A Computational Approach to Synthesizing Guards for Hybrid Systems

Samuel Coogan[a], Murat Arcak[a]

[a] *Department of Electrical Engineering and Computer Sciences, University of California, Berkeley*

Abstract

We propose a technique for synthesizing switching guards for hybrid systems to satisfy a given state-based safety constraint. Using techniques from sum of squares (SOS) optimization, we design guards defined by semialgebraic sets that trigger mode switches, and we guarantee that the synthesized switching policy does not allow Zeno executions. We demonstrate our approach on an example of switched affine systems and on an application to traffic ramp metering.

Keywords: Hybrid systems; guard synthesis; computational methods

1. Introduction

Hybrid systems have emerged as a powerful modeling paradigm for complex systems comprised of continuous and discrete components. Often, the discrete mode at any given time can be chosen by a controller. Examples of such systems include traffic networks where vehicle flow rate is modeled as a continuous-valued variable whose evolution is governed by discrete choices of intersection signals and ramp metering devices. An important task for such systems is to design a policy for switching among the modes to satisfy a *safety* property whereby the system is guaranteed not to enter an unsafe region of the state space (*e.g.* to maintain a certain traffic throughput or to prevent queues from growing too large).

Verifying safety properties and synthesizing safe control strategies for hybrid systems has received considerable attention, *e.g.*, [1], [2], [3], [4] [5], [6], [7], [8]. One common approach to the problem of controller synthesis is to calculate a controlled invariant set via an iterative algorithm [9]. The algorithm is initialized with the safe set and iteratively removes trajectories that may be forced to exit the set due to disturbance inputs or system dynamics, thereby eliminating choices for the discrete mode at each time. If the algorithm terminates at a fixed point, this final set is the maximal controlled invariant set, and a least restrictive controller is obtained as a byproduct of the computation [10]. Each step of the iteration requires computing *controllable* and *uncontrollable predecessors* and then solving a *reach-avoid* problem [9] from these predecessors. These subproblems are often formulated as the solution to a Hamilton-Jacobi (HJ) equation (or a pair of coupled HJ equations) [11, 9, 12].

The primary difficulty of such methods is that the solution of the HJ equations is, in general, computationally taxing. In addition, solution approaches often suffer from numerical difficulties caused by discontinuities in the Hamiltonian [11]. Finally, nearly all computational approaches, such as the prevalent *level-set method* [2], require numerical approximation whose accuracy must be considered. For example, if the numerical approximation is not contained within the maximal controlled invariant set, the synthesis algorithm may identify unsafe states as safe. For some classes of hybrid systems, solutions to HJ equations is efficiently computable [13], however the class of such systems is very limited.

In this work, we synthesize switching guards that ensure the hybrid system satisfies a state-based safety constraint using sum of squares (SOS) programming. We consider hybrid systems with a finite number of modes in which the state evolution is governed by a differential inclusion with no continuous control input, and we synthesize guards that trigger transitions between modes. Guards are assumed to be *semialgebraic* sets, *i.e.* a guard is a subset of the continuous state space which satisfies a collection of polynomial inequalities and equalities. Other applications of SOS programming to control theory include region-of-attraction analysis and Lyapunov function calculation, [14], [15], hybrid system verification, [6], and calculation of finite-time invariant regions, [16] (see [17] for an overview).

Our switching guard synthesis procedure relies on knowing the reach set from a given set in a particular mode, or at least an overapproximation of this set. Finding such sets can be difficult and is an active area of research. The focus of this paper is on classes of systems where the computation of reach sets is amenable to analytical or numerical procedures, and the difficulty in controller synthesis lies in choosing when to switch between discrete modes. We offer a synthesis procedure for this task which relies on SOS programming and demonstrate our approach on several examples.

Section 2 introduces notation and reviews hybrid systems and SOS programming. Section 3 states the problem formulation, and Section 4 presents the guard synthesis approach. In Section 5, we apply this method to onramp metering for freeway traffic control. We offer directions for future research in Section 6. This paper extends our conference paper [18]. Extensions include specializing our results to the case of switched affine systems and full development of an extensible application to freeway traffic control.

2. Preliminaries

2.1. Notation

The set $\mathbb{R}_{\geq 0}$ (resp. $\mathbb{R}_{\leq 0}$) is the set of nonnegative (resp. nonpositive) real numbers. For a set X, 2^X is the set of all subsets of X and $\mathbf{cl}(X)$ is the closure of X. For a vector v, $\mathrm{Dim}(v)$ is the dimension of v. The notation 0_n denotes the n-dimensional vector of zeros, and if the dimension is evident, the subscript is suppressed. We denote elementwise nonnegativity of a vector v by $v \succeq 0$. An asterisk $(*)$ used as a subscript denotes a placeholder to be replaced with elements from an index set which is evident from context.

2.2. Hybrid Systems

A *hybrid system* is a tuple $H = (Q, X, I, f, R, \mathcal{G})$ where the total state space $Q \times X$ consists of a finite set Q of *modes* and a continuous state space $X = \mathbb{R}^n$. The system is initialized in a set $I \subseteq Q \times X$, and we define $I(q) \triangleq \{x : (q, x) \in I\}$. We consider differential inclusions such that

$$\dot{x}(t) \in f(q, x(t)) \quad \text{for almost all } t \tag{1}$$

where $f(\cdot, \cdot) : Q \times X \to 2^X$ constrains the continuous evolution while in mode q. Mild assumptions on $f(q, \cdot)$ guarantee the existence and absolute continuity of solutions [19, §3.3]. In particular, we further assume $f(q, \cdot)$ is locally bounded. This formulation is general and can accommodate, for example, parameter uncertainty or disturbance inputs.

We define the reset map as follows: $R(\cdot, \cdot, \cdot) : Q \times Q \times X \to 2^X$ where $R(q, q', x) \subseteq X$ is the set of continuous states which can be reached when the system undergoes a transition from discrete state q to q' while at $x \in X$. We denote the domain of R for fixed q, q' by $\mathcal{R}_{q \to q'} \triangleq \mathrm{Dom}(R(q, q', \cdot)) \subseteq X$. For a set $M \subseteq \mathcal{R}_{q \to q'}$, we understand $R(q, q', M) \triangleq \bigcup_{x \in M} R(q, q', x)$. Note that if a transition from q to q' is not possible, then $\mathcal{R}_{q \to q'} = \emptyset$.

A *set of guards* $\mathcal{G}$ for a hybrid system is a collection of sets $\mathcal{G} = \{\mathcal{G}_{q \to q'}\}_{q, q' \in Q}$ such that

$$\mathcal{G}_{q \to q'} \subseteq \mathcal{R}_{q \to q'}. \tag{2}$$

Each $\mathcal{G}_{q \to q'}$ is called a *guard*, and if $x \in \mathcal{G}_{q \to q'}$, we say the guard from mode q to q' is *active*. Let

$$\mathcal{G}_q \triangleq \bigcup_{q' \in Q} \mathcal{G}_{q \to q'}. \tag{3}$$

The purpose of the guards is to trigger mode transitions and the corresponding reset of the continuous state dictated by the reset map. In this work, we consider synthesizing a set of guards so that the hybrid system satisfies a safety property.

An *execution* of a hybrid system H is a sequence of mode transition times $\{\tau_i\}_{i=1}^N$ with $\tau_0 = 0$, $\tau_i \leq \tau_{i+1}$ along with a state trajectory $(q(t), x(t))$ where $q(t)$ is constant and $x(t) \in X \backslash \mathcal{G}_{q(t)}$ for all $t \in [\tau_i, \tau_{i+1})$ if $\tau_i < \tau_{i+1}$, and $\dot{x}(t) \in f(q(t), x(t))$ for almost all $t \in [\tau_i, \tau_{i+1})$ if $\tau_i < \tau_{i+1}$. We allow the case where $N = \infty$ and the case where $N < \infty$, $\tau_N = \infty$. We denote the continuous state immediately prior to the ith transition by $x(\tau'_{i-1})$, *i.e.* $x(\tau'_{i-1}) \triangleq \lim_{t \to \tau_i^-} x(t)$ if $\tau_{i-1} < \tau_i$, or $x(\tau'_{i-1}) \triangleq x(\tau_i)$ if $\tau_{i-1} = \tau_i$. We further require $x(\tau'_i) \in \mathcal{G}_{q(\tau_i) \to q(\tau_{i+1})}$, and $x(\tau_{i+1}) \in R(q(\tau_i), q(\tau_{i+1}), x(\tau'_i))$ for $i = 1, \ldots, N-2$ and for $i = N-1$ if $q(\tau_N) \neq q(\tau_{N-1})$. If $N = \infty$ but $\sup_i \tau_i < \infty$, the execution is called *Zeno*. For a detailed discussion of the types of executions possible in hybrid systems, see [20].

