{V}_s^3\mathcal{C}_s + \mathcal{C}_s^2\mathcal{V}_s^2)$ [6]. From (12) and (13), we get the complexity of calling the SDP solver to search for a barrier certificate is approximately $O(n^{4D})$.

4 Experiments

In this section, we first demonstrate the application of our methods by two examples and then compare our LP relaxation method with the SOS relaxation

method with respect to ability and efficiency on 10 examples. We used examples of high computational complexity from related works in the experiments [4,5,8, 16–19,22,24].

Example 2. Consider the following nonlinear continuous system [22]

$$\begin{bmatrix} \dot{x_1} \\ \dot{x_2} \\ \dot{x_3} \\ \dot{x_4} \end{bmatrix} = \begin{bmatrix} -x_1 + x_2^3 - 3x_3x_4 \\ -x_1 - x_2^3 \\ x_1x_4 - x_3 \\ x_1x_3 - x_4^3 \end{bmatrix},$$

with the location invariant $\Psi = \{\mathbf{x} \in \mathbb{R}^4 : -1 \leq x_1, x_2, x_3, x_4 \leq 1\}$. We will verify that all trajectories of the system starting from the initial set $\Theta = \{\mathbf{x} \in \mathbb{R}^4 : 0 \leq x_1, x_2, x_3, x_4 \leq 0.5\}$. will never enter the unsafe set $X_u = \{\mathbf{x} \in \mathbb{R}^4 : -1 \leq x_1, x_2, x_3, x_4 \leq -0.5\}$.

Let the degree bound D of the representation (7) be 8, and $\lambda(\mathbf{x})$ in Theorem 5 be 1, respectively. Our algorithm succeeds to yield the barrier certificate

$$B(\mathbf{x}) = \underbrace{-12.9713x_1^4 - 16.6808x_2^4 - 93.4687x_3^4 - 0.5426x_4^4 + \ldots + 779.0477}_{70 \quad terms}.$$

Therefore, the safety of the above system is verified. □

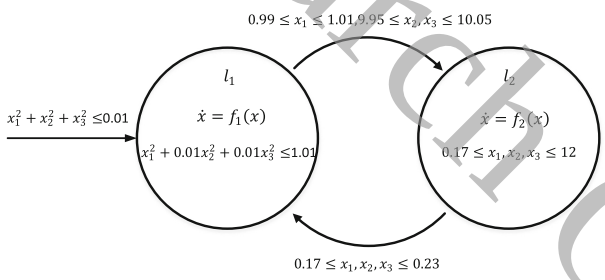

Fig. 1. The hybrid automata of the system in Example 3

Example 3. Consider the hybrid automata of the system depicted in Fig. 1, where

$$\mathbf{f}_1(\mathbf{x}) = \begin{bmatrix} -x_2 \\ -x_1 + x_3 \\ x_1 + (2x_2 + 3x_3)(1 + x_3^2) \end{bmatrix}, \quad \mathbf{f}_2(\mathbf{x}) = \begin{bmatrix} -x_2 \\ -x_1 + x_3 \\ -x_1 - 2x_2 - 3x_3 \end{bmatrix}.$$

Our task is to verify that the system will never enter the unsafe set

$$X_u(\ell_2) = \{\mathbf{x} \in \mathbb{R}^3 : x_1 \geq 5\}.$$

Let the degree bound D of the representation (7) be 6, and $\lambda_{\ell_1} = -0.2$, $\lambda_{\ell_2} = 0$, and $\gamma_{(\ell_1,\ell_2)} = \gamma_{(\ell_2,\ell_1)} = 1$. Applying Algorithm 1, we obtain the polynomial barrier certificate with degree 2:

$$B_{\ell_1}(\mathbf{x}) = \underbrace{-48.0832x_1^2 - 0.6225x_2^2 - 0.0005x_3^2 + \cdots + 1075.8714}_{10 \quad terms},$$
$$B_{\ell_2}(\mathbf{x}) = \underbrace{-0.8002{x_1}^2 + 0.4692{x_2}^2 + 0.5978x_3^2 + \cdots + 423.7896}_{10 \quad terms}.$$

Meanwhile, we apply the SOS relaxation based method to compute a barrier certificate with degree < 4. However, the SDP solver cannot return any barrier certificate. As discussed above, our LP relaxation based approach can find the barrier certificate that the SOS relaxation based method cannot yield. □

We compared our LP relaxation based method with the SOS relaxation based one over a set of benchmarks gathered from the related works. Table 1 shows the result. Here, the LP problems were settled by the *linprog* command in Matlab while the SDP problems were solved by the Matlab toolbox *SeDuMi* [29]. The experiments were performed on Intel(R) Core(TM) at 2.60 GHz with 8 GB of memory under Windows 8.

