Comparing Forward and Backward Reachability as Tools for Safety Analysis

Ian M. Mitchell

Department of Computer Science, University of British Columbia,
2366 Main Mall, Vancouver, BC, Canada V6T 1Z4
mitchell@cs.ubc.ca
http://www.cs.ubc.ca/~mitchell

Technical Report TR-2006-23[1]
April 3, 2007

Abstract. Using only the existence and uniqueness of trajectories for a generic dynamic system with inputs, we define and examine eight types of forward and backward reachability constructs. If the input is treated in a worst-case fashion, any forward or backward reach set or tube can be used for safety analysis, but if the input is treated in a best-case fashion only the backward reach tube always provides the correct results. Fortunately, forward and backward algorithms can be exchanged if well-posed reverse time trajectories can be defined. Unfortunately, backward reachability constructs are more likely to suffer from numerical stability issues, especially in systems with significant contraction—the very systems where forward simulation and reachability are most effective.

1 Introduction

Except for the simplest of examples, analytic verification of safety properties for continuous and hybrid systems is rarely possible. With the goal of broadening the applicability and automating the process, numerical methods for verifying or validating such properties have been the subject of much study. The approximation of reachable sets is one major category of such numerical methods. There are two fundamental types of reachability: forward and backward. Many algorithms have been proposed to compute one of these reachable sets (see Section 3), and some type of equivalence is often informally mentioned when a problem statement requires computation of the other set. The contribution of this paper is a detailed examination of the distinctions between these two sets. We make rather strong assumptions about the existence and uniqueness of trajectories, so it is the negative conclusions that hold the most significance.

Section 2 informally discusses the relationship between reachability and safety and defines some of the terminology, while Section 3 covers previous work. The body of the paper beings in Section 4 by examining the question of when various forms of forward and/or backward reachability can be used to prove system

---

[1] A condensed version of this work appears in [1].

safety: in some cases any form will do, but in some cases only one type of backward reachability gives the correct result. Section 5 then demonstrates that the formulation of the reachability problem and the algorithm used to solve it need not work in the same temporal direction, since forward and backward algorithms can be interchanged for systems which are reversible.

Unfortunately, these algorithms find only approximations. In Section 6, trajectory sensitivity analysis [2] is extended to examine the way in which numerical error may grow as these algorithms are run. Even though the backward reachability formulation may be applicable to more problems, we conclude that it is also more likely to experience numerical stability problems, regardless of whether it is implemented by a forward or backward algorithm.

2   Reachability and Safety Analysis

Safety analysis of a given system seeks to discover whether the system—or more accurately, the mathematical model representing the system—can enter a specified set of unsafe states. Since many systems operate correctly only when started correctly, a set of initial states is also often specified. Mathematically, we will specify a safety analysis problem by a tuple $S = (H, I, T)$ where H is a system model, I is the initial set, and T is the unsafe set or target.

We define the concepts more formally in Section 4, but informally *reachability analysis* seeks to determine whether trajectories of H can reach T from I. There are two types of analysis. *Forward reachability* starts with states in I and follows trajectories forward in time. If any of these trajectories intersect with T the system is unsafe. *Backward reachability* starts with states in T and follows trajectories backwards in time. If any of these backwards trajectories intersects I the system is unsafe.

Under these definitions it sounds like reachability can be determined by simulating individual trajectories of H, and simulation is in fact the typical method by which safety is disproved. Proof of safety, however, requires a guarantee that all possible trajectories have been investigated; a challenging task in continuous and hybrid systems where the number of states is infinite. Consequently, the term *reachability algorithm* is usually reserved for techniques that determine the set of states traversed by all trajectories emanating from a given set.

While the terms are not used consistently in the literature, we will in this paper distinguish two different objects that a reachability algorithm might generate: the *reach set* is the set of states occupied by trajectories at exactly some specified time, and the *reach tube* is the set of states traversed by those same trajectories over all times prior to and including the specified time. Thus, the reach tube always contains the reach set. Forward and backward versions of both reach sets and tubes can be specified.

While we examine their properties and appropriateness in terms of the fully specified safety analysis problem S, forward and backward reachable sets and tubes may be more or less appropriate for other tasks; for example, backward reach tubes for finding the set of states which achieves a target set despite the

unknown but bounded disturbance of exogenous inputs, or forward reach sets for demonstrating system liveness.

3   Related Work

There are two main classes of *direct* reachability algorithms, those that work directly with continuous representations. *Lagrangian* approaches represent the set or tube with information that moves with the flow of the underlying dynamics, and are typically described in terms of forward reachability. A few are designed for systems without inputs [3], many permit inputs which expand the size of the reach set [4–8] and some permit inputs which shrink the reach set [9]. The theory is often based on linear continuous dynamics, although most schemes have demonstrated computational extensions to handle the nonlinear case. These schemes have also shown the best scalability; for example, results for systems with hundreds of dimensions have been reported in [8, 3].

Eulerian approaches work with a discretization that is not moving with the dynamics (although it may be refined during computation), and are typically described in terms of backward reachability [10–14]. All schemes can support systems with inputs which expand the size of the reachable set, and most handle those that shrink it as well. The theory works directly with nonlinear systems, although scalability much beyond four dimensions has not been demonstrated.

The results in Section 6 are derived by a sensitivity analysis of trajectories. Lagrangian reachability algorithms that depend on numerical integration of these (or related) trajectories are clearly affected by such sensitivity. Despite the fact that they do not directly integrate the dynamics, Eulerian schemes will also be subject to similar numerical stability problems since the approximations that they use are based on the evolution of the underlying system.

In addition to the classes of direct algorithms, there are at least two other classes of *indirect* algorithms related to reachability for continuous and/or hybrid systems. Discretization of the state space and dynamics can yield a system on which discrete reachability algorithms can be run; for example [15, 16]. Alternatively, automated Lyapunov type methods can be used to prove invariance properties, such as [17, 18]. How the sensitivity results might apply to these algorithms has not yet been investigated.

The conclusions of Sections 4 and 5 apply to discrete systems as well; in fact, forward and backward reachability have been combined to verify some discrete systems (see [19] and the citations within). However, the nature of the approximation errors (if any) in discrete algorithms is different enough that Section 6 may not apply.

4   Comparing Forward and Backward Reachability

In this section we compare properties of forward and backward reachability for a very generically defined dynamic system H. Trajectories of H will be denoted

by

$$\xi_{\mathsf{H}}(s; z, t, u(\cdot)) : \mathbb{T} \to \mathbb{Z},$$

where $\mathbb{T} = [-\mathcal{T}, +\mathcal{T}] \subset \mathbb{R}$ is the time interval over which the trajectory exists. We employ the semicolon to distinguish between the argument s of ξ_{H} and the trajectory parameters: initial state $z \in \mathbb{Z}$, initial time $t \in \mathbb{T}$ and input signal $u(\cdot) \in \mathbb{U}$. For systems lacking an input signal, we omit it and denote trajectories as $\xi_{\mathsf{H}}(s; z, t)$. While such systems are often called "autonomous" in the control literature, we will avoid this term due to potential confusion with its standard but different meaning in the dynamic systems literature. Instead, we will call them zero input systems.Existence and uniqueness of trajectories ξ_{H} for various types of dynamic systems is a challenging subject by itself; for example, see [20, 21] and the citations within. To maintain the focus of this paper, we make the following rather idealized assumption.

Assumption 1. *For given initial state z, time t, and input signal $u(\cdot)$ drawn from an appropriate class, there exists a unique trajectory $\xi_{\mathsf{H}}(s; z, t, u(\cdot))$ for $s \in \mathbb{T}$.*

By making this strong but generic assumption, many of the results in the next two sections will apply to a broad group of dynamic systems, although we focus on continuous and hybrid systems. It is the negative conclusions that we draw that have the most relevance to future research—if a technique or formulation fails under such a strong but generic assumption, there is little point in pursuing its concrete implementation.

In *continuous systems*, the dynamics are given by an ordinary differential equation (ODE) of the form $\dot{z}(t) = f(z(t), u(t))$, where the state z is continuous. Typically $\mathbb{Z} \subseteq \mathbb{R}^d$, although some state variables may use other domains; for example, angles are often drawn from the periodic set $[0, 2\pi[$. If $f : \mathbb{Z} \times U \to T\mathbb{Z}$ is uniformly continuous, bounded and Lipschitz continuous in z for fixed u, then Assumption 1 is satisfied [22] for fixed $u(\cdot) \in \mathbb{U}$, where

$$\mathbb{U} \triangleq \{\phi : \mathbb{T} \to U \mid \phi(\cdot) \text{ is measurable}\} \tag{1}$$

and $U \subset \mathbb{R}^{d_u}$ is convex and compact. Consequently, we can specify a continuous system as a tuple $\mathsf{H_C} = (\mathbb{Z}, f, U)$.