2.3. Sum of Squares Programming

For a variable x taking values in $\mathbb{R}^n$, we denote by $\mathbb{R}[x]$ the set of all polynomials in x. Define

$$\Sigma[x] \triangleq \left\{ \sigma(x) \in \mathbb{R}[x] : \sigma(x) = \sum_{i=1}^{m} f_i(x)^2, f_i(x) \in \mathbb{R}[x] \right\}. \tag{4}$$

A polynomial $\sigma(x) \in \Sigma[x]$ is called a *sum of squares (SOS)* polynomial. Given $\{p_i(x)\}_{i=0}^m$ with $p_i \in \mathbb{R}[x]$, the problem of finding $\{q_i(x)\}_{i=1}^m$ with $q_i(x) \in \mathbb{R}[x]$ (or $q_i(x) \in \Sigma[x]$, or a mix of constraints for different i's) such that

$$p_0(x) + \sum_{i=1}^{m} q_i(x) p_i(x) \in \Sigma[x] \tag{5}$$

is a semidefinite program [17], and the MATLAB toolbox SOSTOOLS [21] transforms *SOS programs* of the form (5) into semidefinite programs.

3. Problem Formulation

Consider an *unsafe set* $U \subseteq Q \times X$ which includes undesirable regions of the state space. Given a hybrid system H, we call an execution of H *unsafe* if $(q(t), x(t)) \in U$ for some $t \in [0, \tau_N]$. We call H *safe* if there does not exist an unsafe execution of H.

Guard Synthesis Problem. *Given a hybrid system H with unspecified guards and an unsafe set $U \subseteq Q \times X$, synthesize a set of guards $\mathcal{G} = \{\mathcal{G}_{q \to q'}\}_{q,q' \in Q}$ such that H is safe.*

For $S \subset X$, we call $\Phi \subset X$ an *overapproximation of the reach set* from (q, S) if Φ contains all trajectories of the continuous dynamics in mode q that originate in S until a guard is encountered. Specifically, Φ is an overapproximation of the reach set from (q, S) if for all $T > 0$

$$\left.\begin{array}{ll} x(0) \in S & \\ \dot{x}(t) \in f(q, x(t)) & \text{for almost all } t \in [0, T) \\ x(t) \in (X \backslash \mathcal{G}_q) & \text{for all } t \in [0, T) \end{array}\right\} \text{ implies } \begin{array}{l} x(t) \in \Phi \ \forall t \in [0, T) \text{ and} \\ \lim_{t \to T^-} x(t) \in \Phi. \end{array} \tag{6}$$

As we only concern ourselves with overapproximations of reach sets in this work, we will often refer to such overapproximations as simply *reach sets*. We define the set-valued function $\textsc{Reach}(\cdot, \cdot)$ as follows:

$$\textsc{Reach}(q, S) \triangleq \{\Phi : \Phi \text{ is a reach set from } (q, S)\}. \tag{7}$$

Note that if $S \subset X$ is a positively invariant set for the dynamics $\dot{x} \in f(q,x)$, then $S \in \textsc{Reach}(q,S)$.

A number of techniques exist for obtaining such overapproximations. For example, in [6], the authors consider scalar-valued "barrier functions" $B_q(x)$ and use the fact that if

$$\nabla B_q(x)^T v \geq 0 \qquad \text{for all } v \in f(q,x), \text{for all } x \in (X\backslash\mathcal{G}_q) \text{ s.t. } B_q(x) = 0 \tag{8}$$

then $\{x : B_q(x) \geq 0\} \in \textsc{Reach}(q, \{x : B_q(x) \geq 0\})$. The authors of [6] propose a technique for constructing such barrier functions from a basis set of functions using an SOS program. This technique can be incorporated into our approach. We discuss this approach and others for calculating reach sets in the examples below, but otherwise do not concern ourselves with the computation of reach sets.

We now characterize a sufficient condition for safety using reach sets that serves as the foundation for our guard synthesis solution in Section 4.

Lemma 1. *Given unsafe $U \subseteq Q \times X$ and a hybrid system H, if there exists $\{S_q, \Phi_q\}_{q \in Q}$ with $S_q \subseteq X$ and $\Phi_q \subseteq X$ such that*

$$\begin{aligned}
\Phi_q &\in \textsc{Reach}(q, S_q) \; \forall q \in Q && \qquad (9)\\
I(q) &\subseteq S_q \qquad\qquad\;\; \forall q \in Q && \qquad (10)\\
R(q, q', \Phi_q \cap \mathcal{G}_{q \to q'}) &\subseteq S_{q'} \qquad\qquad\;\; \forall q, q' \in Q && \qquad (11)\\
\Phi_q \cap U(q) &= \emptyset \qquad\qquad\;\;\; \forall q \in Q && \qquad (12)
\end{aligned}$$

then H is safe.

The conditions of Lemma 1 are depicted schematically in Fig. 3.

Proof. Suppose not. Then there exists a time t^* and an execution such that $(q(t^*), x(t^*)) \in U$. It must be that $x(t^*) \notin \Phi_{q(t^*)}$ by (12). Let $i^* \triangleq \max\{i : \tau_i \leq t^*\}$. We have $x(\tau_{i^*}) \notin \Phi_{q(\tau_{i^*})}$ since $\Phi_{q(\tau_{i^*})} = \Phi_{q(t^*)}$ is a reach set for mode $q(\tau_{i^*})$. But $x(\tau_0) \in \Phi_{q(\tau_0)}$ by (9) and (10), thus $i^\dagger \triangleq \max\{i : x(\tau_i) \in \Phi_{q(\tau_i)}\}$ is well-defined. We have $x(\tau_{i^\dagger}) \in \Phi_{q(\tau_{i^\dagger})} \implies x(\tau'_{i^\dagger}) \in \Phi_{q(\tau_{i^\dagger})}$ by the definition of reach set and $x(\tau'_{i^\dagger}) \in \mathcal{G}_{q(\tau_{i^\dagger}) \to q(\tau_{i^\dagger+1})}$ by the definition of an execution. Also, $x(\tau_{i^\dagger+1}) \in R(q(\tau_{i^\dagger}), q(\tau_{i^\dagger+1}), x(\tau'_{i^\dagger}))$. Thus

$$x(\tau_{i^\dagger+1}) \in R(q(\tau_{i^\dagger}), q(\tau_{i^\dagger+1}), \Phi_{q(\tau_{i^\dagger})} \cap \mathcal{G}_{q(\tau_{i^\dagger}) \to q(\tau_{i^\dagger+1})}) \subseteq \Phi_{q(\tau_{i^\dagger+1})} \tag{13}$$

by (9), (11) and the property $\Phi \in \textsc{Reach}(q,S) \implies S \subset \Phi$. But this contradicts the definition of $i^\dagger$. □

Note that while Lemma 1 ensures H is safe, it does not establish the nonexistence of Zeno executions. Below is a sufficient condition for ruling out this phenomenon.

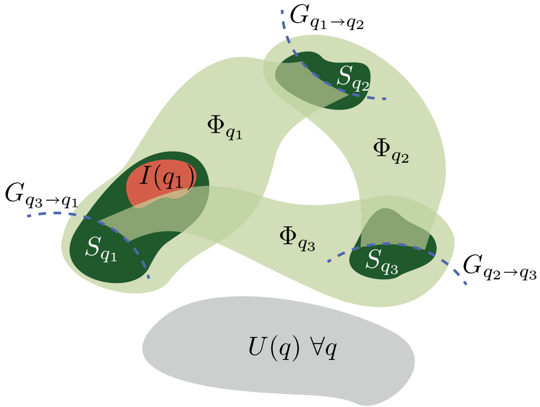


Figure 1: A schematic depiction of the conditions from Lemma 1 for a hybrid system that cycles through modes $q_1 \to q_2 \to q_3 \to q_1 \to \cdots$ where we assume each reset map is the identity map. In the figure, Φ_q is a reach set from (q, S_q) for $q \in \{q_1, q_2, q_3\}$ and satisfies (12). The sets $\{S_q\}$ satisfy (10)–(11). For simplicity, only the boundary of each guard is depicted.

