2009 American Control Conference
Hyatt Regency Riverfront, St. Louis, MO, USA
June 10-12, 2009

A Separation Principle for a Class of Hybrid Automata on a Partial Order

Domitilla Del Vecchio, Michael Malisoff and Rajeev Verma

Abstract— **We consider a parallel composition of two order preserving hybrid automata with imperfect state information. We show that the order preserving properties of the dynamics lead to a separation principle between state estimation and control under safety specifications. We provide a dynamic feedback algorithm that is guaranteed to terminate and whose complexity scales with the number of continuous variables.**

I. Introduction

This note considers the dynamic feedback problem for hybrid automata with imperfect state information under safety specifications. There is an extensive literature on static control synthesis under safety specifications [2], [10], [12] and state estimation [1], [3], [5], [9] for hybrid systems. The control problem under safety specifications can be addressed by computing the set of states that lead to an unsafe configuration independently of an input choice, called the *capture set*. Then, a static feedback is computed that guarantees that the state never enters the capture set. Computational constraints usually limit the system to four or five continuous variables and to two or three discrete states. Furthermore, the usual algorithms are not guaranteed to terminate [10].

Initial work on safety control of hybrid systems with imperfect or partial state information can be found in [6]–[8]. In [6], a controller that relies on a state estimator is proposed for finite state systems. The results are extended to rectangular hybrid automata with imperfect state information. The proposed algorithm has exponential complexity in the size of the system, owing to the need for computing the capture set. In [8], a partial order approach for the design of computationally efficient state estimation and control algorithms was proposed. However, [8] only considered discrete dynamic feedback, and it did not provide an algorithm for computing the capture set. Initial results on the efficient computation of this set for block triangular order preserving hybrid automata in discrete time are in [7]. However, no separation principle was stated and the algorithms only provide over-approximations of the capture set.

In this note, we consider the parallel composition of two hybrid automata whose flows preserve a predefined ordering on the state and input sets. This structure is motivated by the modeling of the dynamics of agents evolving on prespecified paths in a network of routes. This is the case, for example, for the dynamics of vehicles along their lanes. In these systems, conflicts arise at intersections, mergings, traffic circles, etc. Therefore, an unsafe set is well represented by a box in the position space. We show that the capture set is the intersection of a finite number of capture sets obtained with inputs fixed at the corners of the feasible set of inputs. This implies that the state feedback controller that keeps the state out of the capture set never switches on the boundary of the capture set, and that it always lies on the corners of the feasible input set. Since our control algorithm also terminates, we provide a class of hybrid automata for which the control synthesis problem is decidable. The structure of the capture set allows us to obtain the dynamic feedback control map by basically just replacing the state by its estimate in the formula for the static feedback map. This is a separation principle between state estimation and control.

Del Vecchio and Verma were supported by NSF CAREER Award CNS-0642719. Malisoff was supported by NSF/DMS Grant 0708084. Del Vecchio and Verma are with the Systems Laboratory at the University of Michigan, 1301 Beal Avenue, 4417 EECS Building, Ann Arbor, MI 48109, {ddv,rajverma}@umich.edu. Malisoff is with the Department of Mathematics, Louisiana State University, Baton Rouge, LA 70803-4918, malisoff@lsu.edu.

II. Preliminaries: Partial Orders and Class of Systems

A partial order [4] is a set P with a partial order relation "$\leq$", denoted by $(P, \leq)$. For all $x, w \in P$, the $\sup\{x, w\}$, denoted $x \vee w$, is the smallest element that is larger than both x and w. The $\inf\{x, w\}$, denoted $x \wedge w$, is the largest element that is smaller than both x and w. If $S \subseteq P$, $\bigvee S := \sup S$ and $\bigwedge S := \inf S$. If a partially ordered set $(X, \leq)$ is such that $x \wedge w \in X$ and $x \vee w \in X$ for all $x, w \in X$, then $(X, \leq)$ is a *lattice*. An *interval sublattice* of $(P, \leq)$ is a lattice given by $[L, U] = \{w \in P \mid L \leq w \leq U\}$ for some $L, U \in P$.

Let $(P, \leq)$ and $(Q, \leq)$ be partially ordered sets. A map $f : P \to Q$ is an *order preserving map* provided $x \leq w \implies f(x) \leq f(w)$. If $(Z, \leq)$ is another partial order, then a function $f : P \times Q \to Z$ is called order preserving with respect to the x variable if $x \mapsto f(x, \bar{y})$ is order preserving for each $\bar{y} \in Q$; order preserving with respect to y is defined analogously. The *power set* of a set S (denoted by 2^S) is the set of all subsets of S, ordered by inclusion. This partial order is denoted by $(2^S, \subseteq)$. We always use the partial order $(\mathbb{R}^d, \leq)$ defined by: $w \leq z$ if and only if $w_i \leq z_i$ for all $i \in \{1, ..., d\}$.

Let $\mathcal{U}$ be any compact set and $\mathcal{F}(\mathcal{U})$ denote the set of all piecewise continuous functions $\mathbf{u} : \mathbb{R} \to \mathcal{U}$. We establish the partial order $(\mathcal{F}(\mathcal{U}), \leq)$ by defining $\mathbf{u^a} \leq \mathbf{u^b}$ provided that $\mathbf{u^a}(t) \leq \mathbf{u^b}(t)$ for all $t \in \mathbb{R}$, for all $\mathbf{u^a}, \mathbf{u^b} \in \mathcal{F}(\mathcal{U})$.

Definition 1: A *hybrid automaton (with input and imperfect state information)* is a tuple $H = (X, Q, \mathcal{U}, O, f, R, h)$, in which the set X of *continuous variables* is a subset of a Euclidean space, Q is a finite set of *modes*, $\mathcal{U}$ is a *continuous set of inputs*, O is a *continuous set of outputs*, $f : X \times Q \times \mathcal{U} \to X$ is a vector field, $R : X \times \mathcal{U} \to Q$ is the *mode reset map*, and $h : O \to 2^X$ is the *output map*.

The mode reset map R is defined as $R(x, u) := q$ if $(x, u) \in \mathrm{Dom}(q)$, in which $\mathrm{Dom} : Q \to 2^{X \times \mathcal{U}}$ is a map that attaches to

a mode the set of continuous states and inputs in which the mode holds. We assume that $\bigcup_{q\in Q}\text{Dom}(q) = X\times\mathcal{U}$ and that $\text{Dom}(q^a)\cap\text{Dom}(q^b) = \emptyset$ for all $q^a \neq q^b$. This guarantees that at any point (x,u) in the continuous set of state and input variables, there is always a unique mode that holds. When $X \subseteq \mathbb{R}^d$ for some d, we have $\dot{x} = f(x,q,u)$, in which $x = (x_1,...,x_d)$ and thus $f(x,q,u) = (f_1(x,q,u),...,f_d(x,q,u))$ with $f_i(x,R(x,u),u) \in \mathbb{R}$. For an output measurement $z \in O$, the function $h(z)$ returns the set of all possible continuous states that may have generated such an output measurement. We refer to the set $h(z)$ as the set of continuous states compatible with output observation z.