The generalization to *hybrid systems* involves a form of hybrid automaton (HA) adapted from [14]: we simplify to a single input, but that input may affect the guards and domains. The state of a hybrid system is $z = (q, x) \in \mathbb{Q} \times \mathbb{X} = \mathbb{Z}$, where q is the discrete state and x is the continuous state. The full HA is given

by the tuple $\mathsf{H_H} = (\mathbb{Q}, \mathbb{X}, f, D, G, r, U)$, where

$\mathbb{Q}$	discrete states;
$\mathbb{X}$	continuous states;
$f : \mathbb{Q} \times \mathbb{X} \times U_C \to \mathrm{T}\mathbb{X}$	continuous dynamics (vector field);
$D : \mathbb{Q} \times U_D \to P(\mathbb{X})$	domain of continuous evolution;
$G : \mathbb{Q} \times \mathbb{Q} \times U_D \to P(\mathbb{X})$	guard conditions for discrete evolution;
$r : \mathbb{Q} \times \mathbb{Q} \times \mathbb{X} \times U \to \mathbb{X}$	reset function;
$U = (U_C, U_D)$	continuous and discrete input sets;

$$(2)$$

where $P(\mathbb{X})$ is the power set (set of all subsets) of $\mathbb{X}$. As in [14], we assume that the discrete inputs are constant during continuous evolution. We will call the boundaries of the domains and guards the *switching surfaces*. Although (2) allows switching to occur away from these surfaces when the interiors of the domains and guards intersect, in order to satisfy Assumption 1 $\mathsf{H_H}$ must be deterministic for fixed $u(\cdot)$—while the system may be nondeterministic with respect to state, all nondeterminism must be expressed through the input signals. Formal mathematical conditions under which Assumption 1 holds are available for some subclasses of this hybrid automata [20, 21]. At a minimum, Assumption 1 will require that $\mathsf{H_H}$ be non-Zeno and non-blocking, and that f statisfies conditions to ensure existence of the continuous components of the trajectories.

4.1 Maximal Reachability

When performing safety analysis with forward reachability, the single input's authority is used to make the reach set and tube as large as possible. We will use the subscript "1+" to denote a single input used to maximize the size of the reachable set and tube and call these constructs *maximal*.

$$F_{1+}(\mathsf{H}, S, t) \triangleq \{\hat{z} \in \mathbb{Z} \mid \exists u(\cdot) \in \mathbb{U}, \exists z \in S, \xi_{\mathsf{H}}(t; z, 0, u(\cdot)) = \hat{z}\}, \tag{3}$$

$$F_{1+}(\mathsf{H}, S, [0, t]) \triangleq \{\hat{z} \in \mathbb{Z} \mid \exists u(\cdot) \in \mathbb{U}, \exists z \in S, \exists s \in [0, t], \xi_{\mathsf{H}}(s; z, 0, u(\cdot)) = \hat{z}\}. \tag{4}$$

In the corresponding backward reachability problems, the input is used to drive as many states as possible towards the target set. The result is that the size of the reachable set and tube are again maximized.

$$B_{1+}(\mathsf{H}, S, t) \triangleq \{z \in \mathbb{Z} \mid \exists u(\cdot) \in \mathbb{U}, \exists \hat{z} \in S, \xi_{\mathsf{H}}(0; z, -t, u(\cdot)) = \hat{z}\}, \tag{5}$$

$$B_{1+}(\mathsf{H}, S, [0, t]) \triangleq \{z \in \mathbb{Z} \mid \exists u(\cdot) \in \mathbb{U}, \exists \hat{z} \in S, \exists s \in [0, t], \xi_{\mathsf{H}}(0; z, -s, u(\cdot)) = \hat{z}\}. \tag{6}$$

The relationships between these four sets is easy to establish and should not be surprising.

Proposition 1.

$$F_{1+}(\mathsf{H}, S, [0, t]) = \bigcup_{\hat{t} \in [0,t]} F_{1+}(\mathsf{H}, S, \hat{t}) \qquad B_{1+}(\mathsf{H}, S, [0, t]) = \bigcup_{\hat{t} \in [0,t]} B_{1+}(\mathsf{H}, S, \hat{t})$$

Proof. We prove only the forward reachability case; the backward reachability case is similar. Assume $\hat{z} \in F_{1+}(\mathsf{H}, S, [0, t])$. Since all the quantifiers in (4) are existential, it is equivalent to

$$F_{1+}(\mathsf{H}, S, [0, t]) = \{\hat{z} \in \mathbb{Z} \mid \exists s \in [0, t], \exists u(\cdot) \in \mathbb{U}, \exists z \in S, \xi_{\mathsf{H}}(s; z, 0, u(\cdot)) = \hat{z}\}.$$

Therefore, $\hat{z} \in F_{1+}(\mathsf{H}, S, s)$. Now assume $\hat{z} \in F_{1+}(\mathsf{H}, S, \hat{t})$ for some $u(\cdot)$. Using that same $u(\cdot)$ and letting $s = \hat{t}$ in (4), we get $\hat{z} \in F_{1+}(\mathsf{H}, S, [0, t])$ for all $t \geq \hat{t}$.
□

Reachability for zero input systems is a special case of maximal reachability; for example, the forward reach set is given by

$$F_0(\mathsf{H}, S, t) \triangleq \{\hat{z} \in \mathbb{Z} \mid \exists z \in S, \xi_{\mathsf{H}}(t; z, 0) = \hat{z}\}.$$

4.2 Minimal Reachability

The presence of the existential quantifiers in (3)–(6) begs an obvious question: what if some are replaced by universal quantifiers?[2] The interpretation is that we seek only those states that trajectories are forced to reach no matter what input is chosen. Consequently, the reachable sets and tubes are as small as possible, we use the "$1-$" notation, and call these constructs *minimal.*

$$F_{1-}(\mathsf{H}, S, t) \triangleq \{\hat{z} \in \mathbb{Z} \mid \forall u(\cdot) \in \mathbb{U}, \exists z \in S, \xi_{\mathsf{H}}(t; z, 0, u(\cdot)) = \hat{z}\}, \tag{7}$$

$$F_{1-}(\mathsf{H}, S, [0, t]) \triangleq \{\hat{z} \in \mathbb{Z} \mid \forall u(\cdot) \in \mathbb{U}, \exists z \in S, \exists s \in [0, t], \xi_{\mathsf{H}}(s; z, 0, u(\cdot)) = \hat{z}\}, \tag{8}$$

$$B_{1-}(\mathsf{H}, S, t) \triangleq \{z \in \mathbb{Z} \mid \forall u(\cdot) \in \mathbb{U}, \exists \hat{z} \in S, \xi_{\mathsf{H}}(0; z, -t, u(\cdot)) = \hat{z}\}, \tag{9}$$

$$B_{1-}(\mathsf{H}, S, [0, t]) \triangleq \{z \in \mathbb{Z} \mid \forall u(\cdot) \in \mathbb{U}, \exists \hat{z} \in S, \exists s \in [0, t], \xi_{\mathsf{H}}(0; z, -s, u(\cdot)) = \hat{z}\}. \tag{10}$$

Unfortunately, the properties that hold in the purely existential maximal case above no longer apply.

Proposition 2.

$$\bigcup_{\hat{t} \in [0,t]} F_{1-}(\mathsf{H}, S, \hat{t}) \subseteq F_{1-}(\mathsf{H}, S, [0, t]) \qquad \bigcup_{\hat{t} \in [0,t]} B_{1-}(\mathsf{H}, S, \hat{t}) \subseteq B_{1-}(\mathsf{H}, S, [0, t])$$

Proof. It is clear that the tubes must contain the sets. To show that the backward tube may be larger than the union of the backward sets, consider the purely continuous system H_1 for $x \in \mathbb{R}$

$$\dot{x} = b - 2, \qquad \text{where } |b| \leq 1, \tag{11}$$

with target set $S = [-1, 0]$. Since $\dot{x} \leq -1$, it is easy to see that $B_{1-}(\mathsf{H}_1, S, [0, t]) = [-1, t]$, a set that grows for all time. However, $B_{1-}(\mathsf{H}_1, S, t) = [3t - 1, t]$, and in particular $B_{1-}(\mathsf{H}_1, S, t) = \emptyset$ for all $t > 1/2$. Therefore, $\cup_{t>0} B_{1-}(\mathsf{H}_1, S, t) = [-1, 1/2]$ is a fixed set.

---

[2] Replacing *all* of the existential quantifiers with universal quantifiers can be accomplished trivially by looking at the complements of the sets (3)–(6).

To show that the forward tube may be larger than the union of the forward sets, let $\dot{x} = b + 2$ in (11) and use the same S. The analysis proceeds in the same manner: $\dot{x} \geq +1$, $F_{1-}(\mathsf{H}_1, S, [0, t]) = [-1, t]$ is unbounded as t grows but $\cup_{t>0} F_{1-}(\mathsf{H}_1, S, t) = [-1, 1/2]$ is a fixed set. $\qquad\square$

The problem arises because the choice of t in the reach set definitions is fixed before any other variable is quantified, while the choice of $s \in [0, t]$ in the reach tube definition occurs after all other variables are quantified. For maximal reachability all the quantifiers are existential, so their ordering does not matter. However, once the input's quantifier is changed to be universal, the order in which the trajectory's time interval is chosen matters a great deal.

We close with the observation that these same results apply to systems with competing inputs $u(\cdot)$ and $v(\cdot)$. As we will see in the next section, the backwards reach tube is the most useful of the minimal constructs, so we give its definition in this context.

$$B_2(\mathsf{H}, S, [0, t]) \triangleq \{z \in \mathbb{Z} \mid \exists v(\cdot) \in \mathbb{V}, \forall u(\cdot) \in \mathbb{U},$$
$$\exists \hat{z} \in S, \exists s \in [0, t], \xi_{\mathsf{H}}(0; z, -s, u(\cdot), v(\cdot)) = \hat{z}\}. \tag{12}$$

Additional care must be taken in the adversarial two input case when defining the sets $\mathbb{U}$ and $\mathbb{V}$ from which the input signals are drawn, so as to properly account for what information the two players have about each others' signal choice [23]. For example, in the purely continuous case one practical choice is to define $\mathbb{U}$ as in (1) and $\mathbb{V}$ as the class of nonanticipative strategies [13, 14]. In conjunction with this question of relative knowledge, the ordering of the inputs in (12) may need to be swapped for some problems.