Proposition 1. *If conditions* (9)–(11) *of Lemma 1 hold,* $\mathbf{cl}(\cup_q \Phi_q)$ *is compact, and*

$$\mathbf{cl}(R(q, q', \Phi_q \cap \mathcal{G}_{q \to q'})) \cap \mathbf{cl}(\mathcal{G}_{q'}) = \emptyset \quad \forall q, q' \in Q \tag{14}$$

then no executions of H are Zeno.

Proof. When $\mathbf{cl}(\cup_q \Phi_q)$ is compact, we have $\mathbf{cl}(R(q, q', \Phi_q \cap \mathcal{G}_{q \to q'})) \subset \Phi_{q'}$ compact, thus the distance between $\mathbf{cl}(R(q, q', \Phi_q \cap \mathcal{G}_{q \to q'}))$ and $\mathbf{cl}(\mathcal{G}_{q'})$ is strictly greater than 0. Since this holds for all $q, q' \in Q$ and since $f(q, \cdot)$ is locally bounded for all $q \in Q$ by assumption, there is a minimum dwell time and thus Zeno executions are prevented. □

We will primarily be interested in systems which do not allow trajectories to become unbounded, thus the assumption that $\mathbf{cl}(\cup_q \Phi_q)$ is compact is reasonable. It is possible to extend Proposition 1 to the case when $\mathbf{cl}(\cup_q \Phi_q)$ is not compact by making additional assumptions on the reset map. Condition (14) can also be relaxed by considering all cycles of the hybrid automaton instead of all transitions, see [1]. It is not difficult to adjust the algorithm presented below (specifically, (20)) for this more general case.

4. Guard Synthesis Algorithm

Consider the hybrid system H and unsafe set $U \subset Q \times X$. Let $U(q) \triangleq \{x : (q, x) \in U\}$. We assume $I(q)$ can be over-approximated with a semialgebraic set, that is, $I(q) \subseteq \{x \in X : \lambda_{I(q)}(x) \succeq 0\}$ where $\lambda_{I(q)}(\cdot)$ is a vector-valued, polynomial function. We also assume each $\{\mathcal{R}_{q \to q'}\}$ and $U(q)$ can similarly be over approximated with vector-valued polynomial functions $\lambda_{\mathcal{R}_{q \to q'}}(x)$ and $\lambda_{U(q)}(x)$.

The output dimensions of each function need not be the same. Similarly, we assume the reset map $R(q, q', \cdot)$ satisfies

$$R(q, q', x) \subseteq \{\xi : \lambda_{R_{q \to q'}}(x, \xi) \succeq 0\} \tag{15}$$

where $\lambda_{R_{q \to q'}}(x, \xi)$ is a vector-valued polynomial function in the variables x and ξ for each $q, q' \in Q$.

We now present a theorem which forms the basis of our guard synthesis algorithm by converting conditions (9)–(14) into SOS constraints where feasibility is sufficient for each condition. In particular, (16)–(18) below correspond to (10)–(12), and (20) corresponds to (14). Equation (19) ensures that guard transitions are only taken when in the domain of the reset map. These equations can be interpreted as special cases of the *Positivstellensatz* condition [22].

Theorem 1. *Given a hybrid system H and a set of bounded reach sets $\{\Phi_q\}$ and sets $\{S_q\}$ with $\Phi_q \in \textsc{Reach}(q, S_q)$ described by $\Phi_q \triangleq \{x : \phi_q(x) \succeq 0\}$ and $S_q(x) \triangleq \{x : s_q(x) \succeq 0\}$ with $\phi_q(x), s_q(x)$ vector-valued, polynomial functions.*

Consider a set $\{g_{q \to q'}(\cdot)\}$ of vector-valued, polynomial functions defining a set of guards $\mathcal{G} \triangleq \{\mathcal{G}_{q \to q'}\}_{q,q' \in Q}$ by $\mathcal{G}_{q \to q'} \triangleq \{x : \phi_q(x) \succeq 0$ and $g_{q \to q'}(x) \succeq 0\}$.

If there exists a set $\{\sigma_{k,*}(x)\}_{k=1}^{12}$ where each $\sigma_{k,*}(x)$ is a vector of SOS polynomials with $*$ replaced by elements from an appropriate index set such that:

1. *(Reach sets contain initial condition:)*

$$s_q^{(i)}(x) - \sigma_{1,q,i}(x)^T \lambda_{I(q)}(x) \in \Sigma[x] \tag{16}$$

for all $i = 1, \dots, Dim(s_q(x))$ for all $q \in Q$,

2. *(When encountering a guard, reach sets transition into reach sets via the reset map:)*

$$\begin{aligned} s_{q'}^{(i)}(\xi) - \sigma_{2,q \to q',i}(x,\xi)^T \phi_q(x) - \sigma_{3,q \to q',i}(x,\xi)^T g_{q \to q'}(x) \\ -\sigma_{4,q \to q',i}(x,\xi)^T \lambda_{R_{q \to q'}}(x,\xi) \in \Sigma[x,\xi] \end{aligned} \tag{17}$$

for all $i = 1, \dots, Dim(s_{q'}(x))$ for all $q, q' \in Q$,

3. *(Reach sets do not intersect the unsafe set:)*

$$-(1 + \sigma_{5,q}(x)^T \lambda_{U(q)}(x) + \sigma_{6,q}(x)^T \phi_q(x)) \in \Sigma[x] \tag{18}$$

for all $q \in Q$,

4. *(Transitions only occur within the domain of the reset maps:)*

$$\lambda_{\mathcal{R}_{q \to q'}}^{(i)}(x) - \sigma_{7,q \to q',i}(x)^T \phi_q(x) - \sigma_{8,q \to q',i}(x)^T g_{q \to q'}(x) \in \Sigma[x] \tag{19}$$

for all $i = 1, \dots, Dim(\lambda_{\mathcal{R}_{q \to q'}}(x))$ and for all $q, q' \in Q$

then H is safe.

Furthermore, if

$$\begin{aligned}&-(1+\sigma_{9,q'\to q''}(x,\xi)^T g_{q'\to q''}(\xi)+\sigma_{10,q}(x,\xi)^T\phi_q(x)\\ &+\sigma_{11,q\to q'}(x,\xi)^T g_{q\to q'}(x)+\sigma_{12,q\to q'}(x,\xi)^T\lambda_{R_{q\to q'}}(x,\xi))\in\Sigma[x,\xi]\end{aligned} \tag{20}$$

for all $q, q', q'' \in Q$ then no execution of H is Zeno.

Proof. We will show that (16)–(20) imply (2), (10)–(12), and (14), and then we apply Lemma 1 and Proposition 1.

- (19) $\implies$ (2). We have (19) implies

  $$\begin{bmatrix}\phi_q(x)\\ g_{q\to q'}(x)\end{bmatrix}\succeq 0 \implies \lambda^{(i)}_{\mathcal{R}_{q\to q'}}(x)\succeq 0. \tag{21}$$

  Indeed, suppose not for a particular x'. Then

  $$\lambda^{(i)}_{\mathcal{R}_{q\to q'}}(x')-\sigma_{7,q\to q',i}(x')^T\phi_q(x')-\sigma_{8,q\to q',i}(x')^T g_{q\to q'}(x')<0 \tag{22}$$

  since $\sigma_{7,*}(x')\geq 0$ and $\sigma_{8,*}(x')\geq 0$, contradicting (19). Since this holds for all $i=1,\ldots,\mathrm{Dim}(\lambda_{\mathcal{R}_{q\to q'}}(x))$, we have $\mathcal{G}_{q\to q'}=\{x:\begin{bmatrix}\phi_q^T(x) & g_{q\to q'}^T(x)\end{bmatrix}^T\succeq 0\}\subseteq\mathcal{R}_{q\to q'}$ and therefore $\mathcal{G}$ is a valid guard set.