Let $\mathbf{z} : [0,\infty) \to O$ denote any output signal of H. Let $\mathcal{F}_{\mathbf{u}}(O)$ denote the set of all possible output measurements $\mathbf{z} : [0,\infty) \to O$ from H obtained using a given input $\mathbf{u} \in \mathcal{F}(\mathcal{U})$. We use $t \mapsto \phi(t,x,\mathbf{u})$ to denote the flow (or trajectory) of H starting at initial condition $x \in X$ at initial time zero, when input signal $\mathbf{u}$ is applied to H. When $\mathbf{u}$ is a constant, say $\mathbf{u}(t) \equiv \bar{u} \in \mathcal{U}$, we write $\phi(t,x,\bar{u})$ to mean $\phi(t,x,\mathbf{u})$. Since there is no continuous state reset, the flow is continuous with respect to time. We also use the notation $\phi_{w_1,...,w_k}(t,x,\mathbf{u})$ to denote the time evolution of any subset $\{w_1,...,w_k\}$ of the continuous variables x_i.

Consider the partial order on a Euclidean space defined by componentwise ordering and the partial order $(\mathcal{F}(\mathcal{U}),\leq)$.

Definition 2: We say that $H = (X,Q,\mathcal{U},O,f,R,h)$ is *order preserving* provided there exist constants $u_m, u_M \in \mathbb{R}$ and a positive constant γ such that the following hold:

(i) $\mathcal{U} = [u_m, u_M] \subset \mathbb{R}$;
(ii) The flow $\phi(t,x,\mathbf{u})$ is order preserving with respect to the x variable and with respect to the $\mathbf{u}$ variable;
(iii) $f_1(x,R(x,u),u) \geq \gamma$ for all $(x,u) \in X\times\mathcal{U}$;
(iv) For all $z \in O$, $h(z) = [\wedge h(z), \vee h(z)] \subseteq X$.

A sufficient condition for item (ii) is that for all $(x,u) \in X\times\mathcal{U}$, the function $f(x,R(x,u),u)$ is order preserving with respect to the x variable and with respect to u [11].

The motivation for order preserving hybrid automata comes from modeling the longitudinal dynamics of vehicles that are constrained to given paths. In these cases, (i) models the fact that the amount of throttle and braking (or jerk) is physically bounded; (ii) models the fact that larger acceleration inputs lead to larger longitudinal speeds and larger longitudinal displacements; and (iii) can model the physical constraint that vehicles cannot move in reverse.

Example 1: Let $p \in \mathbb{R}$ represent the coordinate of a vehicle along its lane. The longitudinal dynamics of the vehicle can be modeled as $\ddot{p} = [\mathcal{R}^2/(J_w + \mathcal{M}\mathcal{R}^2)](f_w - f_{brake} - \frac{\rho_{air}}{2}C_D A_f U^2 - C_{rr}\mathcal{M}g - \mathcal{M}g\sin(\theta_{road}))$, in which $\mathcal{R}$ is the tire radius, J_w is the wheel inertia, $\mathcal{M}$ is the mass of the vehicle, $f_w = \tau_w\mathcal{R}$ where τ_w is the drive shaft output torque, f_{brake} is the brake force, ρ_{air} is the air density, C_D is the drag coefficient, A_f is the projected front area of the vehicle, U is the longitudinal vehicle velocity, C_{rr} is the rolling resistance coefficient, g is the gravity constant, and θ_{road} is the road gradient [13]. For automatic driving, f_w and f_{brake} are control inputs to the longitudinal dynamics of the vehicle. Set the total force $F = f_w - f_{brake}$, $\delta = [\mathcal{R}^2/(J_w + \mathcal{M}\mathcal{R}^2)](-\frac{\rho_{air}}{2}C_D A_f U^2 - C_{rr}\mathcal{M}g)$, $b = \mathcal{R}^2/(J_w + \mathcal{M}\mathcal{R}^2)$, and $\theta_{road} = 0$. Assuming that all of the parameters are exactly known, δ is also known to the on-board vehicle controller as the vehicle is aware of its own longitudinal velocity U. Thus, one can set $F = (u-\delta)/b$ to get the vehicle model

$$\dot{x}_1 = x_2,\ \dot{x}_2 = u, \tag{1}$$

in which we set $x_1 = p$, so the continuous set of variables is $X \subseteq [0,\infty)^2$. The flow of these dynamics preserves the ordering with respect to the (x_1,x_2) and the u variables.

Definition 3: Let $H_i = (X_i,Q_i,\mathcal{U}_i,O_i,f_i,R_i,h_i)$ for $i \in \{1,2\}$ be hybrid automata. Their *parallel composition* $H = H_1\|H_2$ is the hybrid automaton $H = (X,Q,\mathcal{U},O,f,R,h)$, in which $X = X_1\times X_2$, $Q = Q_1\times Q_2$, $\mathcal{U} = \mathcal{U}_1\times\mathcal{U}_2$, $O = O_1\times O_2$, $f = (f_1,f_2)$, $R = (R_1,R_2)$, and $h = (h_1,h_2)$.

In this note, we focus on the safety control of the parallel composition of order preserving hybrid automata with imperfect state information. We only consider the case of two parallel order preserving automata, but our results can be generalized to an arbitrary number of parallel automata.

III. Safety control with Imperfect State Information

Given a hybrid automaton $H = (X,Q,\mathcal{U},O,f,R,h)$ and a set of states $B \subseteq X$, consider the problem of designing a controller that, on the basis of the output measurements, prevents the state from entering B. We refer to B as a *bad set*. Let $\hat{x}(t,\hat{x}^0,\mathbf{u},\mathbf{z}) \in 2^X$ denote the set of all possible continuous states at time t given an initial set of possible states $\hat{x}^0 \in 2^X$, input $\mathbf{u}$, and output signal $\mathbf{z}$. We call $\hat{x}$ the state estimate, and we denote it by $\hat{x}(t)$ when $\hat{x}^0$, $\mathbf{u}$, and $\mathbf{z}$ are clear.

Problem 1: (*Dynamic Feedback Safety Control Problem*) Given a hybrid automaton $H = (X,Q,\mathcal{U},O,f,R,h)$, compute

$$\bar{W} := \left\{ \begin{array}{l} \hat{x}^0 \in 2^X \mid \exists\ \mathbf{u} \in \mathcal{F}(\mathcal{U}) \text{ such that } \forall \mathbf{z} \in \mathcal{F}_{\mathbf{u}}(O) \\ \text{and } \forall t \geq 0, \text{we have } \hat{x}(t,\hat{x}^0,\mathbf{u},\mathbf{z}) \cap B = \emptyset \end{array} \right\}$$

and a function $\bar{g} : \bar{W} \to 2^{\mathcal{U}}$ such that $\hat{x}(t,\hat{x}^0,\mathbf{u},\mathbf{z}) \cap B = \emptyset$ for all convex sets $\hat{x}^0 \in \bar{W}$, all $t \geq 0$, and all $\mathbf{z} \in \mathcal{F}_{\mathbf{u}}(O)$ when $\mathbf{u} \in \mathcal{F}(\mathcal{U})$ is chosen to satisfy $\mathbf{u}(r) \in \bar{g}(\hat{x}(r))$ for all $r \geq 0$.