4.3 Application to Safety Analysis

Having defined the maximal and minimal forward and backward reach sets and tubes, we examine which can be used to solve the safety problem $\mathsf{S} = (\mathsf{H}, I, T)$ under various assumptions about the input's behaviour. Throughout this section we assume that H satisfies Assumption 1.

Proposition 3. *The following properties are equivalent.*

1. H *is safe over horizon* $t \leq \mathcal{T}$ *for all possible inputs* $u(\cdot) \in \mathbb{U}$.
2. $F_{1+}(\mathsf{H}, I, s) \cap T = \emptyset$ *for all* $s \in [0, t]$.
3. $F_{1+}(\mathsf{H}, I, [0, t]) \cap T = \emptyset$.
4. $B_{1+}(\mathsf{H}, T, s) \cap I = \emptyset$ *for all* $s \in [0, t]$.
5. $B_{1+}(\mathsf{H}, T, [0, t]) \cap I = \emptyset$.

Proof. Proposition 1 establishes 2 $\iff$ 3 and 4 $\iff$ 5.

1 $\iff$ 2: We actually prove the converse. Assume that there exists $\hat{z} \in F_{1+}(\mathsf{H}, I, s) \cap T$ for some $s \in [0, t]$. By (3) there exists $u(\cdot) \in \mathbb{U}$ and $z \in I$ such that $\xi_{\mathsf{H}}(s; z, 0, u(\cdot)) = \hat{z}$. Since $z \in I$ and $\hat{z} \in T$, this trajectory reaches from I to T and thus H is unsafe. Now assume that H is unsafe, so that there

exists an input $u(\cdot) \in \mathbb{U}$ and initial condition $z \in I$ giving rise to a trajectory $\xi_{\mathsf{H}}(\cdot; z, 0, u(\cdot))$ such that $\xi_{\mathsf{H}}(s; z, 0, u(\cdot)) = \hat{z} \in T$ for some $s \in [0, t]$. By (3), $\hat{z} \in F_{1+}(\mathsf{H}, I, s)$ which implies that $F_{1+}(\mathsf{H}, I, s) \cap T \neq \emptyset$.

$1 \iff 4$: Again we prove the converse. Assume that there exists $z \in B_{1+}(\mathsf{H}, T, s) \cap I$ for some $s \in [0, t]$. By (5) there exists $u(\cdot) \in \mathbb{U}$ and $\hat{z} \in T$ such that $\xi_{\mathsf{H}}(0; z, -s, u(\cdot)) = \hat{z}$. Since $z \in I$ and $\hat{z} \in T$, this trajectory reaches from I to T and thus H is unsafe. Now assume that H is unsafe, so that there exists an input $u(\cdot) \in \mathbb{U}$ and initial condition $z \in I$ giving rise to a trajectory $\xi_{\mathsf{H}}(\cdot; z, 0, u(\cdot))$ such that $\xi_{\mathsf{H}}(s; z, 0, u(\cdot)) = \hat{z} \in T$ for some $s \in [0, t]$. Because H is time-independent, we can shift the time variable such that $\xi_{\mathsf{H}}(0; z, -s, u(\cdot)) = \hat{z}$. By (5), $z \in B_{1+}(\mathsf{H}, T, s)$ which implies that $B_{1+}(\mathsf{H}, T, s) \cap I \neq \emptyset$. $\qquad\square$

Based on this proposition, we can use any of the reach sets or tubes to demonstrate the safety of systems despite the actions of bounded exogenous inputs, or of systems without any inputs. The situation is not quite so favourable for proving the existence of an input which guarantees safety.

Proposition 4. *Given horizon $t \leq \mathcal{T}$, there exists an input $u(\cdot) \in \mathbb{U}$ (which may depend on initial state) that keeps H safe if and only if $B_{1-}(\mathsf{H}, T, [0, t]) \cap I = \emptyset$. Such an input may exist only if $B_{1-}(\mathsf{H}, T, s) \cap I = \emptyset$ for all $s \leq t$, but the converse is not necessarily true.*

Proof. We first prove the claims for the reach tube. Let $S = B_{1-}(\mathsf{H}, T, [0, t]) \cap I$.

H safe $\implies (S = \emptyset)$: Assume $z \in S$ but that H is safe for input $u(\cdot) \in \mathbb{U}$ and derive a contradiction. By (10), there exists $\hat{z} \in T$ and $s \in [0, t]$ such that $\xi_{\mathsf{H}}(0; z, -s, u(\cdot)) = \hat{z}$. But then this trajectory reaches from I to T under input $u(\cdot)$, which is a contradiction that H is safe for input $u(\cdot)$

H safe $\impliedby (S = \emptyset)$: Assume that $S = \emptyset$. Then for all $z \in I$, z is in the complement of $B_{1-}(\mathsf{H}, T, [0, t])$. Negating (10), there exists $u(\cdot) \in \mathbb{U}$ such that for all $\hat{z} \in T$ and $s \in [0, t]$, $\xi_{\mathsf{H}}(0; z, -s, u(\cdot)) \neq \hat{z}$; in other words, for any initial state in I, there is an input which gives rise to a trajectory which does not reach T during the interval $[0, t]$. Hence, there is an input $u(\cdot)$ which makes H is safe during this interval.

The "only if" claim for the reach set is a simple outcome of combining Proposition 2 and the proof for $\implies$ above. The converse is not necessarily true because for the reach set the input is chosen after the time t, and for larger t the input may drive trajectories right through the unsafe set T and out the other side [13]. An example can be found in [24]. $\qquad\square$

Based on this proposition, we can use the minimal backwards reach tube to prove the existence of a safe input for any state in the initial set. This same technique can be applied to systems with two adversarial inputs, although as noted above additional care must be taken in defining the sets from which the competing inputs are drawn, the order in which the inputs act, and conditions under which Assumption 1 is valid. While complicated to define, the two input minimal reach tube can be used successfully for safety verification [13]. Unfortunately, the same cannot be said of the minimal forward reachability constructs.

Proposition 5. *The forward minimal reach set and tube provide no information about whether there exists an input $u(\cdot) \in \mathbb{U}$ that makes* H *safe.*

Proof. Consider first the forward reach tube. Let $S = F_{1-}(\mathsf{H}, I, [0, t]) \cap T$. We show that any combination of S empty or nonempty with H safe or unsafe is possible. The two easy cases are the ones that should hold. For $S \neq \emptyset$ and H unsafe, take $I \cap T \neq \emptyset$. For $S = \emptyset$ and H safe, take $T = \emptyset$.

Now consider $S = \emptyset$. Then for all $\hat{z} \in T$, $\hat{z}$ is in the complement of $F_{1-}(\mathsf{H}, I, [0, t])$. Negating (8), there exists $u(\cdot) \in \mathbb{U}$ such that for all $z \in I$ and $s \in [0, t]$, $\xi_{\mathsf{H}}(s; z, 0, u(\cdot)) \neq \hat{z}$; in other words, for any unsafe state $\hat{z}$ in T, there is an input such that no trajectory emanating from the initial set I arrives at $\hat{z}$ during the interval $[0, t]$—so far, so good. Unfortunately, this proof only applies once $\hat{z} \in T$ is selected; there is nothing in this proof to stop the chosen input from driving all those trajectories into some other part of T, thus rendering the system unsafe.

Finally, consider $\hat{z} \in S$. By (8), for all $u(\cdot) \in \mathbb{U}$ there exists $z \in I$ and $s \in [0, t]$ such that $\xi_{\mathsf{H}}(s; z, 0, u(\cdot)) = \hat{z}$; in other words, for all inputs there exists a trajectory starting from somewhere in I that will arrive at $\hat{z}$ at or before time t. However, this is not the safety question that we sought to answer. For all these $z \in I$, there may still exist some other $\hat{u}(\cdot) \in \mathbb{U}$ that ensures $\xi_{\mathsf{H}}(s; z, 0, \hat{u}(\cdot)) \notin T$ for all $s \in [0, t]$, and hence that H is safe.

The forward reach set can fail for safety verification in either of the ways that the forward reach tube or the backward reach set fails. $\qquad \square$

The essential problem with minimal forward reachability is that the state lying in the initial set is chosen after the input while the state lying in the target set is chosen before, rather than the other way around.

4.4 Examples of Forward and Backward Reachability for Safety

In this section we examine the various reachability constructs in terms of the purely continuous system H_2 for $x \in \mathbb{R}^2$.

$$\dot{x} = \begin{bmatrix} +1 \\ u \end{bmatrix}, \qquad \text{where } |u| \leq 1. \tag{13}$$

The motion of H_2 is easy to visualize: translation to the right at unit speed, and the choice of input determines vertical motion at unit speed.