- (16) $\implies$ (10). Applying reasoning similar to (21), we conclude from (16) that $\lambda_{I(q)}(x)\succeq 0 \implies s_q^{(i)}(x)\succeq 0$ for all $q\in Q$ and for all $i=1,\ldots,\mathrm{Dim}(s_q)$. This implies $S_q\supseteq I(q)$ for all $q\in Q$.

- (17) $\implies$ (11). Similarly, we conclude from (17)

  $$\begin{bmatrix}\phi_q(x)\\ g_{q\to q'}(x)\\ \lambda_{R_{q\to q'}}(x,\xi)\end{bmatrix}\succeq 0 \implies s_{q'}^{(i)}(\xi)\succeq 0 \tag{23}$$

  for all i and all $q,q'\in Q$. Equivalently,

  $$x\in\Phi_q\cap\mathcal{G}_{q\to q'} \text{ and } \xi\in R(q,q',x)\implies \xi\in S_{q'}, \tag{24}$$

  thus $R(q,q',\Phi_q\cup\mathcal{G}_{q\to q'})\subseteq S_{q'}$.

- (18) $\implies$ (12). We have (18) implies

  $$\left\{x:\begin{matrix}\lambda_{U(q)}(x)\succeq 0\\ \phi_q(x)\succeq 0\end{matrix}\right\} \text{ is empty.} \tag{25}$$

  Indeed, suppose not and let $\lambda_{U(q)}(x')\succeq 0$ and $\phi_q(x')\succeq 0$. Then $-(1+\sigma_{5,q}(x')^T\lambda_{U(q)}(x')+\sigma_{6,q}(x')^T\phi_q(x'))<0$, a contradiction.

  Applying Lemma 1, we have that H is safe.

- (20) $\implies$ (14). Finally, (20) implies

$$\left\{ x : \begin{bmatrix} \phi_q(x) \\ g_{q\to q'}(x) \\ g_{q'\to q''}(\xi) \\ \lambda_{R_{q\to q'}}(x,\xi) \end{bmatrix} \succeq 0 \right\} \text{ is empty } \forall q, q', q'' \in Q \tag{26}$$

which gives (14), thus preventing Zeno executions.

□

A number of remarks are in order:

Remark 1. It is sometimes more convenient or necessary to represent $U(q)$ as

$$U(q) = \{x : (\lambda_{1,U(q)}(x) \succeq 0) \vee \ldots \vee (\lambda_{J,U(q)}(x) \succeq 0)\}. \tag{27}$$

For such $U(q)$, we can simply verify (18) for each $\lambda_{j,U(q)}(x)$, $j = 1, \ldots, J$.

Remark 2. If a convenient vector-valued polynomial inequality description exists for the safe set (i.e., $\mathrm{Safe}(q) = (q, X \backslash U(q)) = \{x : \lambda_{\mathrm{Safe}(q)}(x) \succeq 0\}$), we can replace (18) with a constraint of the form:

$$\lambda^{(i)}_{\mathrm{Safe}(q)}(x) - \sigma^T_{5,q,i}(x)\phi_q(x) \in \Sigma[x] \quad \forall q \in Q, \forall i = 1, \ldots, \mathrm{Dim}(\lambda_{\mathrm{Safe}(q)}(x)). \tag{28}$$

Remark 3. If Φ_q is an invariant set for the dynamics in mode q, then we can let $s_q(x) = \phi_q(x)$.

Remark 4. If $R(q, q', \cdot)$ is a polynomial function rather than a set-valued map as in (15), we can use the following two equations in place of (17) and (20), respectively:

$$s^{(i)}_{q'}(R(q,q',x)) - \sigma_{2,q\to q',i}(x)^T \phi_q(x) - \sigma_{3,q\to q',i}(x)^T g_{q\to q'}(x) \in \Sigma[x] \tag{29}$$

$$\begin{aligned} -(1 + \sigma_{9,q'\to q''}(x)^T g_{q'\to q''}(R(q,q',x)) + \sigma_{10,q}(x)^T \phi_q(x) \\ + \sigma_{11,q\to q'}(x)^T g_{q\to q'}(x)) \in \Sigma[x]. \end{aligned} \tag{30}$$

We use Theorem 1 as a guide for synthesizing guards. In particular, we fix the degrees of the SOS variables and the guards. We also introduce an iterative procedure solving a convex problem at each stage as is commonly done when solving bilinear SOS problems related to control, see *e.g.* [15], [14], and [6], and has been found to be effective despite lack of convergence guarantees. Our proposed iterative procedure seeks a feasible solution to (16)–(20) and alternates between the following two steps:

GS-1) Fix $\phi_q(x)$, $s_q(x)$ and $g_{q\to q'}(x)$ and solve for the SOS variables $\{\sigma_{k,*}(x)\}_{k=1}^{12}$.

GS-2) Fix all SOS variables except $\sigma_{1,q}(x)$, $\sigma_{5,q}(x)$ for all q. Solve for $\phi_q(x)$, $s_q(x)$, $g_{q\to q'}(x)$, $\sigma_{1,q}(x)$, and $\sigma_{5,q}(x)$ for all q, q'.

We initialize the iteration by relaxing the synthesis requirements, *e.g.* considering a smaller unsafe region or smaller set of initial conditions, and initializing with functions $\{\phi_q(x)\}_{q\in Q}$, $\{s_q(x)\}_{q\in Q}$ and $\{g_{q\rightarrow q'}(x)\}_{q,q'\in Q}$ known to satisfy this simpler problem. The algorithm then iteratively solves GS-1) and GS-2), adjusting the problem constraints until a feasible solution is found for the original problem.

4.1. Discussion

Our synthesis algorithm is similar in spirit to the nonconvex, worst-case safety verification procedure proposed in [6], however there are some key differences. Our proposed algorithm is a method for synthesizing safe control strategies, while [6] seeks to verify that a given strategy is safe. In addition, [6] verifies scalar-valued barrier functions by checking the flow of the vector field along the boundary of the barrier, specifically condition (8). We do not specify how the reach sets are obtained, and checking the vector field flow along the barrier is a possible method. However, in principle, reach sets can be obtained using other methods and we allow for vector-valued reach sets as in Section 5.

We remark that the above synthesis procedure only provides a sufficient condition for verifying safety, however our control synthesis approach offers several advantages over existing synthesis techniques. For example, our approach requires over-approximation of (forward) reach sets rather than computation of reach-avoid sets. Over-approximating reach sets is often more tractable and can be easily incorporated in an SOS framework using barrier functions as described above. Our approach does not require discretization of space or time and thus does not suffer from the numerical approximation issues of the HJ approaches as described in the Introduction. In particular, our approach is guaranteed to be sound and "exact" guards in the form of polynomial functions are obtained. Finally, while the improvements in computational time compared to HJ solutions is highly problem dependent, steady progress in convex optimization and in SOS program solutions [23, 24] suggests that our approach will be applicable to ever larger systems.

4.2. Example

Consider a hybrid system with three modes $Q = \{\mathtt{A}, \mathtt{B}, \mathtt{C}\}$ where each mode is an affine system governed by $\dot{x} = A_q(x - \bar{x}_q) + w$ for $q \in Q$ where $x \in \mathbb{R}^2$, $w \in \mathcal{W} \triangleq \{w : w^T w \leq 1\} \subset \mathbb{R}^2$, and

$$A_{\mathtt{A}} = \begin{bmatrix} -4 & 2 \\ 1 & -1 \end{bmatrix} \qquad A_{\mathtt{B}} = \begin{bmatrix} -2 & -2 \\ -1 & -2 \end{bmatrix} \qquad A_{\mathtt{C}} = \begin{bmatrix} -2 & 0 \\ 0 & -1 \end{bmatrix} \tag{31}$$

$$\bar{x}_{\mathtt{A}} = \begin{bmatrix} -1 & -1.5 \end{bmatrix}^T \qquad \bar{x}_{\mathtt{B}} = \begin{bmatrix} 0 & -.5 \end{bmatrix}^T \qquad \bar{x}_{\mathtt{C}} = \begin{bmatrix} 0 & 0.5 \end{bmatrix}^T. \tag{32}$$

Assume $\mathcal{R}_{\mathtt{A}\rightarrow\mathtt{B}} = \mathcal{R}_{\mathtt{B}\rightarrow\mathtt{C}} = \mathcal{R}_{\mathtt{C}\rightarrow\mathtt{A}} = \mathbb{R}^2$ and $\mathcal{R}_{\mathtt{A}\rightarrow\mathtt{C}} = \mathcal{R}_{\mathtt{B}\rightarrow\mathtt{A}} = \mathcal{R}_{\mathtt{C}\rightarrow\mathtt{A}} = \emptyset$. Fig. 2 depicts the resulting hybrid automaton. Let the unsafe set be $U(q) = \{x : x^T x \geq 4\}$ for all $q \in Q$ and assume the system is initialized near the origin with $I(q) = \{x : x^T x \leq 0.0001\}$ for all $q \in Q$.