This problem can be interpreted as one of finding a winning strategy for the control, which plays against the environment. The control map $\bar{g}$ is determined using $\bar{W}$. The computation of this set is in general hard; the complexity can be exponential in the size of the system [6]. Additionally, the known algorithms for its computation are not guaranteed to terminate. We solve Problem 1 for the parallel composition of two order preserving hybrid automata by showing that for such a class, a *separation principle* between state estimation and control design holds. We first define the following safety control problem with *perfect* state information:

Problem 2: (*Static Feedback Safety Control Problem*) Given a hybrid automaton $H = (X,Q,\mathcal{U},O,f,R,h)$ with $O = X$ and $h(z) \equiv z$, compute

$$W := \left\{ \begin{array}{l} x^0 \in X \mid \exists\ \mathbf{u} \in \mathcal{F}(\mathcal{U}) \text{ such that} \\ \forall t \geq 0, \text{we have } \phi(t,x^0,\mathbf{u}) \notin B \end{array} \right\}$$

(called the *escape set*) and a function $g : W \to 2^{\mathcal{U}}$ such that $\phi(t,x^0,\mathbf{u}) \notin B$ for all $x^0 \in W$ and all $t \geq 0$ when $\mathbf{u} \in \mathcal{F}(\mathcal{U})$ is chosen to satisfy $\mathbf{u}(r) \in g(\phi(r,x^0,\mathbf{u}))$ for all $r \geq 0$.

We define the separation principle between state estimation and control in the context of this note as follows, where we use V_i's to denote the admissible control values and the sets $S_{i,j}$ to specify where these values occur:

Definition 4: We say that the *(set-based) separation principle* holds for $H = (X, Q, \mathcal{U}, O, f, R, h)$ provided there exists a finite index set $\mathcal{I}$, an integer M, and subsets $S_{i,1}, \ldots, S_{i,M}, W_{i,1}, \ldots, W_{i,M} \subseteq X$ and $V_i \subseteq \mathcal{U}$ for each $i \in \mathcal{I}$ so that (i) the piecewise defined feedback $[\bar{g}(\hat{p}) := V_i$ if $(\hat{p} \cap S_{i,j} \neq \emptyset) \wedge (\hat{p} \cap W_{i,j} = \emptyset)$ for all $j]$ solves Problem 1 and (ii) the piecewise defined feedback $[g(p) := V_i$ if $(p \in S_{i,j}) \wedge (p \notin W_{i,j})$ for all $j]$ solves Problem 2.

Definition 4 means that given a static feedback map g that solves Problem 2, a dynamic feedback map $\bar{g}$ that solves Problem 1 is obtained by replacing the state by the state estimate and membership by set intersection. Hence, if a separation principle holds, then the dynamic feedback safety control problem is no harder than the static one. As we explain next, the sets $S_{i,j}$ and $W_{i,j}$ determine the escape set W and can be computed with linear complexity algorithms.

IV. Solution to Problems 1-2 and the Separation Principle

Consider $H = H_1 \| H_2$, in which the H_i's are order preserving hybrid automata with input sets $[u_{i,m}, u_{i,M}]$. Let $x_i = (x_{i,1}, ..., x_{i,n_i}) \in X_i \subseteq \mathbb{R}^{n_i}$ denote the continuous state of H_i. Given open intervals $\mathcal{I}_i = (L_i, U_i)$ for constants $L_i, U_i \in \mathbb{R}$ with $L_i < U_i$ for $i \in \{1, 2\}$, the bad set B is defined by

$$B = \{x \in X \mid x_{i,1} \in \mathcal{I}_i \text{ for } i = 1, 2\}. \tag{2}$$

The choice of a B that only restricts the $x_{i,1}$'s is motivated by the modeling of conflicts between vehicles at traffic intersections, railway/highway crossings, railway mergings, traffic circles, etc. In these cases, $x_{i,1}$ typically represents the position of a vehicle along its route, as illustrated by Example 1 for the case of vehicles driving in their lanes.

Our proof of a separation principle relies on the notion of the *capture set* $C := \{x \in X \mid \forall \mathbf{u} \in \mathcal{F}(\mathcal{U}), \exists t \geq 0 \text{ s.t. } \phi(t, x, \mathbf{u}) \in B\} = X \setminus W$. It is the set of all states x such that independently of the input $\mathbf{u}$, the flow takes the system from x to B in finite time. The static control map that solves Problem 2 prevents a state outside C from entering C. Let $x = (x_1, x_2) \in X$ with $x_i = (x_{i,1}, ..., x_{i,n_i})$ for $i = 1, 2$. Let $u_B := (u_{1,m}, u_{2,M})$ and $u_C := (u_{1,M}, u_{2,m})$, so $u_B, u_C \in \mathcal{U} = \mathcal{U}_1 \times \mathcal{U}_2$. Define $C_B = \{x \in X \mid \exists t \geq 0 \text{ s.t. } \phi(t, x, u_B) \in B\}$ and $C_C = \{x \in X \mid \exists t \geq 0 \text{ s.t. } \phi(t, x, u_C) \in B\}$.

Theorem 1: $C = C_B \cap C_C$.

To show Theorem 1, we first prove:

Lemma 1: Let $x^0 \in X$ be given. Set $R_C(x^0) = \bigcup_{t \geq 0} \phi_{x_{1,1}, x_{2,1}}(t, x^0, u_C)$, $R_B(x^0) = \bigcup_{t \geq 0} \phi_{x_{1,1}, x_{2,1}}(t, x^0, u_B)$, and $R(x^0) = \{(x_{1,1}, x_{2,1}) \in \mathbb{R}^2 \mid \exists p_1, p_2 \in \mathbb{R}$ such that $(x_{1,1}, p_1) \in R_C(x^0)$, $(x_{1,1}, p_2) \in R_B(x^0)$ and $p_1 \leq x_{2,1} \leq p_2\}$. Then, $\phi_{x_{1,1}, x_{2,1}}(t, x^0, \mathbf{u}) \in R(x^0)$ for all $t \geq 0$ and all $\mathbf{u} \in \mathcal{F}(\mathcal{U})$.

Proof: Fix $\mathbf{u} = (\mathbf{u_1}, \mathbf{u_2}) \in \mathcal{F}(\mathcal{U})$. For each $x^*_{1,1} > x^0_{1,1}$, we can use (iii) from Definition 2 for H_1 to find a $\bar{t} > 0$ such that $\phi_{x_{1,1}}(\bar{t}, x^0, u_{1,M}) = x^*_{1,1}$, since $f_{1,1} \geq \gamma$ everywhere. Similarly, let $\bar{t}' > 0$ be such that $\phi_{x_{1,1}}(\bar{t}', x^0, \mathbf{u_1}) = x^*_{1,1}$, and set $x^*_{2,1} = \phi_{x_{2,1}}(\bar{t}', x^0, \mathbf{u_2})$. Since $\phi_{x_{1,1}}$ is strictly increasing in time, $\phi_{x_{1,1}}(t, x^0, u_{1,M}) < x^*_{1,1}$ for all $t < \bar{t}$, and $\phi_{x_{1,1}}(t, x^0, \mathbf{u_1}) < x^*_{1,1}$ for all $t < \bar{t}'$. Since the flow is order preserving with respect to the input, we know that if $0 \leq t < \bar{t}$, then $\phi_{x_{1,1}}(t, x^0, \mathbf{u_1}) \leq \phi_{x_{1,1}}(t, x^0, u_{1,M}) < x^*_{1,1}$. This gives $\bar{t}' \geq \bar{t}$. Hence, by the order preserving property of the flow with respect to the input and its increasing property in the $x_{2,1}$ coordinate, $p_1 := \phi_{x_{2,1}}(\bar{t}, x^0, u_{2,m}) \leq \phi_{x_{2,1}}(\bar{t}, x^0, \mathbf{u_2}) \leq \phi_{x_{2,1}}(\bar{t}', x^0, \mathbf{u_2}) = x^*_{2,1}$. Hence, any $(x^*_1, x^*_2) \in \bigcup_{t \geq 0} \phi(t, x^0, \mathbf{u})$ admits $p_1 \leq x^*_{2,1}$ such that $(x^*_{1,1}, p_1) \in R_C(x^0)$. Analogous reasoning (that considers times $t \geq \bar{t}$ and $t \geq \bar{t}'$ instead of below these values, with the roles of m and M interchanged, and then concludes that $\bar{t}' \leq \bar{t}$) provides $p_2 \geq x^*_{2,1}$ satisfying $(x^*_{1,1}, p_2) \in R_B(x^0)$. ■