We examine the situation for two initial and two target sets.

$$\begin{aligned} I_1 &= [0, +2] \times [0, +2] \quad T_1 = [+2, +4] \times [+7, +8] \\ I_2 &= [0, +2] \times [0, +4] \quad T_2 = [+2, +4] \times [+5, +8] \end{aligned}$$

This system can be used to demonstrate Proposition 3 on universal safety for all inputs $u(\cdot)$. Any combination involving I_2 or T_2 will be unsafe, which can be seen by choosing $u(t) = +1$ which drives the trajectory starting in the top left corner of the initial set to enter the target set at $t = 3$ (and exit at $t = 4$). The same
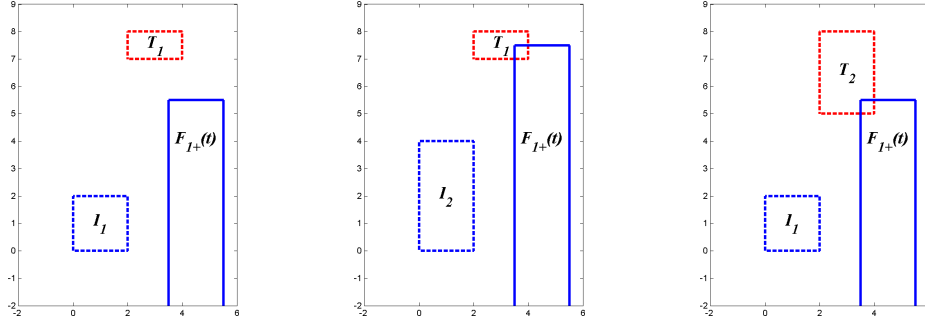
Fig. 1. Forward maximal reach sets $F_{1+}(\mathsf{H}_2, I_i, t)$ at $t = 3.5$ for various combinations of initial and target set. The system is safe for the leftmost case but not the other two, since the reach set intersects the target set in only the center and rightmost cases.
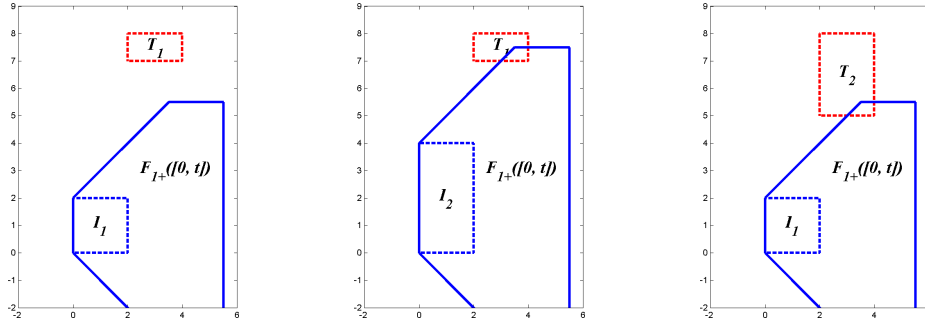

Fig. 2. Forward maximal reach tubes $F_{1+}(\mathsf{H}_2, I_i, [0, t])$ at $t = 3.5$ for various combinations of initial and target set.

input drives the upper left corner of I_1 closest to T_1, but no trajectory entering T_1 exists and for this lone case H_2 is safe. Figures 1–4 show the various maximal reach sets and tubes at $t = 3.5$ for three of the combinations of initial and target set (the case I_2 and T_2 is clearly unsafe and is not shown). The conclusion of all four types of reachability analysis concur, thus supporting Proposition 3.

For a demonstration of the negative result for the backward reach set in Proposition 4 on existential safety for some input $u(\cdot)$, we use H_1 from (11). Let $I_5 = [+1, +2]$ and $T_5 = [-1, 0]$. As shown in the proof of Proposition 2

$$B_{1-}(\mathsf{H}_1, T_5, t) = \begin{cases} [3t - 1, t], & t \leq 1/2; \\ \emptyset & \text{otherwise.} \end{cases}$$

Therefore $B_{1-}(\mathsf{H}_1, T_5, t) \cap I_5 = \emptyset$ for all t. However, $B_{1-}(\mathsf{H}_1, T_5, [0, t]) = [-1, t]$ and the tube component of Proposition 4 demonstrates that (H_1, I_5, T_5) is actually unsafe for all $t > 1$. A demonstration of the negative result for the forward reach set in Proposition 5 can be constructed by using $\dot{x} = b + 2$ as the dynamics and swapping the initial I_5 and unsafe T_5 sets in the discussion above.
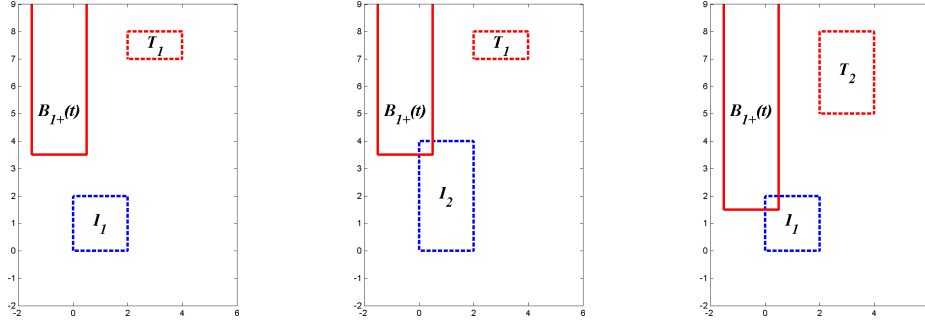
Fig. 3. Backward maximal reach sets $B_{1+}(\mathsf{H}_2, T_i, t)$ at $t = 3.5$ for various combinations of initial and target set.
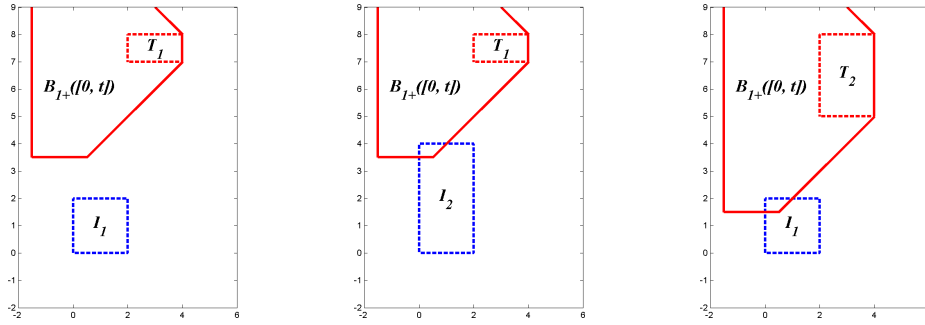

Fig. 4. Backward maximal reach tubes $B_{1+}(\mathsf{H}_2, T_i, [0, t])$ at $t = 3.5$ for various combinations of initial and target set.

For a demonstration of the tube components of Propositions 4 and 5 we return to H_2 from (13). Again, we choose two initial and two target sets.

$$I_3 = [0, +2] \times [-2, +2] \quad T_3 = [+5, +7] \times [-2, +2]$$
$$I_4 = [0, +2] \times [-4, +4] \quad T_4 = [+5, +7] \times [-4, +4]$$

These initial and target sets are horizontally aligned, so for any initial state with $x_2 \geq 0$, choose $u(t) = +1$ and for any initial state with $x_2 \leq 0$, choose $u(t) = -1$. With these input signals it is easy to see that either initial set with T_3 is safe, while either initial set with T_4 is unsafe.

Figures 5 and 6 show the two minimal reach tubes for three of the combinations of initial and target sets (the unsafe case I_4 and T_4 is not shown but is an easy extrapolation from those given). Both tubes reach a fixed point at $t = 2$ (for I_3 or T_3) or $t = 4$ (for I_4 or T_4), and it is that fixed point which is shown. The failure of the forward reach tube to correctly distinguish safe and unsafe situations can be seen in the two right subplots of Figure 5.
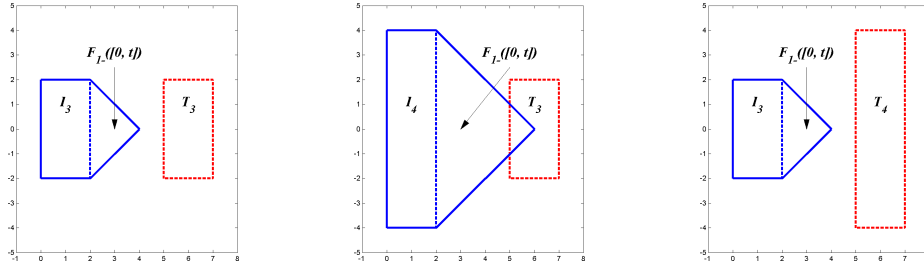
Fig. 5. Fixed points of the forward minimal reach tubes. The two cases on the left are actually safe, while the case on the right is unsafe. The forward reach tube demonstrates that it is inappropriate for existential safety verification in the two cases on the right.
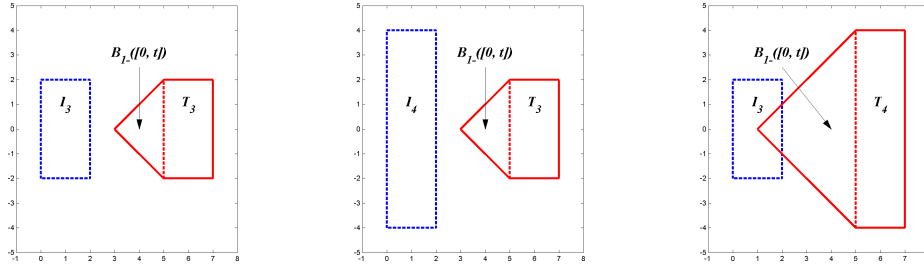
Fig. 6. Fixed points of the backward minimal reach tubes. Safety is correctly determined for the two cases on the left, and a lack of safety for the case on the right.