In Table 1, n and $|L|$ denote the number of the system variables and the number of the locations; $d_{\mathbf{f}}$ denotes the maximal degree of the polynomials in the vector fields; $d_l(B)$ and $d_s(B)$ denote the degrees of the barrier certificates obtained from LP and SDP solvers, respectively; D_l and D_s are the degree bounds of the nonnegative representation derived from the LP relaxation and SOS relaxation, respectively; $\mathcal{V}_l$ and $\mathcal{V}_c$ denote the numbers of the decision variables of the LP and SDP, respectively; T_l and T_s represent the entire computation times in seconds spent by LP and SDP solvers, respectively.

Table 1. Algorithm performance on benchmarks

Examples	n	$\|L\|$	$d_{\mathbf{f}}$	LP				SDP			
				$d_l(B)$	D_l	$\mathcal{V}_l$	$T_l(s)$	$d_s(B)$	D_s	$\mathcal{V}_s$	$T_s(s)$
Ex.1 from [18]	2	1	2	3	6	164	0.0221	3	6	292	0.1870
Ex.2 from [16]	2	1	3	4	6	328	0.0782	4	8	597	0.1179
Ex.3 from [8]	2	1	3	2	4	91	0.0140	2	6	299	0.1129
Ex.4 from [18]	2	1	1	3	4	129	0.0053	4	6	287	0.1193
Ex.5 from [18]	2	1	2	2	4	56	0.0073	2	4	95	0.1358
Ex.6 from [24]	3	1	2	4	6	917	0.1051	4	6	942	0.2187
Ex.7 from [19]	3	1	3	4	6	1379	0.1444	4	8	2977	0.2421
Ex.8 from [5]	3	1	2	4	6	890	0.0966	1	4	225	0.1815
Ex.9 from [4]	2	3	1	2	2	156	0.0941	2	2	278	0.1820
Ex.10 from [17]	3	2	3	2	4	370	0.0331	4	8	5952	0.8481

For 7 of the examples, both LP relaxation and SOS relaxation can successfully find the barrier certificates of polynomial forms with the same degree. However, as discussed in the Sect. 3.3, the number of decision variables in LP relaxation is much smaller than that in SOS relaxation. Plus the more efficiency LP solvers provide, our LP relaxation based method is much more efficient than the SOS relaxation method. For Ex.4 and Ex.10, SOS relaxation based method cannot find polynomial barrier certificates whose degrees are less than 4, whereas our LP relaxation method can yield two barrier certificates with the degrees 3 and 2, respectively. Ex.8 displays the opposite case where SOS relaxation performs better.

In fact, LP relaxation and SOS relaxation use different sufficient conditions for polynomial positivity and give different encodings of barrier certificate generation. Theoretically, for the given degree bound of the polynomial, there are cases where the SOS relaxation can find a barrier certificate, however, the LP relaxation cannot, and vice versa. Even for the cases that can be solved by both of them, there is no theoretical result predicting which method will produce barrier certificates of lower-degree. Thus, our LP relaxation based method and the SOS relaxation based method complement each other.

5 Related Work

A barrier certificate is a special kind of inductive invariant, thus research on safety verification using inductive invariants is related to our work. Sankaranarayanan et al. presented methods adopting the ideal theory over polynomial rings and quantifier elimination to automatically generate algebraic invariants for algebraic hybrid systems [21,23]. Sturm and Tiwari presented the application of quantifier elimination to formal verification and synthesis of continuous and switched dynamical systems [30]. Based on Gröbner basis manipulations, Rodríguez-Carbonell constructed polynomial invariants (a set of polynomial equations) for linear hybrid systems [20]. Platzer et al. adopted iterative fixedpoint calculations to find differential invariants, a boolean combination of multiple polynomial inequalities, to verify semi-algebraic hybrid systems [15]. Gulwani et al. defined a similar invariant with a different inductive condition and used the Farkas's theory and SMT solvers to solve it [11]. Sogokon et al. combined semi-algebraic abstractions with deductive verification method to generate semi-algebraic invariants for polynomial continuous systems [28].