Proof: (*Theorem 1*). The inclusion $C \subseteq C_B \cap C_C$ is immediate from the definitions of these sets. To show that $C_B \cap C_C \subseteq C$, let $x^0 \in C_C \cap C_B$ and $\mathbf{u} \in \mathcal{F}(\mathcal{U})$ be given. Then there are times $\bar{t} > 0$ and $\bar{t}' > 0$ such that $c_2 := \phi_{x_{1,1}, x_{2,1}}(\bar{t}, x^0, u_C) \in \mathcal{I}_1 \times \mathcal{I}_2$ and $c_1 := \phi_{x_{1,1}, x_{2,1}}(\bar{t}', x^0, u_B) \in \mathcal{I}_1 \times \mathcal{I}_2$. By the definitions of $R_C(x^0)$ and $R_B(x^0)$, we have $c_1 \in R_B(x^0)$ and $c_2 \in R_C(x^0)$. By Lemma 1, $\phi_{x_{1,1}, x_{2,1}}(t, x^0, \mathbf{u}) \in R(x^0)$ for any $\mathbf{u} \in \mathcal{F}(\mathcal{U})$ and $t \geq 0$. Let $\Delta(x^0)$ denote the subset of $R(x^0)$ bounded by $R_C(x^0)$, $R_B(x^0)$, and the segment connecting c_1 with c_2. The flow $\phi_{x_{1,1}, x_{2,1}}(t, x^0, \mathbf{u})$ starts inside $\Delta(x^0)$, but since $\|\phi_{x_{1,1}, x_{2,1}}(t, x^0, \mathbf{u})\| \to \infty$ (by condition (iii) from Definition 2), it must exit $\Delta(x^0)$. Since the flow is continuous in time, there is a time at which $\phi_{x_{1,1}, x_{2,1}}(t, x^0, \mathbf{u})$ crosses $R_B(x^0)$, $R_C(x^0)$, or the segment connecting c_1 with c_2. However, by Lemma 1, the flow cannot cross $R_B(x^0) \cup R_C(x^0)$. Therefore, it must cross the segment connecting c_1 with c_2, which must lie in the convex set $\mathcal{I}_1 \times \mathcal{I}_2$. Therefore, $\phi_{x_{1,1}, x_{2,1}}(t, x^0, \mathbf{u})$ must enter $\mathcal{I}_1 \times \mathcal{I}_2$. Hence, $C \supseteq C_B \cap C_C$. ■

In all of what follows, we assume that the flow of H is continuous with respect to initial conditions. Note for later use that since B is open, both C_C and C_B are open. A consequence of Theorem 1 is that a static control map $g : X \setminus C \to 2^{\mathcal{U}}$ that solves Problem 2 is given by

$$g(x) := \{u_B\} \text{ if } x \notin C_B, \text{ or } \{u_C\} \text{ otherwise.} \tag{3}$$

A less restrictive map, which enforces a safe control input only on the boundary of C is given by

$$g(x) := \begin{cases} \{u_B\} & \text{if } x \in C_C \ \wedge \ x \in \partial C_B \\ \{u_C\} & \text{if } x \in C_B \ \wedge \ x \in \partial C_C \\ \{u_C, u_B\} & \text{if } \begin{cases} x \in \bar{C}_C \wedge x \notin C_C \\ \wedge x \in \bar{C}_B \wedge x \notin C_B \end{cases} \\ \mathcal{U} & \text{otherwise.} \end{cases} \tag{4}$$

Therefore, we can solve Problem 2 by simply computing C_C and C_B. The controller (4) never switches on ∂C. We next show that for order preserving hybrid automata, computing C_B and C_C is also sufficient to solve Problem 1.

Theorem 2: Let $\hat{x} \subset X$ be a convex set of continuous states. Then, the following are equivalent: (i) There exists an input $\mathbf{u} \in \mathcal{F}(\mathcal{U})$ such that $\phi(t, \hat{x}, \mathbf{u}) \cap B = \emptyset$ for all $t \geq 0$ and (ii) $\hat{x} \cap C_B = \emptyset$ or $\hat{x} \cap C_C = \emptyset$.

Proof: For any not necessarily convex set $\hat{x} \subset X$, if $\hat{x} \cap C_B = \emptyset$, then $\phi(t, \hat{x}, u_B) \cap B = \emptyset$ for all $t \geq 0$, and similarly for C_C. This follows from the definitions of C_B and C_C. We thus focus on showing that if $\hat{x} \cap C_B \neq \emptyset$ and $\hat{x} \cap C_C \neq \emptyset$, then

there is no $\mathbf{u} \in \mathcal{F}(\mathcal{U})$ such that $\phi(t, \hat{x}, \mathbf{u}) \cap B = \emptyset$ for all $t \geq 0$. If $x_{1,1} \geq U_1$, then for all $\mathbf{u} \in \mathcal{F}(\mathcal{U})$, all x_2, and all $t \geq 0$, we know that $\phi(t, x, \mathbf{u}) \notin B$, by condition (iii) from Definition 2. Hence, we may assume that $x_{1,1} \leq U_1$ for all $x \in \hat{x}$; otherwise, replace $\hat{x}$ with the convex set $\{x \in \hat{x} \mid x_{1,1} \leq U_1\}$ without relabeling. Let $x^0 \in C_C \cap \hat{x}$ and $\bar{x}^0 \in C_B \cap \hat{x}$, and suppose $\mathbf{u} \in \mathcal{F}(\mathcal{U})$ is such that $\phi(t, \hat{x}, \mathbf{u}) \cap B = \emptyset$ for all $t \geq 0$. Then, $\phi_{x_{1,1},x_{2,1}}(t_C, x^0, u_C) \in \mathcal{I}_1 \times \mathcal{I}_2$ for some $t_C > 0$ and $\phi_{x_{1,1},x_{2,1}}(t_B, \bar{x}^0, u_B) \in \mathcal{I}_1 \times \mathcal{I}_2$ for some $t_B > 0$. This implies that $R_C(x^0) \cap B \neq \emptyset$ and $R_B(\bar{x}^0) \cap B \neq \emptyset$.