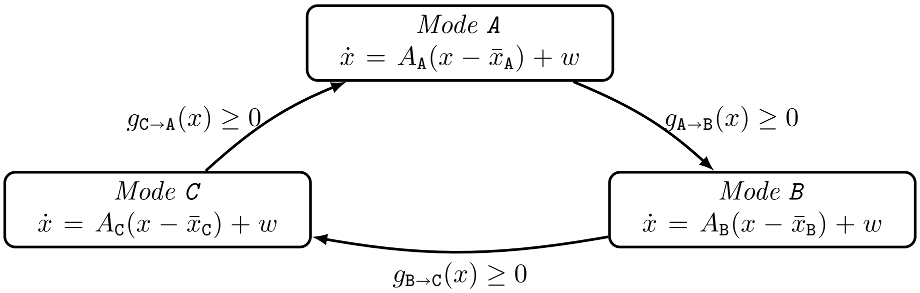

Mode A
$\dot{x} = A_{\mathtt{A}}(x - \bar{x}_{\mathtt{A}}) + w$
$g_{\mathtt{C}\to\mathtt{A}}(x) \geq 0$
$g_{\mathtt{A}\to\mathtt{B}}(x) \geq 0$
Mode C
$\dot{x} = A_{\mathtt{C}}(x - \bar{x}_{\mathtt{C}}) + w$
Mode B
$\dot{x} = A_{\mathtt{B}}(x - \bar{x}_{\mathtt{B}}) + w$
$g_{\mathtt{B}\to\mathtt{C}}(x) \geq 0$

Figure 2: Transition diagram for the example in Section 4.2. Transitions are taken when guard functions become nonnegative.

We seek to design guard functions $g_{\mathtt{A}\to\mathtt{B}}(x)$, $g_{\mathtt{B}\to\mathtt{C}}(x)$, and $g_{\mathtt{C}\to\mathtt{A}}(x)$ to ensure that the system does not enter the unsafe region. Since the dynamics of each mode are affine, we search for positive definite matrices $P_q \in \mathbb{R}^{2\times 2}$ such that the ellipsoid $\mathcal{E}_q \triangleq \{x \mid (x-\bar{x}_q)^T P_q (x - \bar{x}_q) \leq 1\}$ is a robustly invariant set [25]. We then define

$$s_{\mathtt{A}}(x) = \phi_{\mathtt{A}}(x) \triangleq \begin{bmatrix} 1 - (x-\bar{x}_{\mathtt{A}})^T P_{\mathtt{A}}(x-\bar{x}_{\mathtt{A}}) & -g_{\mathtt{A}\to\mathtt{B}}(x)\end{bmatrix}^T \tag{33}$$

$$s_{\mathtt{B}}(x) = \phi_{\mathtt{B}}(x) \triangleq \begin{bmatrix} 1 - (x-\bar{x}_{\mathtt{B}})^T P_{\mathtt{B}}(x-\bar{x}_{\mathtt{B}}) & -g_{\mathtt{B}\to\mathtt{C}}(x)\end{bmatrix}^T \tag{34}$$

$$s_{\mathtt{C}}(x) = \phi_{\mathtt{C}}(x) \triangleq \begin{bmatrix} 1 - (x-\bar{x}_{\mathtt{C}})^T P_{\mathtt{C}}(x-\bar{x}_{\mathtt{C}}) & -g_{\mathtt{C}\to\mathtt{A}}(x)\end{bmatrix}^T \tag{35}$$

so that $\Phi_q \in \textsc{Reach}(q, S_q)$ where Φ_q and S_q are defined as in Theorem 1. In (33), we include $-g_{\mathtt{A}\to\mathtt{B}}(x)$ since the continuous state can only evolve in a region where the guard from `A` to `B` is not active, and similarly for (34) and (35). We restrict to guards that are half-planes and obtain

$$P_{\mathtt{A}} = \begin{bmatrix} 4.721 & -2.574 \\ -2.574 & 1.529 \end{bmatrix} \quad P_{\mathtt{B}} = \begin{bmatrix} 1.907 & 2.264 \\ 2.264 & 3.340 \end{bmatrix} \quad P_{\mathtt{C}} = \begin{bmatrix} 3.457 & 0.198 \\ 0.198 & 0.338 \end{bmatrix} \tag{36}$$

and

$$g_{\mathtt{A}\to\mathtt{B}} = \begin{bmatrix} -1.529 & -2.214 \end{bmatrix}^T x - 1.755 \tag{37}$$

$$g_{\mathtt{B}\to\mathtt{C}} = \begin{bmatrix} 3.055 & -1.318 \end{bmatrix}^T x - 1.561 \tag{38}$$

$$g_{\mathtt{C}\to\mathtt{A}} = \begin{bmatrix} -5.305 & 4.024 \end{bmatrix}^T x - 2.813. \tag{39}$$

We thus certify safety for any initial condition within $\bigcup_{q\in Q}(q, \Phi_q)$ shown in Fig. 3. Note that none of the ellipsoids $\mathcal{E}_q$ in Fig. 3 are completely contained within the safe region, thus mode switching is required to achieve safety.

The above example is easily extended to general switched affine systems with an arbitrary number of modes.

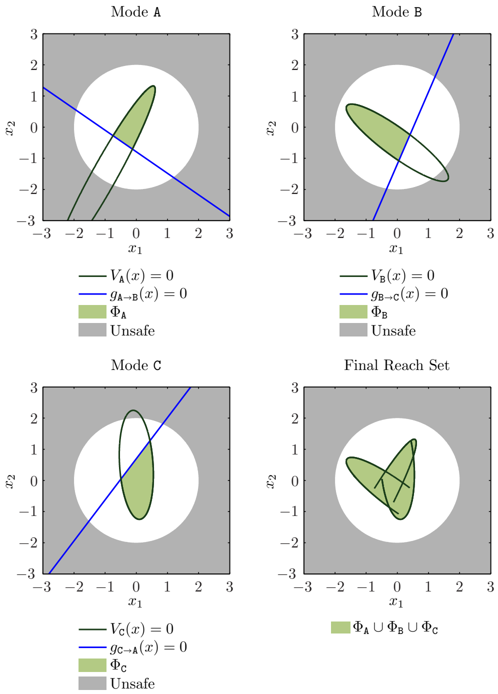


Figure 3: Positively invariant sets, reach sets, and guards for Example A generated by the guard synthesis algorithm. In the figure, $V_q(x) \triangleq 1 - (x - \bar{x}_q)^T P_q (x - \bar{x}_q)$.

5. Application to Highway Onramp Metering

5.1. Model Definition

We now present an application of our technique to the cell transmission model of freeway traffic flow [26]. We consider a freeway model consisting of two segments, see Fig. 4 (adapted from Example 1, [27]). The number of vehicles in segment 0 (resp. segment 1) at time t is $n_0(t)$ (resp. $n_1(t)$). An onramp with queue length $n_r(t)$ merges with segment 0. Let $n(t) = \begin{bmatrix} n_0(t) & n_1(t) & n_r(t) \end{bmatrix}^T$.

Vehicles flow into segment 1 at a constant rate of I_1 vehicles per hour (vph), and vehicles join the onramp queue at a constant rate of I_r vph. In addition, $f_1(n)$ vph flow from segment 1 to segment 0, $f_0(n)$ vph flow out of segment 0, and vehicles also exit segment 1 via an offramp at a rate $\beta f_1(t)$ vph for some $\beta \in (0,1)$ where $f_0(\cdot)$ and $f_1(\cdot)$ are defined subsequently.