For barrier certificates with convex conditions, the technique of sum-of-squares decomposition of semidefinite polynomials provides much better efficiency and thus is quite popular. Prajna et al. generated barrier certificates for semi-algebraic hybrid systems [16,17]. Kong et al. proposed a method to generate a barrier certificate defined over an exponential condition for semi-algebraic hybrid systems by SDP [13]. Dai et al. utilized different weaker conditions flexibly to synthesize different kinds of barrier certificates with more expressiveness efficiently using SDP [9]. Yang et al. presented a hybrid symbolic-numeric method to compute the exact inequality invariants of polynomial hybrid systems via

SOS relaxation [31]. Sloth et al. proposed compositional conditions for barrier certificates to verify the safety property of a group of interconnected hybrid systems [27].

LP relaxation based techniques have been successfully applied in stability analysis of nonlinear systems. Ahmadi et al. introduced two different positive representations: DSOS and SDSOS to take the place of SOS, and combined linear programming and second order cone programming to solve them [1]. Sankaranarayanan et al. investigated the stability of continuous systems with polyhedral domains. They used the Handelman positive representation to synthesize Lyapunov functions [22,25]. Ben Sassi et al. used polyhedra templates to analyse the reachability of polynomial systems. They reduced the problem of reachability analysis to a set of optimization problems involving polynomials over bounded polyhedra, then adopted the Bernstein expansions of polynomials to build LP relaxations [26]. In the paper, we treat the more general semi-algebraic hybrid systems and generate barrier certificates using KVH positivstellensatz. It is the first attempt to use LP relaxation for computing barrier certificates.

6 Conclusion

We have presented a linear programming (LP) relaxation based approach for generating barrier certificates of semi-algebraic hybrid systems. The main feature of this approach is that it uses an LP relaxation to encode the set of nonnegativity constraints associated with the barrier certificates. Thanks to the low computational complexity and the high numerical stability of LP, our approach is more efficient than the popular SOS relaxation based methods when treating barrier certificates with convex conditions. The conclusion is supported by a theoretical analysis on complexity and the experiments taken on a set of benchmarks gathered from the literature.

Acknowledgments. This material is supported in part by Key Basic Research Program of China (Grant No. 2014CB340703), the National Natural Science Foundation of China (Grant Nos. 61321064, 61361136002, 11471209, 11571350, 61672435, 61561146394, 91318301 and 61602348), the Innovation Program of Shanghai Municipal Education Commission (Grant No. 14ZZ046), the project on the Integration of Industry, Education and Research of Jiangsu Province (Grant No. BY2014126-03), the project SWU116007 funded by Southwest University. We would like to thank anonymous reviewers for their very valuable comments.

References

1. Ahmadi, A.A., Majumdar, A.: Dsos and sdsos optimization: Lp and socp-based alternatives to sum of squares optimization. In: 2014 48th Annual Conference on Information Sciences and Systems (CISS), pp. 1–5. IEEE (2014)
2. Alur, R.: Formal verification of hybrid systems. In: Proceedings of the International Conference on Embedded Software (EMSOFT), pp. 273–278. IEEE (2011)