It follows from Lemma 1 that for all $\mathbf{u} \in \mathcal{F}(\mathcal{U})$ and all $t \geq 0$, any $(\bar{x}_{1,1}, \bar{x}_{2,1}) \in \bigcup_{t \geq 0} \phi_{x_{1,1},x_{2,1}}(t, x^0, \mathbf{u})$ is such that for $(\bar{x}_{1,1}, p_1) \in R_C(x^0)$ we have that $\bar{x}_{2,1} \geq p_1$. Choose any $(\bar{x}_{1,1}, p_1) \in (\mathcal{I}_1 \times \mathcal{I}_2) \cap R_C(x^0)$. There always exists an $\bar{x}_{2,1}$ such that $(\bar{x}_{1,1}, \bar{x}_{2,1}) \in \bigcup_{t \geq 0} \phi_{x_{1,1},x_{2,1}}(t, x^0, \mathbf{u})$, by property (iii) from Definition 2 and the fact that $\bar{x}_{i,1} \geq x^0_{i,1}$ for $i = 1, 2$. Since $(\bar{x}_{1,1}, p_1) \in \mathcal{I}_1 \times \mathcal{I}_2$, it must be that $\bar{x}_{1,1} \in \mathcal{I}_1$. However, because $\phi_{x_{1,1},x_{2,1}}(t, x^0, \mathbf{u})$ does not intersect $\mathcal{I}_1 \times \mathcal{I}_2$ for any t, $(\bar{x}_{1,1}, \bar{x}_{2,1}) \notin \mathcal{I}_1 \times \mathcal{I}_2$, which in turn implies that $\bar{x}_{2,1} \geq U_2$, because $\bar{x}_{2,1} \geq p_1$. Similar arguments show that there is a $\bar{x}'_{2,1}$ such that $(\bar{x}_{1,1}, \bar{x}'_{2,1}) \in \bigcup_{t \geq 0} \phi_{x_{1,1},x_{2,1}}(t, \bar{x}^0, \mathbf{u})$ and $\bar{x}'_{2,1} \leq L_2$, based on considering points $(\bar{x}'_{1,1}, p'_1) \in (\mathcal{I}_1 \times \mathcal{I}_2) \cap R_B(\bar{x}^0)$.

Any other point $x \in \hat{x}$ must be such that when $\phi_{x_{1,1}}(t, x, \mathbf{u}) \in \mathcal{I}_1$ either $\phi_{x_{2,1}}(t, x, \mathbf{u}) \geq U_2$ or $\phi_{x_{2,1}}(t, x, \mathbf{u}) \leq L_2$ (since $\phi_{x_{1,1},x_{2,1}}(t, x, \mathbf{u})$ never enters $\mathcal{I}_1 \times \mathcal{I}_2$). Therefore, $\hat{x}$ is partitioned into two sets. For one of these sets, $(x_{1,1}, x_{2,1})$ will be taken above $\mathcal{I}_1 \times \mathcal{I}_2$, while for the other set it will be taken below $\mathcal{I}_1 \times \mathcal{I}_2$ by the same control input $\mathbf{u}$ that keeps $\hat{x}$ outside B. Thus, we have the disjoint union $\hat{x} = \hat{x}_{above} \bigcup \hat{x}_{below}$. More precisely, any $x^a \in \hat{x}_{above}$ admits $\bar{x}_{1,1} \in \mathcal{I}_1$ and $\bar{x}_{2,1} \geq U_2$ for which $(\bar{x}_{1,1}, \bar{x}_{2,1}) \in \bigcup_{t \geq 0} \phi_{x_{1,1},x_{2,1}}(t, x^a, \mathbf{u})$. Similarly, any $x^b \in \hat{x}_{below}$ admits $\tilde{x}_{1,1} \in \mathcal{I}_1$ and $\tilde{x}_{2,1} \leq L_2$ for which $(\tilde{x}_{1,1}, \tilde{x}_{2,1}) \in \bigcup_{t \geq 0} \phi_{x_{1,1},x_{2,1}}(t, x^b, \mathbf{u})$. As we showed in the previous paragraph, $\hat{x}_{above}$ and $\hat{x}_{below}$ are both nonempty. We show that this leads to a contradiction.

Since $\hat{x}$ is convex, the segment $\hat{s}$ joining any pairs $x^a \in \hat{x}_{above}$ and $x^b \in \hat{x}_{below}$ is again in $\hat{x}$. Also, since the first component $f_{1,1}$ for H_1 is bounded below by a positive constant, we can find a constant $\kappa > 0$ so that each point $z \in \hat{s}$ admits a unique time $t(z) \in [0, \kappa]$ so that $\phi_{x_{1,1}}(t(z), z, \mathbf{u}) = U_1$. Since the flow map is jointly continuous in time and the initial state, one readily shows that each constant $\mu > 0$ admits a constant $\Delta(\mu) > 0$ so that $\sup\{|t(z) - t(z')| : z, z' \in \hat{s}; \|z - z'\| \leq \Delta(\mu)\} \leq \mu$. Choose a constant $\delta_* > 0$ so that $\sup\{|\phi_{x_{2,1}}(t_1, z_1, \mathbf{u}) - \phi_{x_{2,1}}(t_2, z_2, \mathbf{u})| : t_1, t_2 \in [0, \kappa]; z_1, z_2 \in \hat{s}; |t_1 - t_2| \leq \delta_*; \|z_1 - z_2\| \leq \delta_*\} \leq (U_2 - L_2)/2$, and choose $z_b, z_a \in \hat{s}$ so that $z_a \in \hat{x}_{above}$, $z_b \in \hat{x}_{below}$, and $\|z_a - z_b\| \leq \min\{\delta_*, \Delta(\delta_*)\}$. By our choice of δ_*, we conclude that $|\phi_{x_{2,1}}(t(z_a), z_a, \mathbf{u}) - \phi_{x_{2,1}}(t(z_b), z_b, \mathbf{u})| \leq (U_2 - L_2)/2$. Since $\phi_{x_{2,1}}(t(z_a), z_a, \mathbf{u}) \geq U_2$ and $\phi_{x_{2,1}}(t(z_b), z_b, \mathbf{u}) \leq L_2$, this is a contradiction. This contradiction came from assuming that there is an input signal $\mathbf{u}$ that keeps $\phi(t, \hat{x}, \mathbf{u}) \cap B = \emptyset$ at all $t \geq 0$. Therefore such an input signal cannot exist. ■

Theorem 2 shows that a convex set $\hat{x}^0 \subseteq X$ intersects both C_B and C_C, if and only if each $\mathbf{u}$ admits an $x \in \hat{x}^0$ and a time $t^* > 0$ such that $\phi(t^*, x, \mathbf{u}) \in B$. This in turn is true if and only if each input $\mathbf{u}$ admits an output $\mathbf{z}$ and a time t^* such that $\hat{x}(t^*, \hat{x}^0, \mathbf{u}, \mathbf{z}) \cap B \neq \emptyset$, i.e., $\hat{x}^0 \in 2^X \setminus \bar{W}$.

Theorem 3: (*Solution to Problem 1*) Consider the hybrid automaton $H = H_1 || H_2$, in which H_1 and H_2 are order preserving hybrid automata. A convex set $\hat{x} \in 2^X$ is in $\bar{W}$ if and only if $\hat{x} \cap C_B = \emptyset$ or $\hat{x} \cap C_C = \emptyset$. Furthermore, assume that the initial state estimate $\hat{x}^0 \in \bar{W}$ is convex. Then, we can solve Problem 1 using the map $\bar{g} : \bar{W} \to 2^{\mathcal{U}}$ defined by $\bar{g}(\hat{x}) = \{u_B\}$ if $\hat{x} \cap C_B = \emptyset$ and $\bar{g}(\hat{x}) = \{u_C\}$ otherwise.