4.5 Time Dependent Systems

Throughout the paper we assume that the systems of interest are time independent,[3] but the definitions of reach sets and tubes in Sections 4.1 and 4.2 can be modified to accomodate systems whose dynamics depend on time. For forward reachability this process is straightforward and results in the set of states reached by trajectories starting from a given set at a given time. For backward reachability the result is the set of states that will reach a given target at a given time. While this set might be useful in other applications, safety specifications rarely include a time of failure; consequently time dependent systems are one area where forward reachability formulations may prove more useful. Of course, it is always possible to transform a time dependent system into a time independent system by adding a dimension to the dynamics. Furthermore, the results in the next section permitting a change in algorithmic temporal direction apply to time dependent systems, although additional care must be taken in handling the time variable.

---

[3] We will avoid the use of the term "autonomous" because of its conflicting definitions in the control and dynamic systems literature.

5 Exchanging Forward and Backward Reachability

Despite the negative conclusions regarding the minimal forward reach tube $F_{1-}(\mathsf{H}, S, [0, t])$, algorithms for its computation may still be useful if they can be used to compute backward reach tubes. In order to establish the situations under which forward and backward reachability may be interchanged, we must be able to reverse the direction of time in our dynamic system. Under Assumption 1, the following assumption will be relatively easily satisfied.

Assumption 2. *For a given dynamic system* H, *there exists a* backward *dynamic system* $\overleftarrow{\mathsf{H}}$ *such that for all* $t, s \in \mathbb{T}$

$$\xi_{\mathsf{H}}(s; z, t, u(\cdot)) = \hat{z} \qquad \Longleftrightarrow \qquad \xi_{\overleftarrow{\mathsf{H}}}(s; \hat{z}, t, u(\cdot)) = z.$$

Furthermore, $\xi_{\overleftarrow{\mathsf{H}}}$ *satisfies the conditions of Assumption 1.*

For the continuous $\mathsf{H_C}$, $\xi_{\overleftarrow{\mathsf{H_C}}}$ satisfies the ODE

$$\dot{z}(t) = \overleftarrow{f}(z(t), u(t)) \triangleq -f(z(t), u(t)) \tag{14}$$

and $\overleftarrow{\mathsf{H_C}} = (\mathbb{Z}, -f, U)$. If f satisfies the sufficient conditions mentioned above for $\xi_{\mathsf{H_C}}$ to satisfy Assumption 1, then so will $\xi_{\overleftarrow{\mathsf{H_C}}}$.

The case for the HA $\mathsf{H_H}$ is considerably more complex. In addition to the reversed continuous evolutions satisfying (14), there must exist reversed versions $\overleftarrow{G}$ and $\overleftarrow{r}$ of the guards and reset which satisfy

$$\begin{aligned} x \in G(q, \hat{q}, u_D) &\qquad \Longleftrightarrow \qquad \hat{x} \in \overleftarrow{G}(\hat{q}, q, u_D), \\ r(q, \hat{q}, x, u) = \hat{x} &\qquad \Longleftrightarrow \qquad \overleftarrow{r}(\hat{q}, q, \hat{x}, u) = x. \end{aligned} \tag{15}$$

With these definitions, $\overleftarrow{\mathsf{H_H}} = (\mathbb{Q}, \mathbb{X}, -f, D, \overleftarrow{G}, \overleftarrow{r}, U)$. Conditions under which $\xi_{\overleftarrow{\mathsf{H_H}}}$ would satisfy Assumption 1 are even more challenging to come by, although there has been some work [25]. However, if we can find a well posed $\overleftarrow{\mathsf{H}}$, then the temporal direction of our favourite reachability algorithm is irrelevant.

Proposition 6. *If* H *satisfies the conditions of Assumptions 1 and 2, then*

$$\begin{aligned} F_{1+}(\mathsf{H}, S, [0, t]) = B_{1+}(\overleftarrow{\mathsf{H}}, S, [0, t]) &\qquad F_{1+}(\mathsf{H}, S, t) = B_{1+}(\overleftarrow{\mathsf{H}}, S, t) \\ F_{1-}(\mathsf{H}, S, [0, t]) = B_{1-}(\overleftarrow{\mathsf{H}}, S, [0, t]) &\qquad F_{1-}(\mathsf{H}, S, t) = B_{1-}(\overleftarrow{\mathsf{H}}, S, t) \end{aligned}$$

Proof. We prove the claim for the minimal reach tubes; the proofs for the remaining claims are similar. Assume $\hat{z} \in F_{1-}(\mathsf{H}, S, [0, t])$. By (7), for all $u(\cdot) \in \mathbb{U}$ there exists $z \in S$ and $s \in [0, t]$ such that $\xi_{\mathsf{H}}(t, z, 0, u(\cdot)) = \hat{z}$. Under Assumption 2, $\xi_{\overleftarrow{\mathsf{H}}}(t; \hat{z}, 0, u(\cdot)) = z$. Because $\overleftarrow{\mathsf{H}}$ is time independent, we can shift the time variable to get $\xi_{\overleftarrow{\mathsf{H}}}(0; \hat{z}, -t, u(\cdot)) = z$, which by (9) implies $z \in B_{1-}(\overleftarrow{\mathsf{H}}, S, [0, t])$. The proof in the converse direction is similar. $\square$

6 Reachable Set Sensitivity

Section 4.3 demonstrated that the backward reach tube is the most generally applicable of the reachability operators to verification tasks. However, reach sets and tubes can rarely be determined analytically, so they must be approximated numerically. In this section we examine equations for the sensitivity of trajectories with respect to initial conditions. From these equations we can draw the conclusion that for some types of systems accurate numerical approximation of backwards reachability may not be possible.

The sensitivity analysis techniques used in this section force us to abandon the very general dynamic system definition used in the previous sections. Furthermore, we will assume that the number of states in discrete systems or the discrete component of hybrid systems is small enough that the discrete component of the reachable sets or tubes can be represented exactly. Therefore, we will focus our attention on continuous systems and the continuous component of hybrid systems. Since the former are a subset of the latter, we perform the analysis for hybrid systems and except where noted assume that $\mathsf{H} = \mathsf{H_H}$ and that Assumptions 1 and 2 hold.

For the purposes of this analysis the domains D and guards G are specified by implicit surface functions

$$D(q, u_D) = \{x \in \mathbb{X} \mid \psi_D(q, x, u_D) \leq 0\}$$
$$G(q, \hat{q}, u_D) = \{x \in \mathbb{X} \mid \psi_G(q, \hat{q}, x, u_D) \leq 0\}$$

for all $q, \hat{q} \in \mathbb{Q}$ and $u_D \in U_D$. The switching surfaces are then given by the zero level sets of these functions, and the normals of those switching surfaces by the local gradients. In order to study perturbations, we make the following assumption about the components of the hybrid system; the assumption also ensures that the switching surfaces and their normals are well defined.

Assumption 3. *The vector field f, reset r and implicit surface functions of the domains ψ_D and guards ψ_G are differentiable with respect to their continuous parameter x when all other parameters are held fixed.*

Sensitivity equations for a class of hybrid systems called differential-algebraic-discrete were derived in [2]. Here we adapt these results to HA of the form (2) by ignoring sensitivity with respect to parameter or discrete state, removing the algebraic component and adding a continuous reset. Details are omitted because the derivation follows directly from [2]. Sensitivity with respect to (constant) problem parameters can be derived in a similar manner. We do not consider sensitivity with respect to the input, and hence assume throughout that $u(\cdot) \in \mathbb{U}$ is fixed.

For convenience, define the matrices

$$\mathbf{F}(q, x, u) \triangleq \frac{\partial f(q, x, u)}{\partial x} \qquad \mathbf{R}(q, \hat{q}, x, u) \triangleq \frac{\partial r(q, \hat{q}, x, u)}{\partial x}$$

6.1 Trajectory Sensitivity Analysis

In this section we examine the effects on a trajectory's position due to small perturbations of the continuous portion of its initial state.

$$\xi_H(t; z_0 + \delta x, 0, u(\cdot)) = \xi_H(t; z_0, 0, u(\cdot)) + \Xi_H(t; \xi_H(\cdot))\delta x + \mathcal{O}(\delta x^2), \qquad (16)$$

where the initial state is $z_0 = (q_0, x_0)$, the perturbation is purely continuous $z_0 + \delta x = (q_0, x_0 + \delta x)$, $\xi_H(\cdot) = \xi_H(\cdot; z_0, 0, u(\cdot))$, and the *sensitivity matrix* is defined as

$$\Xi_H(t; \xi_H(\cdot)) \triangleq \frac{\partial \xi_H(t; z_0, 0, u(\cdot))}{\partial x_0}.$$

The continuous evolution of the HA is governed by an ODE, and sensitivity analysis of ODEs is well established; for example, see [26, section 4.6 and exercise 6.4]. Using what is essentially a Taylor series expansion, it can be shown that the sensitivity matrix solves the ODE

$$\frac{d}{dt}\Xi_H(t) = \mathbf{F}(q, x, u)\Xi_H(t), \qquad (17)$$

where $z = (q, x) = \xi_H(t; z_0, 0, u(\cdot))$ and $u = u(t)$. The initial condition for (17) is $\Xi_H(0) = \mathbf{I}$, where $\mathbf{I}$ is the identity matrix of appropriate size.