3. Alur, R., Courcoubetis, C., Halbwachs, N., Henzinger, T.A., Ho, P.H., Nicollin, X., Olivero, A., Sifakis, J., Yovine, S.: The algorithmic analysis of hybrid systems. Theor. Comput. Sci. **138**(1), 3–34 (1995)
4. Alur, R., Dang, T., Ivančić, F.: Predicate abstraction for reachability analysis of hybrid systems. ACM Trans. Embed. Comput. Syst. (TECS) **5**(1), 152–199 (2006)
5. Bouissou, O., Chapoutot, A., Djaballah, A., Kieffer, M.: Computation of parametric barrier functions for dynamical systems using interval analysis. In: 2014 IEEE 53rd Annual Conference on Decision and Control (CDC), pp. 753–758. IEEE (2014)
6. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press, Cambridge (2004)
7. Brown, C.W.: QEPCAD B: a program for computing with semi-algebraic sets using CADs. ACM SIGSAM Bull. **37**(4), 97–108 (2003)
8. Chen, X., Abraham, E., Sankaranarayanan, S.: Taylor model flowpipe construction for non-linear hybrid systems. In: Proceedings of the 2012 IEEE 33rd Real-Time Systems Symposium (RTSS), pp. 183–192. IEEE (2012)
9. Dai, L., Gan, T., Xia, B., Zhan, N.: Barrier certificates revisited. To appear J. Symbolic Comput. (2016)
10. Dolzmann, A., Sturm, T.: Redlog: computer algebra meets computer logic. ACM Sigsam Bull. **31**(2), 2–9 (1997)
11. Gulwani, S., Tiwari, A.: Constraint-based approach for analysis of hybrid systems. In: Proceedings of the 20th International Conference on Computer Aided Verification (CAV), pp. 190–203 (2008)
12. Henzinger, T.A.: The theory of hybrid automata. In: Proceedings of the 11th Annual IEEE Symposium on Logic in Computer Science, pp. 278–292. IEEE Computer Society (1996)
13. Kong, H., He, F., Song, X., Hung, W.N.N., Gu, M.: Exponential-condition-based barrier certificate generation for safety verification of hybrid systems. In: Sharygina, N., Veith, H. (eds.) CAV 2013. LNCS, vol. 8044, pp. 242–257. Springer, Heidelberg (2013). doi:10.1007/978-3-642-39799-8_17
14. Lasserre, J.B.: Polynomial programming: Lp-relaxations also converge. SIAM J. Optim. **15**(2), 383–393 (2005)
15. Platzer, A., Clarke, E.M.: Computing differential invariants of hybrid systems as fixedpoints. Form. Methods Syst. Des. **35**(1), 98–120 (2009)
16. Prajna, S., Jadbabaie, A., Pappas, G.: A framework for worst-case and stochastic safety verification using barrier certificates. IEEE Trans. Autom. Control **52**(8), 1415–1429 (2007)
17. Prajna, S., Jadbabaie, A.: Safety verification of hybrid systems using barrier certificates. In: Alur, R., Pappas, G.J. (eds.) HSCC 2004. LNCS, vol. 2993, pp. 477–492. Springer, Heidelberg (2004). doi:10.1007/978-3-540-24743-2_32
18. Ratschan, S., She, Z.: Safety verification of hybrid systems by constraint propagation-based abstraction refinement. ACM Trans. Embed. Comput. Syst. **6**(1), 573–589 (2007)
19. Ratschan, S., She, Z.: Providing a basin of attraction to a target region of polynomial systems by computation of lyapunov-like functions. SIAM J. Control Optim. **48**(7), 4377–4394 (2010)
20. Rodríguez-Carbonell, E., Tiwari, A.: Generating polynomial invariants for hybrid systems. In: Proceedings of the 8th ACM International Conference on Hybrid Systems: Computation and Control, pp. 590–605 (2005)
21. Sankaranarayanan, S.: Automatic invariant generation for hybrid systems using ideal fixed points. In: Proceedings of the 13th ACM International Conference on Hybrid Systems: Computation and Control, pp. 221–230. ACM (2010)

22. Sankaranarayanan, S., Chen, X., Abrahám, E.: Lyapunov function synthesis using handelman representations. In: The 9th IFAC Symposium on Nonlinear Control Systems, pp. 576–581 (2013)
23. Sankaranarayanan, S., Sipma, H., Manna, Z.: Constructing invariants for hybrid systems. Formal Methods Syst. Des. **32**, 25–55 (2008)
24. Sassi, M.A.B., Sankaranarayanan, S.: Stabilization of polynomial dynamical systems using linear programming based on bernstein polynomials. arXiv preprint arXiv:1501.04578 (2015)
25. Sassi, M.A.B., Sankaranarayanan, S., Chen, X., Ábrahám, E.: Linear relaxations of polynomial positivity for polynomial lyapunov function synthesis. IMA J. Math. Control Inform., 1–34 (2015). doi:10.1093/imamci/dnv003
26. Sassi, M.A.B., Testylier, R., Dang, T., Girard, A.: Reachability analysis of polynomial systems using linear programming relaxations. In: Chakraborty, S., Mukund, M. (eds.) ATVA 2012. LNCS, vol. 7561, pp. 137–151. Springer, Heidelberg (2012)
27. Sloth, C., Pappas, G.J., Wisniewski, R.: Compositional safety analysis using barrier certificates. In: Proceedings of the 15th ACM International Conference on Hybrid Systems: Computation and Control, pp. 15–24. ACM (2012)
28. Sogokon, A., Ghorbal, K., Jackson, P.B., Platzer, A.: A method for invariant generation for polynomial continuous systems. In: Jobstmann, B., Leino, K.R.M. (eds.) VMCAI 2016. LNCS, vol. 9583, pp. 268–288. Springer, Heidelberg (2016). doi:10.1007/978-3-662-49122-5_13
29. Sturm, J.F.: Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. Optim. Methods Softw. **11**(12), 625–653 (1999)
30. Sturm, T., Tiwari, A.: Verification and synthesis using real quantifier elimination. In: Proceedings of the International Symposium on Symbolic and Algebraic Computation, ISSAC, pp. 329–336. ACM Press (2011)
31. Yang, Z., Wu, M., Lin, W.: Exact verification of hybrid systems based on bilinear SOS representation. ACM Trans. Embed. Comput. Syst. **14**(1), 1–19 (2015)