Theorem 3 follows from the remarks before the theorem, and because if $\hat{x}^0 \cap C_B = \emptyset$, then u_B will keep $\hat{x}(t)$ outside C_B for all t. Similarly, if $\hat{x}^0 \cap C_B$ is not empty and $\hat{x}^0 \in \bar{W}$, then $\hat{x}^0 \cap C_C = \emptyset$. Thus, u_C will be applied as long as $\hat{x}(t) \cap C_B \neq \emptyset$, which will keep $\hat{x}(t)$ outside C_C. If $\hat{x}(t) \cap C_B$ becomes empty again, then we fall back into the first case.

Corollary 1: (*Separation between state estimation and control*) For a hybrid automaton $H = H_1 || H_2$ with order preserving hybrid automata H_i and bad set B given by (2), a separation principle holds.

This corollary follows immediately from our Definition 4 for the separation principle, the map $\bar{g}$ in Theorem 3 and the static control map g in (3). A less restrictive alternative to the $\bar{g}(\hat{x})$ given in Theorem 3 that applies safe control inputs only when the set $\hat{x}$ hits the boundary of C_C or of C_B is:

$$\bar{g}(\hat{x}) := \begin{cases} \{u_B\} & \text{if } (\hat{x} \cap C_C \neq \emptyset) \ \wedge \ (\hat{x} \cap \partial C_B \neq \emptyset) \\ \{u_C\} & \text{if } (\hat{x} \cap C_B \neq \emptyset) \ \wedge \ (\hat{x} \cap \partial C_C \neq \emptyset) \\ \{u_C, u_B\} & \text{if } \begin{cases} (\hat{x} \cap \bar{C}_C \neq \emptyset) \wedge (\hat{x} \cap C_C = \emptyset) \\ \wedge (\hat{x} \cap \bar{C}_B \neq \emptyset) \wedge (\hat{x} \cap C_B = \emptyset) \end{cases} \\ \mathcal{U} & \text{otherwise.} \end{cases} \quad (5)$$

This map is the extension of (4) to the case of partial information. Even if restrictive, the map in Theorem 3 guarantees that any convex set $\hat{x}^0$ that starts in $\bar{W}$ (and that therefore by definition admits a control signal that keeps it outside B at all time) will never intersect B.

V. Algorithm Implementation

We next provide an algorithm for the symbolic computation of the sets C_B and C_C, which has linear complexity with the number of continuous variables. Set $\mathcal{U}_C := \{u = (u_1, u_2) \, | u_i \in \{u_{i,m}, u_{i,M}\}$ constant$\}$, i.e., the set of all constant inputs that lie at the corners of the feasible set of inputs $\mathcal{U} = [u_{1,m}, u_{1,M}] \times [u_{2,m}, u_{2,M}]$. With a view towards digital implementation, we illustrate our algorithm in discrete time. We assume that for each hybrid automaton H_i, the first component $f_{i,1}$ of f_i does not depend on $x_{i,1}$ and the discrete state reset map R_i also does not depend on $x_{i,1}$. This structure is satisfied, e.g., by chains of integrators and feedback linearizable systems after feedback linearization. It can model the longitudinal dynamics of vehicles, as well as mode transitions induced by speed and acceleration changes.

Set $\bar{x}_i = (x_{i,2}, ..., x_{i,n_i})$ for $i = 1, 2$. For any variable v, let v' denote its value at step $k + 1$ and v its value at step k. Letting $\Delta T > 0$ denote the discretization time, the discrete time version of the dynamics of each H_i we consider is $x'_{i,1} = x_{i,1} + F_{i,1}(\bar{x}_i, u_i)$, $\bar{x}'_i = \bar{F}_i(\bar{x}_i, u_i)$, where $F_{i,1}(\bar{x}_i, u_i) = f_{i,1}(\bar{x}_i, R_i(\bar{x}_i, u_i))\Delta T$ and $\bar{F}_i(\bar{x}_i, u_i)$ and $F_{i,1}(\bar{x}_i, u_i)$

are order preserving in $\bar{x}_i$ and u_i. Independently of the form of the $\bar{F}_i(\bar{x}_i, u_i)$'s, the key feature that allows the symbolic computation of the C_{u_C}'s is the structure of the discrete time dynamics. Set $\bar{F}_i^0(\bar{x}_i, u_i) := \bar{x}_i$ and $\bar{F}_i^{k+1}(\bar{x}_i, u_i) := \bar{F}_i(\bar{F}_i^k(\bar{x}_i, u_i), u_i)$ for $k = 0, 1, \ldots$.

Proposition 1: Set $L_i^k(\bar{x}_i, u_i) = L_i - \sum_{j=0}^{k-1} F_{i,1}(\bar{F}_i^j(\bar{x}_i, u_i), u_i)$ and $U_i^k(\bar{x}_i, u_i) = U_i - \sum_{j=0}^{k-1} F_{i,1}(\bar{F}_i^j(\bar{x}_i, u_i), u_i)$ for each $k \in \mathbb{N}$ and $i \in \{1, 2\}$. Then,

$$C_{u_C} = \left\{ x \in X \mid \exists k \geq 0 \text{ with } L_i^k(\bar{x}_i, u_{i,C}) < x_{i,1} < U_i^k(\bar{x}_i, u_{i,C}) \ \forall i \right\}$$

for each $u_C = (u_{1,C}, u_{2,C}) \in \mathcal{U}_C$.

Proof: The proof proceeds by iteratively computing the sets of points that are taken to B in k steps. For each $i \in \{1, 2\}$, we have $\{x_i \mid x'_{i,1} \in \mathcal{I}_i\} = \{x_i \mid L_i - F_{i,1}(\bar{x}_i, u_i) < x_{i,1} < U_i - F_{i,1}(\bar{x}_i, u_i)\}$. The set $\{x_i \mid L_i^1(\bar{x}_i, u_i) < x_{i,1} < U_i^1(\bar{x}_i, u_i)\}$ is the set of all points $x_i = (x_{i,1}, \bar{x}_i) \in X_i$ that are mapped to $\mathcal{I}_i$ in one step. Then, one computes the set of all points that are mapped to $\{x_i \mid L_i^1(\bar{x}_i, u_i) < x_{i,1} < U_i^1(\bar{x}_i, u_i)\}$ in one step, which is the set of all points that are mapped to $\mathcal{I}_i$ in two steps. This set is given by $\{x_i \mid L_i^1(\bar{x}'_i, u_i) < x'_{i,1} < U_i^1(\bar{x}'_i, u_i)\}$, which is equal to $\{x_i \mid L_i^1(\bar{F}_i(\bar{x}_i, u_i), u_i) < x_{i,1} + F_{i,1}(\bar{x}_i, u_i) < U_i^1(\bar{F}_i(\bar{x}_i, u_i), u_i)\}$. This leads to $\{x_i \mid L_i - F_{i,1}(\bar{x}_i, u_i) - F_{i,1}(\bar{F}_i(\bar{x}_i, u_i), u_i) < x_{i,1} < U_i - F_{i,1}(\bar{x}_i, u_i) - F_{i,1}(\bar{F}_i(\bar{x}_i, u_i), u_i)\}$. Proceeding inductively shows that if the set that is mapped to $\mathcal{I}_i$ in k steps is given by $\{x_i \mid L_i^k(\bar{x}_i, u_i) < x_{i,1} < U_i^k(\bar{x}_i, u_i)\}$, then the set that is mapped to $\mathcal{I}_i$ in $k+1$ steps is given by $\{x_i \mid L_i^k(\bar{F}_i(\bar{x}_i, u_i), u_i) - F_{i,1}(\bar{x}_i, u_i) < x_{i,1} < U_i^k(\bar{F}_i(\bar{x}_i, u_i), u_i) - F_{i,1}(\bar{x}_i, u_i)\} = \{x_i \mid L_i^{k+1}(\bar{x}_i, u_i) < x_{i,1} < U_i^{k+1}(\bar{x}_i, u_i)\}$. Since this holds for any $u \in \mathcal{U}_C$, the result follows. ∎