The maximum flow rate from the onramp to segment 0 can be controlled via *metering* of traffic using, *e.g.*, traffic lights, and thus the discrete state space is $Q \triangleq \{\mathtt{FF}, \mathtt{M}\}$ where

- Mode FF is the *ramp free-flow* mode and
- Mode M is the *ramp metered* mode.

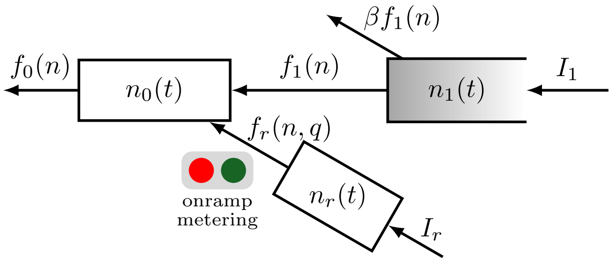


Figure 4: Cell transmission model of a freeway consisting of two segments and a merging onramp.

The resulting mode-dependent flow from the onramp is $f_r(n, q)$, defined subsequently.

The cell transmission model results in the following dynamics:

$$\dot{n}_0 = -f_0(n) + f_1(n) + f_r(n) \tag{40}$$

$$\dot{n}_1 = -(1 + \beta)f_1(n) + I_1 \tag{41}$$

$$\dot{n}_r = -f_r(n, q) + I_r. \tag{42}$$

The flow rates are defined by the following:

$$f_0(n) \triangleq \min\{vn_0(t), F_0\} \tag{43}$$

$$f_1(n) \triangleq \min\{vn_1(t), F_1, -w(n_0(t) - N_c)\} \tag{44}$$

$$f_r(n, q) \triangleq \min\{vn_r(t), F_r(q)\} \tag{45}$$

where v is the free-flow speed of vehicles in miles per hour, N_c is the maximum number of vehicles that can occupy segment 0, w is the *congestion wave speed* (see [26]), F_i for $i \in \{0, 1\}$ represent maximum flow rates, and $F_r(q)$ is the onramp maximum flow rate in mode q. Equations (43)–(45) follow from the fundamental diagram of traffic flow, see [28] and references therein.

The discrete mode $q \in \{\mathtt{FF}, \mathtt{M}\}$ affects the maximum flow rate from the onramp, *i.e*

- $F_r(\mathtt{FF})$ is the maximum *free-flow* rate from the onramp, and
- $F_r(\mathtt{M})$ is the maximum *metered* rate obtained by metering traffic flow from the onramp.

For the numerical results presented, we use the values in Table 1, for which n_1 is nondecreasing[1]. We assume $n_1(0) \geq F_1/v$ and therefore

$$n_1(t) \geq F_1/v \quad \forall t \geq 0 \tag{46}$$

[1] This situation may arise, for instance, during “rush hour” when total freeway input exceeds capacity.

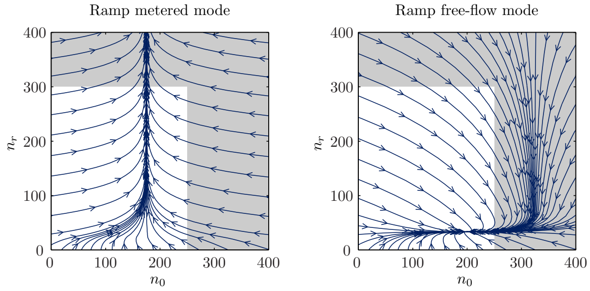


Figure 5: Trajectories of (40)–(42) in the (n_0, n_r)-plane using values in Table 1 for the case when $n_1(t) \geq F_1/v$ $\forall t \geq 0$. The shaded region is unsafe.

$F_0 = 6000$	$v = 60$	$I_1 = 9000$
$F_1 = 6000$	$w = 20$	$I_r = 2000$
$F_r(\texttt{FF}) = 4500$	$\beta = 1/2$	$N_r = 300$
$F_r(\texttt{M}) = 1500$	$N_c = 400$	$\Theta = 1500$
$\delta = 8$	$\alpha_r = 3/4$	

Table 1: Parameter values for Example B

for which

$$(n_0^e, n_r^e) \triangleq (-(F_0 - I_r)/w + N_c, I_r/v) \tag{47}$$

$$= (200, 100/3) \text{ (for values in Table 1)} \tag{48}$$

is an equilibrium occupancy for segment 0 and the onramp in the ramp free-flow mode. Furthermore, $f_1(n)$ reduces to $\min\{F_1, -w(n_0(t) - N_c)\}$ and then $(\dot{n}_0, \dot{n}_r)$ is a function of the metering mode and (n_0, n_r) only. Fig. 5 shows trajectories projected in the (n_0, n_r)-plane assuming (46). We seek to design a strategy for switching between ramp free-flow mode and ramp metered mode to achieve a safety and performance condition, defined in the following subsection.

5.2. Safety Condition

We impose two constraints characterizing desirable characteristics of the freeway model derived above. We first assume that the onramp has finite capacity N_r, and thus we wish to ensure

$$n_r(t) \leq N_r \quad \forall t \geq 0. \tag{49}$$

The practical implications of exceeding this maximum capacity may be an undesirable increase in congestion on streets near the onramp entrance. We also wish to maintain a minimum freeway exit throughput $\beta f_1(t) \geq \Theta$. Assuming (46), this results in an upper bound on $n_0(t)$, thus we have the second safety condition

$$n_0(t) \leq N_0 \quad \forall t \geq 0 \tag{50}$$

where $N_0 = -\Theta/(\beta w) + N_c$. We refer to (49) and (50) as the safety constraints.

Numerical values for N_r and Θ are given in Table 1, which gives $N_0 = 250$. Thus we see that the equilibrium at (n_0^e, n_r^e) in ramp free-flow mode satisfies our safety constraint, however not all trajectories initialized in ramp free-flow mode remain safe before reaching the equilibrium.

5.3. Guard Synthesis

We seek a strategy for switching between ramp free-flow and ramp metered mode such that trajectories converge to the equilibrium (n_0^e, n_r^e) while remaining within the safe region, and do so with a "reasonable" number of switches. We make this precise in the following problem statement:

Guard Synthesis Statement. Design guard sets $\mathcal{G}_{\texttt{FF}\to\texttt{M}}$ and $\mathcal{G}_{M\to\texttt{FF}}$ for transitioning from ramp free-flow mode to ramp metered mode and from ramp metered mode to ramp free-flow mode, respectively, such that

C1) the safety conditions (49) and (50) are satisfied for all trajectories initialized in the free-flow mode with $n_r(0) \leq \alpha_r N_r$ and $n_0(0) \leq N_0$ where $\alpha_r \in [0, 1)$ (we use the value in Table 1),

C2) trajectories converge to the equilibrium (n_0^e, n_r^e) in ramp free-flow mode, and

C3) the ramp queue length decreases by at least δ vehicles (given in Table 1) between subsequent switches to ramp metering mode. This imposes an upper bound on the number of switches required to safely reach the equilibrium point.

Fig. 6(a) demonstrates the naive strategy of switching immediately before reaching the unsafe set, *i.e.* $\mathcal{G}_{\texttt{FF}\to\texttt{M}} = \{(n_0, n_r) : n_0 \geq N_0\}$ and $\mathcal{G}_{\texttt{M}\to\texttt{FF}} = \{(n_0, n_r) : n_r \geq N_r\}$, which results in trajectories that do not converge to the equilibrium, motivating the need for the guard synthesis algorithm.