To treat the discrete jumps that occur in hybrid systems, let t^- and t^+ indicate values just before and just after the instantaneous jump respectively, $z^- = (q^-, x^-) = \xi_H(t^-; z_0, 0, u(\cdot))$ be the state just before the jump, and q^+ be the discrete state just after the jump (so $x^- \in G(q^-, q^+, u)$). For jumps that occur on switching surfaces the difference in post-jump state for two neighboring trajectories depends both on the reset and the difference in time when the jump is enabled (for guard switching surfaces) or forced (for domain switching surfaces). Let $t(z_0)$ be the time of the jump as a function of initial state and τ be its sensitivity. Then

$$\tau = \frac{\partial t(z_0)}{\partial z_0} = -\frac{\nabla\psi(x^-)^T \Xi_H(t^-)}{\nabla\psi(x^-)^T f(q^-, x^-, u)} \qquad (18)$$

where $\psi(x^-)$ is $\psi_D(q^-, x^-, u_D)$ for domain switching surfaces and $\psi_G(q^-, q^+, x^-, u_D)$ for guard switching surfaces. This equation is only valid if the vector field satisfies a transversality condition such that $\nabla\psi(x^-)^T f(q^-, x^-, u) \neq 0$ [2]. During this period, one trajectory is subject to the old vector field and one to the new vector field, so

$$\Xi_H(t^+) = \mathbf{R}(q^-, q^+, x^-, u)\left(\Xi_H(t^-) + f(q^-, x^-, u)\tau\right) - f(q^+, x^+, u)\tau, \qquad (19)$$

where $x^+ = r(q^-, q^+, x^-, u)$ and τ is given in (18). Away from switching surfaces trajectories in a neighborhood can all jump at the same time, so $\tau = 0$.

6.2 Implications for Approximating Reach Sets and Tubes

Given a nominal system trajectory ξ_H, the sensitivity evolution equations (17) and (19) can be solved as if they were a dynamic system to provide quantitative estimates of the form (16) for the effects of small perturbations on the initial conditions. Here, though, we will use them to ascertain conditions under which we cannot expect accurate results from approximate reachability algorithms. Most such algorithms use floating point instead of exact arithmetic, and hence make small errors throughout computation. Taking δx as a small numerical error incurred, for example, by a single floating point operation at time t and state z, algebraic manipulation of (16) arrives at a bound for the error at another time s (ignoring the $\mathcal{O}(\delta x^2)$ terms)

$$\|\xi_H(s; z + \delta x, t, u(\cdot)) - \xi_H(s; z, t, u(\cdot))\| \leq \|\Xi_H(s; \xi_H(\cdot))\|\|\delta x\|. \tag{20}$$

This trajectory-based sensitivity analysis is relevant to direct reachability algorithms because they either track trajectories explicitly (for Lagrangian approaches) or implicitly (for Eulerian); consequently, errors in locating a trajectory translate directly into errors in the approximation of the boundary of the reachable set or tube. It should be noted that Assumption 3 and the vector field transversality condition ensure that the two trajectories in (20) follow the same sequence of discrete states, so we need only consider the difference in their continuous states.

The multiplicative factor $\|\Xi_H(s; \xi_H(\cdot))\|$ in (20) depends on the trajectory $\xi_H(\cdot)$, but there are three ways in which it might grow large. The first (and only option for continuous systems) is continuous evolution by (17), whose explicit solution is given by $\Xi_H(t) = \exp[(t - s)\mathbf{F}]$. Large values of Ξ_H will result if $\mathbf{F}$ has eigenvalues with large positive real components. Discrete evolution by (19) contains the remaining two potential sources of growth. One is the term $\Xi_H(t^+) = \mathbf{R}\Xi_H(t^-)$, which will result in growth if $\mathbf{R}$ has any eigenvalues whose magnitudes are much greater than one. The final potential source of growth in (19) is the term τ given by (18), which may have large components if the denominator $\nabla\psi^T f^- \approx 0$ (we use f^- to denote $f(q^-, x^-, u)$). In summary, numerical ill-conditioning of a mathematical HA model of the form (2) can arise from one of three sources:

$$\text{Real}[\lambda(\mathbf{F})] \gg 0 \qquad \text{continuous evolution,} \tag{21}$$

$$|\lambda(\mathbf{R})| \gg 1 \qquad \text{discrete jumps,} \tag{22}$$

$$\nabla\psi^T f^- \approx 0 \qquad \text{grazing contact with switching surface,} \tag{23}$$

where $\lambda(\mathbf{A})$ are the eigenvalues of matrix $\mathbf{A}$ and $f^-(x) = f(q^-, x, u)$. Because $\mathbf{F}$, $\mathbf{R}$, ψ and f depend on state (and potentially input), checking these conditions explicitly will usually be impractical. However, systems satisfying any of the conditions (21)–(23) are inherently unpredictable; consequently, deterministic models of the form studied here are rarely constructed for such systems. With the notable exception of chaotic systems, conditions (21)–(23) are unlikely to occur in practice when computing forward reachability.

Unfortunately, the same cannot be said of backward reachability. It may be defined in terms of the forward dynamics, but computational approximations will begin with the target set and work backwards along trajectories of the time reversed system. Therefore, let us consider the form of conditions (21)–(23) for $\overleftarrow{\mathsf{H}}$ in terms of the elements of a given H. From (14),

$$\overleftarrow{f} = -f \implies \overleftarrow{\mathbf{F}} = -\mathbf{F} \implies \lambda(\overleftarrow{\mathbf{F}}) = -\lambda(\mathbf{F}).$$

From (15), $r(q, \hat{q}, \overleftarrow{r}(\hat{q}, q, x, u), u) = x$. Taking the derivative with respect to x

$$\mathbf{R}\overleftarrow{\mathbf{R}} = \mathbf{I} \implies \overleftarrow{\mathbf{R}} = \mathbf{R}^{-1} \implies \lambda(\overleftarrow{\mathbf{R}}) = \lambda(\mathbf{R})^{-1}.$$

The equivalent of (23) is a little more difficult to deduce, but as explained in [25] the concern is that the flow field after a forward time jump (before the reverse time jump) is nearly parallel to the switching surface which triggered the jump. To summarize, we restate conditions (21)–(23) for $\overleftarrow{\mathsf{H}}$ in terms of the parameters of H

$$\text{Real}[\lambda(\mathbf{F})] \ll 0 \qquad \text{backward continuous evolution,} \qquad (24)$$
$$|\lambda(\mathbf{R})| \ll 1 \qquad \text{backward discrete jumps,} \qquad (25)$$
$$\nabla \psi^T f^+ \approx 0 \qquad \text{backward grazing contact with switching surface,} \qquad (26)$$

where $f^+(x) = f(q^+, r(q^-, q^+, x, u), u)$ includes the action of the reset. As demonstrated in the next section, these conditions can easily occur for systems whose forward simulations are very well behaved. From these conditions, we draw the following conclusion about the challenges of using numerically approximated backwards reachability.

Remark 1. *Systems which display large amounts of contraction in forward time (ie nearby trajectories get closer together) in either their continuous evolution (of the form (24)) or discrete evolution (of the form (25)) are likely to be numerically ill-conditioned for backwards reachability. Poorly conditioned switching events (of the form (26)) are also more likely to be overlooked when working backward, because the relevant switching surfaces and vector fields are in different discrete modes.*

As a final comment, we note that this ill-conditioning of backwards reachability depends on the set being sought, and not the manner in which it is calculated. Consequently there are unlikely to be issues of ill-conditioning when using a backward algorithm and Proposition 6 to compute a forward reach set—this process involves reversing the dynamics twice and ends up back with forward dynamics. On the other hand, using a forward algorithm and Proposition 6 to determine the backward reach tube may run into ill-conditioning because the dynamics are reversed before the algorithm is applied.
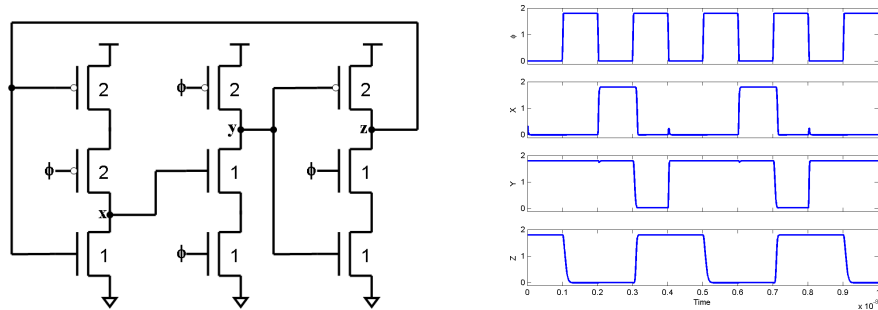
Fig. 7. Left: Yuan's and Svensson's toggle circuit [27]. The numbers next to the transistors are the relative sizing used in the simulations. Right: Simulation of the toggle model H_3 for a typical input signal ϕ.
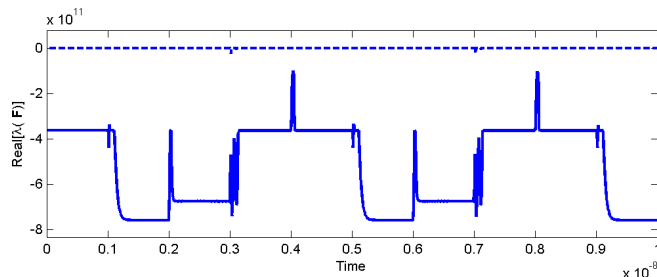

Fig. 8. Upper and lower bounds on the real components of the eigenvalues of the Jacobian $\mathbf{F}$ of the dynamics of H_3 during the simulation in Figure 7.