Since the dynamics of the system are order preserving with respect to the state, we can construct a state estimator that keeps track of only the lower and upper bounds of the current system states as opposed to keeping track of the entire set of current system states. Let us denote the update laws of hybrid automaton H_i by $F_i(x_i, q_i, u_i) := (x_{i,1} + F_{i,1}(\bar{x}_i, u_i), \bar{F}_i(\bar{x}_i, u_i))$ for $i \in \{1, 2\}$. Let $\vee\hat{x}_i$ and $\wedge\hat{x}_i$ denote the upper and lower bounds, respectively, of the set of possible current states $\hat{x}_i$. Let z_i be an output measurement of hybrid automaton H_i and let $h_i(z_i) = [\wedge h_i(z_i), \vee h_i(z_i)]$. Then a state estimator that updates $\vee\hat{x}_i$ and $\wedge\hat{x}_i$ is given by

$$\begin{aligned} \vee\hat{x}'_i &= \bigvee_{q_i \in \hat{q}_i} F_i(\vee\hat{x}_i, q_i, u_i) \wedge \vee h_i(z'_i) \\ \wedge\hat{x}'_i &= \bigwedge_{q_i \in \hat{q}_i} F_i(\wedge\hat{x}_i, q_i, u_i) \vee \wedge h_i(z'_i), \end{aligned} \tag{6}$$

where $\hat{q}_i = \{q_i \in Q_i \mid \exists x_i \in [\wedge\hat{x}_i, \vee\hat{x}_i] \text{ such that } (x_i, u_i) \in \text{Dom}(q_i)\}$. To implement the dynamic feedback, one needs to check whether the set $[\wedge\hat{x}, \vee\hat{x}]$ intersects the sets C_{u_C} given in Proposition 1.

Using the structure of the sets C_{u_C} in Proposition 1 and the order preserving property of the dynamics, it is possible to show that if $[\wedge\hat{x}, \vee\hat{x}] \cap C_{u_C} = \emptyset$ then applying u_C guarantees that $[\wedge\hat{x}', \vee\hat{x}'] \cap C_{u_C} = \emptyset$ (see [7] for details). Because the functions $\bar{F}_i(\bar{x}_i, u_i)$ and $F_{i,1}(\bar{x}_i, u_i)$ are order preserving in the argument $\bar{x}_i$, the functions $L_i^k(\bar{x}_i, u_i)$ and $U_i^k(\bar{x}_i, u_i)$ are order *reversing* in $\bar{x}_i$ for $i = 1, 2$. Hence, a sufficient condition

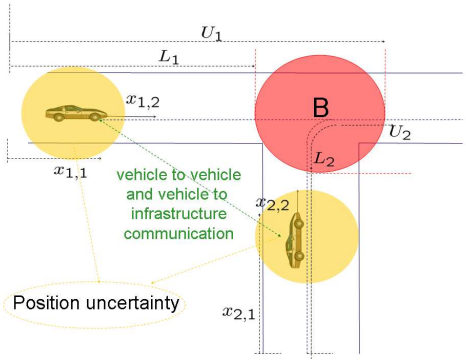

Fig. 1. Vehicles converging toward a traffic intersection. The bad set is defined to be the set of all vehicle 1/vehicle 2 configurations in which the vehicles are both in the red area.

guaranteeing that $[\wedge\hat{x}_{i,1}, \vee\hat{x}_{i,1}] \cap (L_i^k(\bar{x}_i, u_{i,C}), U_i^k(\bar{x}_i, u_{i,C})) = \emptyset$ for all k and all $(\bar{x}_1, \bar{x}_2)$ satisfying $\bar{x}_i \in [\wedge\hat{\bar{x}}_i, \vee\hat{\bar{x}}_i]$ is that for $i = 1, 2$,

$$\left[\bigwedge\hat{x}_{i.1}, \bigvee\hat{x}_{i,1}\right] \cap \left(L_i^k(\bigvee\hat{\bar{x}}_i, u_{i,C}), U_i^k(\bigwedge\hat{\bar{x}}_i, u_{i,C})\right) = \emptyset \ \ \forall k. \tag{7}$$

Also, because $F_{i,1}$ is bounded below by a positive constant (by condition (iii) from Definition 2), the sequences $\{L_i^k(\bar{x}_i, u_i)\}_{k \geq 1}$ and $\{U_i^k(\bar{x}_i, u_i)\}_{k \geq 1}$ are strictly decreasing to $-\infty$. Therefore, condition (7) does not need to be checked for an infinite number of k; it is enough to reach the smallest k for which $(\wedge\hat{x}_{1,1}, \wedge\hat{x}_{2,1}) \geq (U_1^k(\wedge\hat{\bar{x}}_1, u_{1,C}), U_2^k(\wedge\hat{\bar{x}}_2, u_{2,C}))$. This guarantees *termination* of the dynamic algorithm that computes the control map.

VI. Application Example

Consider the problem of designing a safety controller for two communicating vehicles converging to a traffic intersection subject to state uncertainty (Figure 1). The longitudinal dynamics of each vehicle is modeled as in equation (1). Hence, in terms of our earlier notation, $n_1 = n_2 = 2$. To satisfy speed limitations and to prevent a vehicle from going in reverse, we model each of the two vehicles as a hybrid automaton with three modes, as shown in Figure 2, in which $u_i \in [u_{i,m}, u_{i,M}]$, $i \in \{1, 2\}$. The system of two vehicles converging to the intersection is thus given by the parallel composition of two equal hybrid automata as in Figure 2. It can be verified that each of the two hybrid automata is order preserving and that it is continuous with respect to initial conditions. The measurement model is chosen such that $z_i \in [x_i - \Delta, x_i + \Delta]$ for $i \in \{1, 2\}$ in which $\Delta \in \mathbb{R}^2$ is a bounded uncertainty. The algorithms of Section V can thus be implemented to symbolically compute the sets C_C and C_B, and to compute the state estimate in equations (6).