As the safety condition is a function of n_0 and n_r only, we define $\bar{n}(t) \triangleq (n_0(t), n_r(t))$ to be the state of the freeway system and let

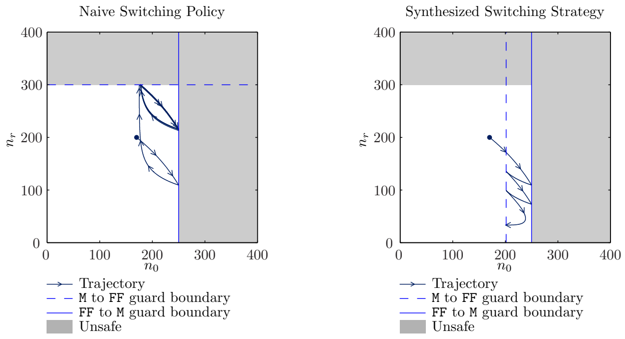
$$S(\texttt{FF}) = S(\texttt{M}) \triangleq \{\bar{n} : \gamma_S(\bar{n}) \succeq 0\}, \qquad \gamma_S(\bar{n}) \triangleq \begin{bmatrix} N_0 - n_0 & N_r - n_r \end{bmatrix}^T \tag{51}$$

be the safe set. We define guards as follows:

$$\mathcal{G}_{\texttt{FF}\to\texttt{M}} \triangleq \{\bar{n} : g_{\texttt{FF}\to\texttt{M}}(\bar{n}) \geq 0\} \tag{52}$$

$$\mathcal{G}_{\texttt{M}\to\texttt{FF}} \triangleq \{\bar{n} : g_{\texttt{M}\to\texttt{FF}}(\bar{n}) \geq 0\} \tag{53}$$

for polynomial functions $g_{\texttt{FF}\to\texttt{M}}(\cdot)$ and $g_{\texttt{M}\to\texttt{FF}}(\cdot)$. We choose $g_{\texttt{M}\to\texttt{FF}}(\bar{n}) = c_0 - n_0$ where c_0 is a design parameter obtained using the SOS guard synthesis



(a) (b)

Figure 6: (a) A naive strategy of switching modes immediately before reaching the unsafe set, initialized from the point $(n_0, n_r) = (170, 200)$. The trajectory does not converge to the equilibrium $(200, 100/3)$. (b) A switching strategy synthesized using the guard synthesis algorithm. The trajectory converges to the equilibrium $(200, 100/3)$ with only four mode switches.

algorithm, and we fix $g_{\texttt{FF}\to\texttt{M}}(\bar{n}) \triangleq n_0 - N_0$. We will see that this formulation is sufficient to obtain guards that guarantee safety, but if it were not, we could allow for higher order guards in the synthesis algorithm.

Let $n_e \triangleq (n_0^e, n_r^e)$ denote the equilibrium in the free-flow mode. To ensure C3), we establish reach sets $\phi_{\texttt{FF}}(\bar{n}, l)$ and $\phi_{\texttt{M}}(\bar{n}, l)$ with

$$\phi_q(\bar{n}, l) \triangleq \begin{bmatrix} V_q^1(\bar{n}) - l & V_q^2(\bar{n}) & -g_{q\to q'}(\bar{n}) & n_r - c_r \end{bmatrix}^T \tag{54}$$

for $(q, q') \in \{(\texttt{FF}, \texttt{M}), (\texttt{M}, \texttt{FF})\}$ where the purpose of l is described below. We limit our scope to the case where $\{n_r \geq c_r\}$ for some $c_r < n_r^e$ (we choose $c_r = 25$) to facilitate the SOS procedure as we can see that $\{\bar{n} : n_r \geq c_r\}$ is invariant and all trajectories with $n_r(0) \leq n_r^e$ and $n_0(0) \leq N_0$ converge to the equilibrium along safe trajectories. Furthermore, we choose $V_q^2(\bar{n}) \triangleq -g_{q'\to q}(\bar{n})$.

We can in fact establish a family of reach sets by encoding the following expression into the SOS guard synthesis algorithm:

$$\left.\begin{aligned} n_r &\geq c_r \\ -g_{q'\to q}(\bar{n}) &\geq 0 \\ -g_{q\to q'}(\bar{n}) &\geq 0 \end{aligned}\right\} \implies \mathbf{f}(\bar{n}, q)^T \nabla V_q^1(\bar{n}) \leq 0 \tag{55}$$

where

$$\mathbf{f}(\bar{n}, q) \triangleq \begin{bmatrix} -f_0(n) + f_1(n) + f_r(n) \\ -f_r(n, q) + I_r \end{bmatrix}. \tag{56}$$

The assumption (46) ensures $\mathbf{f}(\cdot, q)$ is only a function of $\bar{n}$. The advantage of (55)–(56) is that we have guaranteed

$$\{\bar{n} : \phi_q(\bar{n}, l) \succeq 0\} \in \textsc{Reach}(q, \{\bar{n} : \phi_q(\bar{n}, l) \succeq 0\}) \quad \text{for each } l. \tag{57}$$

We choose to parameterize $V_q^1(\bar{n})$ as piecewise linear, and Fig. 7 shows $V_{\texttt{FF}}^1(\bar{n}) - l = 0$ and $V_{\texttt{M}}^1(\bar{n}) - l = 0$ designed from the final guard synthesis algorithm for several values of l. To achieve C3), we impose additional constraints on $\phi_{\texttt{FF}}$ and $\phi_{\texttt{M}}$ for a specific value of l. In particular, we impose

$$\left.\begin{aligned} \phi_{\texttt{FF}}(\bar{n},0) \succeq 0 \\ g_{\texttt{FF}\to\texttt{M}}(\bar{n}) \succeq 0 \end{aligned}\right\} \implies V_{\texttt{FF}}^1(\bar{n}) \leq 0 \tag{58}$$

$$\left.\begin{aligned} \phi_{\texttt{M}}(\bar{n},0) \succeq 0 \\ g_{\texttt{M}\to\texttt{FF}}(\bar{n}) \succeq 0 \end{aligned}\right\} \implies V_{\texttt{M}}^1(n_0, n_r + \delta) \leq 0. \tag{59}$$

Equations (58) and (59) ensure that each time the freeway switches from metered to free-flow and back to metered mode, the ramp queue decreases by at least δ vehicles. We use $l = 0$ in (58)–(59), but any choice would work.

To ensure C2), we search for a quadratic function $\phi_e(n) = (\bar{n}-n_e)^T P_e(\bar{n}-n_e)$ where $P_e \in \mathbb{R}^{2\times 2}$ is a design parameter with the restriction $P_e \succeq 0$ such that $\Phi_e \triangleq \{\bar{n} : \phi_e(\bar{n}) \leq 1\} \subset S(\texttt{FF})$ and Φ_e is invariant. We then ensure $\{\bar{n} : \phi_{\texttt{M}}(\bar{n},0) \succeq 0\} \subset V_e(\bar{n})$, thereby implying that for trajectories within Φ_e at time t, the next switch to free-flow mode will converge to the equilibrium with no further switching. Finally, we must ensure that trajectories initialized with large queue length n_r remain in the safe region so that C1) holds. To this end, we verify $\{\bar{n} : n_r \leq \alpha_r N_r\}$ is invariant in the free-flow mode and ensure that there exists l^* such that

$$\left.\begin{aligned} n_r \leq \alpha_r N_r \\ n_0 \leq N_0 \\ g_{\texttt{FF}\to\texttt{M}}(\bar{n}) \geq 0 \end{aligned}\right\} \implies \phi_{\texttt{M}}(\bar{n}, l^*) \succeq 0 \tag{60}$$

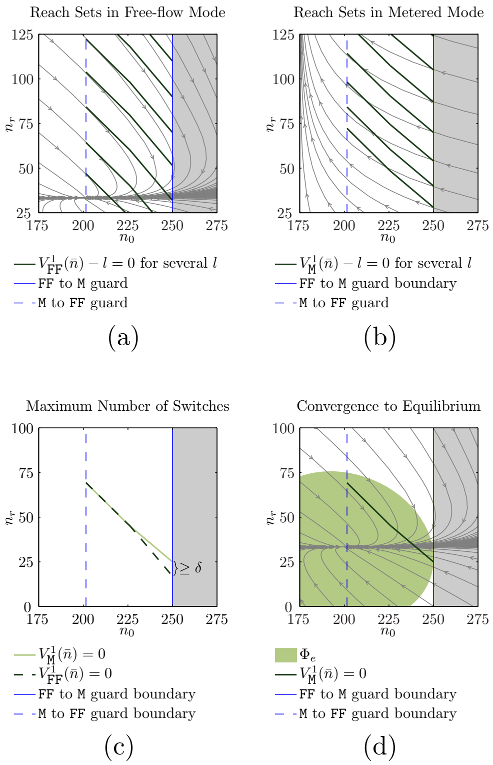
$$\phi_{\texttt{M}}(\bar{n}, l^*) \succeq 0 \implies n_r \leq N_r. \tag{61}$$

Equations (60) and (61) ensure that upon switching to metered mode, trajectories cannot become unsafe before first switching again to free-flow mode, thereby guaranteeing safety.