6.3 Continuous System Sensitivity Example

To illustrate how sensitivity of the continuous evolution can be a major issue in computing reachability for real systems, we examine the toggle circuit [27] whose schematic and typical trace are shown in Figure 7. The model H_3 is based on a simple, short channel transistor model with velocity saturation [28, pp. 62–63]. All capacitances are to ground and are of fixed value, and interconnect capacitance is ignored. To emulate the effect of connecting toggle elements together, the output node z is given an additional capacitive load equivalent to that seen by input ϕ.

The circuit is correctly operating if the period of the output z is twice the period of the input signal ϕ. Forward reachability has been used to demonstrate that under suitable constraints on the input, the output has twice the period of the input and satisfies the same constraints as the input; consequently, a chain of toggle circuits can be used to form a counter [29].

Unfortunately, a similarly successful analysis using backward reachability would be unlikely to succeed. Figure 8 shows the maximum and minimum real components of the eigenvalues of the Jacobian $\mathbf{F}$ of the dynamics for H_3 over the course of the simulation in Figure 7. Even after scaling by 10^{-8} to account for the
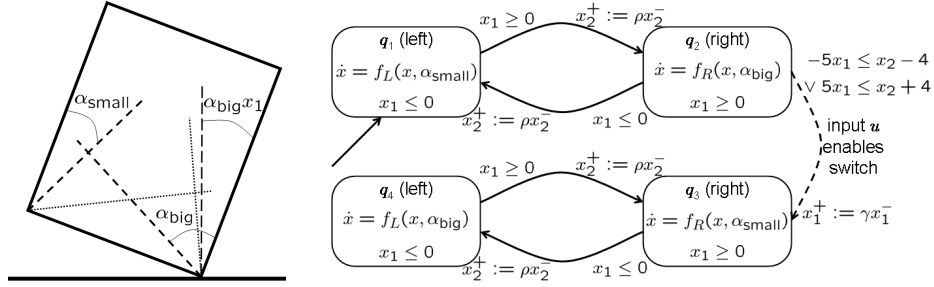
Fig. 9. Rocking block example. The left figure shows the configuration for mode q_2. The block starts with its center of mass at the crossing point of the two dashed lines (modes q_1 and q_2). The input u can enable the center of mass to switch sides to the crossing point of the two dotted lines (modes q_3 and q_4). The right figure shows the corresponding hybrid automaton H_4 with its discrete dynamics, domains, guards and resets. The continuous dynamics are given in (27).

very short time intervals typical of VLSI circuits, the minimum real component of the eigenvalues of $\mathbf{F}$ is $-(10^3)$ or less, which indicates a highly contractive dynamic system. Such systems are great for forward reachability calculations, since overapproximation errors will be rapidly contracted to the point of being negligible. But from (24) we see that backward reachability calculations are unlikely to maintain any accuracy for circuits of this type, since they face expansion factors of the same magnitude. In this case, error in backward reachability could grow by a factor of e^{1000} or more on time intervals as short as those in Figure 7.

6.4 Hybrid System Sensitivity Example

To examine the sensitivity issues that can arise in hybrid systems, we use a modified version of the rocking block example from [21]. Figure 9 shows H_4, where the continuous dynamics are given by

$$f_L(x, \alpha) = \begin{bmatrix} x_2 \\ \frac{1}{\alpha} \sin(\alpha(1 + x_1)) \end{bmatrix}, \qquad f_R(x, \alpha) = \begin{bmatrix} x_2 \\ -\frac{1}{\alpha} \sin(\alpha(1 - x_1)) \end{bmatrix}. \qquad (27)$$

The block in this case has an offset center of mass, which can reside in one of two symmetric locations. The system has a single discrete control u which can decide whether to activate switch (q_2, q_3) and thereby instantaneously move the center of mass; however, this switch has a guard and is only available when the block is not leaning too far to the right. Since the center of mass locations are symmetric, if the switch is activated the dynamics for the left and right modes swap. Because the block position x_1 is measured as a fraction of the angular position of the center of mass, it experiences a reset when (q_2, q_3) is activated as well. The parameters used in the analysis are

$$\alpha_{\text{big}} = \pi/3, \qquad \alpha_{\text{small}} = \pi/6, \qquad \rho = 0.8, \qquad \gamma = \alpha_{\text{big}}/\alpha_{\text{small}} = 2. \qquad (28)$$
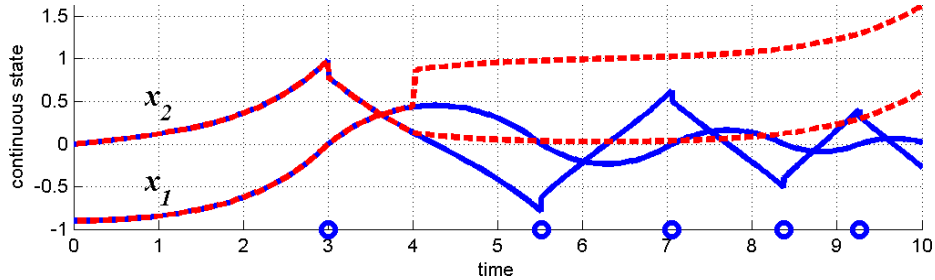
Fig. 10. Rocking block simulation with parameters given in (28), starting from $x = [-0.9 \ 0.0]^T$. The solid curve shows a trajectory in which the center of mass is not shifted, with resulting mode switches denoted by circles on the time axis. The block starts leaning to the left, shifts to the right at $t \approx 3.0$, back to the left at $t \approx 5.5$, and so on. The dashed curve shows a trajectory where the center of mass is shifted at $t = 4$ when $x_1^- \approx 0.43$ (well within the guard). Even though $x_1^+ \approx 0.86 < 1$, the block is still moving to the right after the switch. By $t = 8$, $x_1 > 1$ and it is clear that the block is toppling.

A simulation of trajectories with and without a center of mass shift is shown in Figure 10.

First, we consider sensitivity at the switching surface of the guard for (q_2, q_3). Figure 11 shows an enlarged view of the vector fields (arrows) before the switch $f^- = f(q_2, x)$ and after the switch $f^+ = f(q_3, r(q_2, q_3, x))$, and the switching surface (solid line) for the guard $G(q_2, q_3)$. The vector fields and switching surface are vertically symmetric, so we analyze the case for $x_2 > 0$.[4] The normal of the switching surface is $n = \begin{bmatrix} 5 & -1 \end{bmatrix}^T$. From the figure, it appears that the vector field before the switch will not cause problems via (23), but that the vector field afterwards may via (26). In fact, it is easy to show that $n^T f^- / \|n\| > 0.04$ (a small, but not tiny value), but that $n^T f^+ = 0$ for $x \approx \begin{bmatrix} 0.8255 & 0.1277 \end{bmatrix}^T \approx 1/23 \begin{bmatrix} 19 & 3 \end{bmatrix}^T$, where the decimal values are determined numerically and the fractions by the small angle approximation of sine. Therefore the backward trajectories of H_4 are infinitely sensitive at this switch. This sensitivity would be a problem if, for example, we sought from a set in q_3 a backward reach tube whose boundary passed close to this guard. Very small numerical errors could cause the boundary to incorrectly intersect or fail to intersect the guard, so that the backward reach tube would incorrectly contain or fail to contain states in q_2 respectively. This example highlights the reason why condition (26) is easy to miss: the switching surface examined above is reasonably well behaved with respect to the vector

---

[4] We point out but will not further discuss the fact that the switching surface's normal is not uniquely defined at $x_2 = 0$ and its implicit surface function would not be differentiable at this point. Such points in the state space are yet another situation where care must be taken when defining and computing reachable sets, but we do not yet have theory to analyze them.
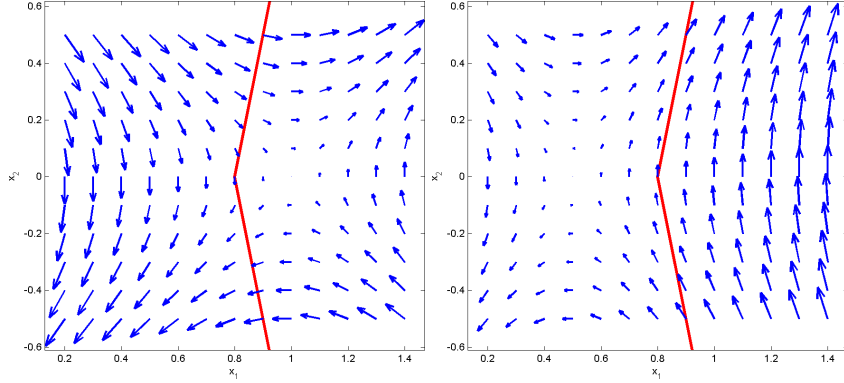
Fig. 11. Comparing the vector fields f^- and f^+ to the guard's switching surface for (q_2, q_3). Switching is allowed to the left of the switching surface. Only a portion of the continuous state space in mode q_2 is shown. While $f^- = f(q_2, x)$ in the left subplot is not parallel to the switching surface, $f^+ = f(q_3, r(q_2, q_3, x))$ in the right subplot is parallel slightly above and slightly below the $x_2 = 0$ line.

fields $f(q_2, x)$ and $f(q_3, x)$ of the modes on both sides of the switch; the problem only arises through the combination of vector field and reset.