Figure 3 shows the capture set in three dimensions. Figure 4 shows an execution of the controlled system when we apply the dynamic feedback in equation (5). The algorithm is simulated on a 2 Ghz Pentium Core Duo machine using Matlab. It takes 6 seconds for the simulation to run 400 time iterations. While the case in which one vehicle is closer to the intersection and running at higher speed than the other is trivially solved by having it cross the intersection first to prevent a collision (assuming the system configuration is

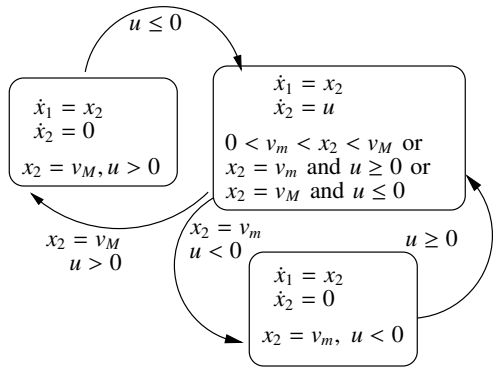

Fig. 2. Hybrid automaton modeling the longitudinal dynamics of a vehicle approaching a traffic intersection.

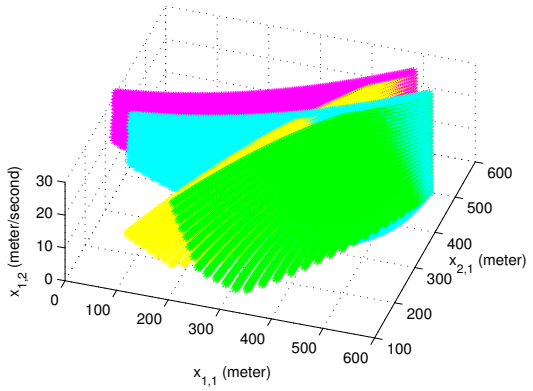

Fig. 3. Capture set in $(x_{1,1}, x_{2,1}, x_{1,2})$ space for a constant $x_{2,2}$. The bad set B is such that $L_i = 500$ and $U_i = 550$ for $i \in \{1, 2\}$. The pink and blue surfaces represent the boundary of the set C_C, while the yellow and green surfaces represent the boundary of the set C_B. All of the surfaces extend to $-\infty$ in the $(x_{1,1}, x_{2,1})$ plane. The capture set C is the set bounded by the yellow and blue surfaces.

outside of the capture set), the case in which it is closer but running at lower speed than the other is non-trivial. This case is automatically resolved by our algorithm as the capture set depends on the speeds: for the same position pair, two different speed pairs can result in different precedence rules for the vehicles. These rules must be strictly respected in order to maintain safety.

The only approximation introduced for obtaining linear complexity in the number of variables occurs when (7) is used to check intersection between the state estimate set and C_B and C_C. The plots of Figure 4 show that the estimated set and thus the state is taken very close to B. Hence, the dynamic control algorithm as a whole is tight. This contrasts with [7], where the dynamic control algorithms lead to quite conservative dynamic controllers.

VII. Conclusions

We proved a separation principle between state estimation and safety control for hybrid automata with imperfect state information whose flows preserve the ordering with respect to the state and the input. This led to a dynamic control algorithm that has linear complexity in the number of continuous variables and that is guaranteed to terminate. We applied the algorithm to a collision avoidance problem.

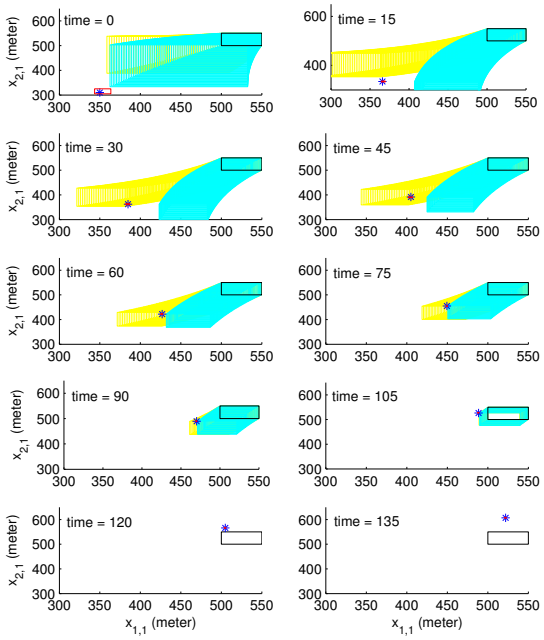

Fig. 4. The plots show the behavior of the system trajectory in the positions plane. The yellow set represents a slice of the set C_B in the position plane for the current speeds of the vehicles. The blue set represents a slice of the set C_C in the position plane for the current speeds of the vehicles. The star represents the position $(x_{1,1}, x_{2,1})$ and the red rectangle around it the uncertainty as given by the state estimator. The bad set B is the black box.

References

[1] A. Alessandri and P. Coletta. Design of Luenberger observer for a class of hybrid linear systems. In *Hybrid Systems: Computation and Control*, pages 7–18. Springer Berlin, Heidelberg, Germany, 2001.

[2] E. Asarin, O. Maler, and A. Pnueli. Symbolic controller synthesis for discrete and timed systems. In *Hybrid Systems II*, pages 1–20. Springer Berlin, Heidelberg, Germany, 1995.

[3] A. Balluchi, L. Benvenuti, M. D. Di Benedetto, and A. Sangiovanni-Vincentelli. Design of observers for hybrid systems. In *Hybrid Systems: Computation and Control*, pages 76–89. Springer Berlin, Heidelberg, Germany, 2002.

[4] B. A. Davey and H. A. Priestley. *Introduction to Lattices and Order*. Cambridge University Press, Cambridge, UK, 2002.

[5] E. De Santis, M. D. Di Benedetto, and G. Pola. On observability and detectability of continuous-time linear switching systems. In *Proc. IEEE Conf. Decision Control*, pages 5777–5782. 2003.

[6] M. De Wulf, L. Doyen, and J.-F. Raskin. A lattice theory for solving games of imperfect information. In *Hybrid Systems: Computation and Control*, pages 153–168. Springer Berlin, Heidelberg, Germany, 2006.

[7] D. Del Vecchio. Observer-based control for block-triangular hybrid automata. In *Proc. IEEE Conf. Decision Control*, pages 1782–1788. 2007.

[8] D. Del Vecchio. A partial order approach to discrete dynamic feedback in a class of hybrid systems. In *Hybrid Systems: Computation and Control*, pages 159–173. Springer Berlin, Heidelberg, Germany, 2007.

[9] D. Del Vecchio, R. M. Murray, and E. Klavins. Discrete state estimators for systems on a lattice. *Automatica*, 42(2):271–285, 2006.

[10] O. Shakernia, G. J. Pappas, and S. Sastry. Semi-decidable synthesis for triangular hybrid systems. In *Hybrid Systems: Computation and Control*, pages 487–500. Springer Berlin, Heidelberg, Germany, 2001.

[11] H. L. Smith. *Monotone Dynamical Systems*. American Mathematical Society, Providence, RI, 1995.

[12] C. J. Tomlin, J. Lygeros, and S. Sastry. A game theoretic approach to controller design for hybrid systems. *Proceedings of the IEEE*, 88(7):949–970, 2000.

[13] R. Verma, D. Del Vecchio, and H. Fathy. Development of a scaled vehicle with longitudinal dynamics of a HMMWV for an ITS testbed. *IEEE/ASME Transactions on Mechatronics*, 13:46–57, February 2008.