We provide results from the guard synthesis algorithm in Fig. 6(b) and Fig. 7. The guard synthesis algorithm iteratively increases δ from -3 to 8 and solves for new guards and reach sets at each step. The final guard guaranteeing C1)–C3) is $g_{\texttt{M}\to\texttt{FF}} = 201.5306 - n_0$.

6. Conclusions

We have presented a technique for synthesizing switching guards for hybrid systems. Our approach synthesizes guard sets as semialgebraic sets that trigger transitions between modes of the hybrid system to guarantee a state-space safety constraint. Lemma 1 and Proposition 1 present the requirements on guard sets and reach sets to ensure safety and prevent Zeno executions. Theorem 1 encodes these requirements into an SOS program which can be solved using an iterative algorithm. As is the case for all bilinear optimization problems widely



(a)

(b)

(c)

(d)

Figure 7: (a) and (b) show $V^1_q(\bar{n}) - l$ for $q \in \{\mathtt{FF}, \mathtt{M}\}$ for several values of l and the guards, (c) indicates that $\phi_{\mathtt{FF}}$ and $\phi_{\mathtt{M}}$ ensures the onramp queue n_r reduces by at least δ vehicles when switching from free-flow to metered and back to free-flow mode, and (d) shows Φ_e and $V^1_{\mathtt{M}}(\bar{n}) = 0$. A trajectory that switches to free-flow mode within Φ_e will converge to the equilibrium safely with no more switching. Note the axes scales.

used in practice, convergence to a solution is not guaranteed. Nonetheless, we demonstrate successful use of this approach on several examples.

In the theoretical development of our approach, we assume reach sets are readily available. In the examples, we compute reach sets by ensuring invariance of the reach set. While this approach is particularly suited for inclusion in the SOS guard synthesis algorithm, many other approaches to reach set computation exists as discussed in the Introduction. Future research should investigate how these approaches can be incorporated in the SOS approach. In addition, we were able to obtain certain performance guarantees in the application to highway onramp metering via slight modification of the conditions in Theorem 1. It would be beneficial to generalize such performance guarantees to broader classes of systems. Finally, it is often desirable to achieve a liveness condition whereby the system is guaranteed to reach a certain region of the state space. Techniques similar to those presented in this paper using underapproximations of reach sets may be applicable to this liveness problem.

7. Acknowledgements

We thank Professor Andy Packard for his insightful comments and suggestions regarding implementation of SOS programs. This research was supported in part by the National Science Foundation under grant ECCS-1101876 and by the Air Force Office of Scientific Research under grant FA9550-11-1-0244.

References

[1] E. Asarin, O. Bournez, T. Dang, O. Maler, and A. Pnueli, "Effective synthesis of switching controllers for linear systems," *Proceedings of the IEEE*, vol. 88, pp. 1011–1025, July 2000.

[2] I. Mitchell and C. Tomlin, "Level set methods for computation in hybrid systems," in *Hybrid Systems: Computation and Control*, vol. 1790 of *Lecture Notes in Computer Science*, pp. 310–323, Springer Berlin/Heidelberg, 2000.

[3] A. Kurzhanski and P. Varaiya, "Ellipsoidal techniques for reachability analysis: internal approximation," *Systems & Control Letters*, vol. 41, no. 3, pp. 201–211, 2000.

[4] E. Haghverdi, P. Tabuada, and G. J. Pappas, "Bisimulation relations for dynamical, control, and hybrid systems," *Theor. Comput. Sci.*, vol. 342, pp. 229–261, Sept. 2005.

[5] M. Boccadoro, Y. Wardi, M. Egerstedt, and E. Verriest, "Optimal control of switching surfaces in hybrid dynamical systems," *Discrete Event Dynamic Systems*, vol. 15, pp. 433–448, 2005.

[6] S. Prajna, A. Jadbabaie, and G. Pappas, "A framework for worst-case and stochastic safety verification using barrier certificates," *IEEE Transactions on Automatic Control*, vol. 52, no. 8, pp. 1415–1428, 2007.

[7] S. Jha, S. Gulwani, S. A. Seshia, and A. Tiwari, "Synthesizing switching logic for safety and dwell-time requirements," in *Proceedings of the 1st ACM/IEEE International Conference on Cyber-Physical Systems*, pp. 22–31, 2010.

[8] M. Althoff, A. Rajhans, B. H. Krogh, S. Yaldiz, X. Li, and L. Pileggi, "Formal verification of phase-locked loops using reachability analysis and continuization," *Commun. ACM*, vol. 56, pp. 97–104, Oct. 2013.

[9] C. Tomlin, J. Lygeros, and S. Sastry, "A game theoretic approach to controller design for hybrid systems," *Proceedings of the IEEE*, vol. 88, pp. 949–970, Jul 2000.

[10] J. Lygeros, C. Tomlin, and S. Sastry, "Controllers for reachability specifications for hybrid systems," *Automatica*, vol. 35, pp. 349–370, 1999.

[11] K. Margellos and J. Lygeros, “Hamilton–Jacobi formulation for reach–avoid differential games,” *IEEE Transactions on Automatic Control*, vol. 56, pp. 1849–1861, Aug 2011.

[12] I. Mitchell, A. Bayen, and C. Tomlin, “A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games,” *IEEE Transactions on Automatic Control*, vol. 50, no. 7, pp. 947–957, 2005.

[13] O. Shakernia, G. J. Pappas, and S. Sastry, “Decidable controller synthesis for classes of linear systems,” in *Hybrid Systems: Computation and Control*, pp. 407–420, Springer, 2000.

[14] Z. Jarvis-Wloszek, R. Feeley, W. Tan, K. Sun, and A. Packard, “Some controls applications of sum of squares programming,” in *Proceedings of the 42nd IEEE Conference on Decision and Control*, vol. 5, pp. 4676–4681, Dec. 2003.

[15] U. Topcu, A. Packard, and P. Seiler, “Local stability analysis using simulations and sum-of-squares programming,” *Automatica*, vol. 44, no. 10, pp. 2669–2675, 2008.

[16] M. M. Tobenkin, I. R. Manchester, and R. Tedrake, “Invariant funnels around trajectories using sum-of-squares programming,” in *Proceedings of the 18th IFAC World Congress, extended version available online: arXiv:1010.3013 [math.DS]*, 2011.

[17] P. A. Parrilo, “Semidefinite programming relaxations for semialgebraic problems,” *Mathematical Programming Ser. B*, vol. 96, no. 2, pp. 293–320, 2003.

[18] S. Coogan and M. Arcak, “Guard synthesis for safety of hybrid systems using sum of squares programming,” in *Proceedings of the 51st IEEE Conference on Decision and Control*, pp. 6138–6143, 2012.

[19] J. P. Aubin, *Viability theory.* Springer, 2009.

[20] R. Goebel, R. G. Sanfelice, and A. R. Teel, *Hybrid Dynamical Systems: modeling, stability, and robustness.* Princeton University Press, 2012.

[21] S. Prajna, A. Papachristodoulou, P. Seiler, and P. A. Parrilo, *SOSTOOLS: Sum of squares optimization toolbox for MATLAB*, 2004.

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

[23] F. Permenter and P. Parrilo, “Basis selection for SOS programs via facial reduction,” *Under review*, 2014.

[24] J. Lofberg, “Pre- and post-processing sum-of-squares programs in practice,” *IEEE Transactions on Automatic Control*, vol. 54, pp. 1007–1011, May 2009.

[25] F. Blanchini, “Set invariance in control,” *Automatica*, vol. 35, no. 11, pp. 1747–1767, 1999.

[26] C. F. Daganzo, “The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory,” *Transportation Research Part B: Methodological*, vol. 28, no. 4, pp. 269–287, 1994.

[27] G. Gomes, R. Horowitz, A. A. Kurzhanskiy, P. Varaiya, and J. Kwon, “Behavior of the cell transmission model and effectiveness of ramp metering,” *Transportation Research Part C: Emerging Technologies*, vol. 16, no. 4, pp. 485–513, 2008.

[28] G. Gomes and R. Horowitz, “Optimal freeway ramp metering using the asymmetric cell transmission model,” *Transportation Research Part C: Emerging Technologies*, vol. 14, no. 4, pp. 244–262, 2006.