As a final case of sensitivity, we examine the effect of the resets in H_4 when the block rocks from left to right or vice versa; for concreteness, consider the switch (q_1, q_2). This case corresponds to the conditions (22) and (25). From Figure 9, $r(q_1, q_2, x) = \begin{bmatrix} \rho x_1 & x_2 \end{bmatrix}^T$, so

$$\mathbf{R} = \begin{bmatrix} \rho & 0 \\ 0 & 1 \end{bmatrix} \qquad \overleftarrow{\mathbf{R}} = \mathbf{R}^{-1} = \begin{bmatrix} \rho^{-1} & 0 \\ 0 & 1 \end{bmatrix}.$$

Since $\rho \in [0, 1]$, $\|\mathbf{R}\| = 1$ and $\|\mathbf{R}^{-1}\| = \rho^{-1}$. Consequently, for $\rho = 0.8$ from (28) backward trajectories are only slightly more sensitive to numerical error than forward trajectories, and concern about the numerical stability of backward reachability would arise only after many, many discrete jumps. However, if the block were made out of a soft material, then it is likely that $\rho \ll 1$; for example, $\rho = 0.001$. Physically, this case would correspond to a block that for stable initial conditions would essentially settle in an upright position after rocking back and forth only one or two times. The sensitivity of forward trajectories remains the same, but errors in backward trajectories (specifically, errors in x_1) would be multiplied by $\rho^{-1} = 1000$ after each mode switch. While it might be appealing to compute the set of states which settle in an upright position through backwards reachability, the quantitative results could be dubious in this highly damped situation.

7 Conclusions and Future Research

Using a very general definition of dynamic system, we demonstrated that backward reach tubes are the most broadly applicable formulation of reachability for demonstrating system safety; that forward and backward algorithms can be interchanged if well-posed backward trajectories can be defined; and that the backward reachability formulation is more likely to suffer from numerical stability problems, particularly for systems displaying significant contraction. We intend to continue studying the sensitivity of reachability algorithms to problem parameters such as inputs, initial and target sets.

References

1. I. M. Mitchell, "Comparing forward and backward reachability as tools for safety analysis," in *Hybrid Systems: Computation and Control* (A. Bemporad, A. Bicchi, and G. Buttazzo, eds.), no. 4416 in Lecture Notes in Computer Science, pp. 428–443, Springer Verlag, 2007.
2. I. A. Hiskens and M. A. Pai, "Trajectory sensitivity analysis of hybrid systems," *IEEE Transactions on Circuits and Systems*, vol. 47, pp. 204–220, February 2000.
3. Z. Han and B. H. Krogh, "Reachability analysis of large-scale affine systems using low-dimensional polytopes," in *Hybrid Systems: Computation and Control* (J. Hespanha and A. Tiwari, eds.), no. 3927 in Lecture Notes in Computer Science, pp. 287–301, Springer Verlag, 2006.
4. T. A. Henzinger, P.-H. Ho, and H. Wong-Toi, "Algorithmic analysis of nonlinear hybrid systems," *IEEE Transactions on Automatic Control*, vol. 43, no. 4, pp. 540–554, 1998.
5. M. Greenstreet and I. Mitchell, "Reachability analysis using polygonal projections," in *Hybrid Systems: Computation and Control* (F. Vaandrager and J. van Schuppen, eds.), no. 1569 in Lecture Notes in Computer Science, pp. 103–116, Springer Verlag, 1999.
6. A. Bemporad, F. D. Torrisi, and M. Morari, "Optimization-based verification and stability characterization of piecewise affine and hybrid systems," in *Hybrid Systems: Computation and Control* (B. Krogh and N. Lynch, eds.), no. 1790 in Lecture Notes in Computer Science, pp. 45–59, Springer Verlag, 2000.
7. E. Asarin, T. Dang, and A. Girard, "Reachability analysis of nonlinear systems using conservative approximation," in *Hybrid Systems: Computation and Control* (O. Maler and A. Pnueli, eds.), no. 2623 in Lecture Notes in Computer Science, pp. 20–35, Springer Verlag, 2003.
8. A. Girard, C. L. Guernic, and O. Maler, "Efficient computation of reachable sets of linear time-invariant systems with inputs," in *Hybrid Systems: Computation and Control* (J. Hespanha and A. Tiwari, eds.), no. 3927 in Lecture Notes in Computer Science, pp. 257–271, Springer Verlag, 2006.
9. A. B. Kurzhanski and P. Varaiya, "Reachability analysis for uncertain systems—the ellipsoidal technique," *Dynamics of Continuous, Discrete and Impulsive Systems Series B: Applications and Algorithms*, vol. 9, no. 3, pp. 347–367, 2002.

10. M. Broucke, M. D. D. Benedetto, S. D. Gennaro, and A. Sangiovanni-Vincentelli, "Optimal control using bisimulations: Implementation," in *Hybrid Systems: Computation and Control* (M. D. D. Benedetto and A. Sangiovanni-Vincentelli, eds.), no. 2034 in Lecture Notes in Computer Science, pp. 175–188, Springer Verlag, 2001.

11. P. Saint-Pierre, "Hybrid kernels and capture basins for impulse constrained systems," in *Hybrid Systems: Computation and Control* (C. J. Tomlin and M. R. Greenstreet, eds.), no. 2289 in Lecture Notes in Computer Science, pp. 378–392, Springer Verlag, 2002.

12. J. A. Sethian and A. Vladimirsky, "Ordered upwind methods for hybrid control," in *Hybrid Systems: Computation and Control* (C. J. Tomlin and M. R. Greenstreet, eds.), no. 2289 in Lecture Notes in Computer Science, pp. 393–406, Springer Verlag, 2002.

13. I. M. Mitchell, A. M. Bayen, and C. J. 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.

14. Y. Gao, J. Lygeros, and M. Quincampoix, "The reachability problem for uncertain hybrid systems revisited: The viability theory perspective," in *Hybrid Systems: Computation and Control* (J. Hespanha and A. Tiwari, eds.), no. 3927 in Lecture Notes in Computer Science, pp. 242–256, Springer Verlag, 2006.

15. A. Tiwari and G. Khanna, "Series of abstractions for hybrid automata," in *Hybrid Systems: Computation and Control* (C. J. Tomlin and M. R. Greenstreet, eds.), no. 2289 in Lecture Notes in Computer Science, pp. 465–478, Springer Verlag, 2002.

16. M. Kloetzer and C. Belta, "Reachability analysis of multi-affine systems," in *Hybrid Systems: Computation and Control* (J. Hespanha and A. Tiwari, eds.), no. 3927 in Lecture Notes in Computer Science, pp. 348–362, Springer Verlag, 2006.

17. M. Johansson and A. Rantzer, "Computation of piecewise quadratic Lyapunov functions for hybrid systems," *IEEE Transactions on Automatic Control*, vol. 43, no. 4, pp. 555–559, 1998.

18. S. Prajna and A. Jadbabaie, "Safety verification of hybrid systems using barrier certificates," in *Hybrid Systems: Computation and Control* (R. Alur and G. J. Pappas, eds.), no. 2993 in Lecture Notes in Computer Science, pp. 477–492, Springer Verlag, 2004.

19. C. Stangier and T. Sidle, "Invariant checking combining forward and backward traversal," in *Formal Methods in Computer-Aided Design* (A. J. Hu and A. K. Martin, eds.), no. 3312 in Lecture Notes in Computer Science, pp. 414–429, Springer Verlag, 2004.

20. M. Broucke and A. Arapostathis, "Continuous selections of trajectories of hybrid systems," *Systems and Control Letters*, vol. 47, pp. 149–157, 2002.

21. J. Lygeros, K. H. Johansson, S. N. Simic, J. Zhang, and S. Sastry, "Dynamical properties of hybrid automata," *IEEE Transactions on Automatic Control*, vol. 48, pp. 2–17, January 2003.

22. L. C. Evans and P. E. Souganidis, "Differential games and representation formulas for solutions of Hamilton-Jacobi-Isaacs equations," *Indiana University Mathematics Journal*, vol. 33, no. 5, pp. 773–797, 1984.

23. A. B. Kurzhanski and P. Varaiya, "On reachability under uncertainty," *SIAM Journal of Control and Optimization*, vol. 41, no. 1, pp. 181–216, 2002.

24. I. M. Mitchell, "Comparing forward and backward reachability as tools for safety analysis," Tech. Rep. TR-2006-23, Department of Computer Science, University of British Columbia, Vancouver, BC, Canada, December 2006.

25. I. A. Hiskens, "Non-uniqueness in reverse time of hybrid system trajectories," in *Hybrid Systems: Computation and Control* (M. Morari and L. Thiele, eds.), no. 3414 in Lecture Notes in Computer Science, pp. 339–353, Springer Verlag, 2005.

26. U. M. Ascher and L. R. Petzold, *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations.* Philadelphia: Society for Industrial and Applied Mathematics, 1998.

27. J. Yuan and C. Svensson, "High-speed CMOS circuit technique," *IEEE Journal of Solid-State Circuits*, vol. 24, pp. 62–70, February 1989.

28. D. A. Hodges, H. G. Jackson, and R. A. Saleh, *Analysis and Design of Digital Integrated Circuits in Deep Submicron Technology.* New York: McGraw Hill, third ed., 2004.

29. M. R. Greenstreet, "Verifying safety properties of differential equations," in *Computer Aided Verification* (R. Alur and T. A. Henzinger, eds.), no. 1102 in Lecture Notes in Computer Science, pp. 277–287, Springer Verlag, 1996.