



REPORT *RAPPORT*

MAS

Modelling, Analysis and Simulation



Modelling, Analysis and Simulation

Observability of Hybrid Systems and Turing Machines

P.J. Collins, J.H. van Schuppen

REPORT MAS-E0413 AUGUST 2004

CWI is the National Research Institute for Mathematics and Computer Science. It is sponsored by the Netherlands Organization for Scientific Research (NWO).

CWI is a founding member of ERCIM, the European Research Consortium for Informatics and Mathematics.

CWI's research has a theme-oriented structure and is grouped into four clusters. Listed below are the names of the clusters and in parentheses their acronyms.

Probability, Networks and Algorithms (PNA)

Software Engineering (SEN)

Modelling, Analysis and Simulation (MAS)

Information Systems (INS)

Copyright © 2004, Stichting Centrum voor Wiskunde en Informatica

P.O. Box 94079, 1090 GB Amsterdam (NL)

Kruislaan 413, 1098 SJ Amsterdam (NL)

Telephone +31 20 592 9333

Telefax +31 20 592 4199

ISSN 1386-3703

Observability of Hybrid Systems and Turing Machines

ABSTRACT

In this paper we discuss the observability of hybrid systems and Turing machines. We give an elementary example to show that observability is undecidable for Turing machines with output. Since many classes of hybrid systems simulate Turing machines, this immediately shows these classes are undecidable. We discuss the observability of piecewise-affine hybrid systems, and give a number of examples illustrating different observability properties.

2000 Mathematics Subject Classification: 93B07

Keywords and Phrases: Observability; hybrid systems; Turing machines

Note: The authors gratefully acknowledge the financial support of the European Commission through the project Control and Computation (IST-2001-33520) of the Program Information Societies and Technologies.

Observability of Hybrid Systems and Turing Machines

Pieter Collins and Jan H. van Schuppen

Abstract—In this paper we discuss the observability of hybrid systems and Turing machines. We give an elementary example to show that observability is undecidable for Turing machines with output. Since many classes of system simulate Turing machines, we can then show that observability for these classes is undecidable. We discuss the observability of piecewise-affine hybrid systems, and give examples illustrating different observability properties.

I. INTRODUCTION

In linear control theory, the basic system properties of stability, controllability and observability are straightforward to check. Beyond linear systems, a number of undecidability results for various classes of nonlinear systems exist. Most existing undecidability results concern the *reachability relation*, that is, given two points x_0 and x_1 , is there a system trajectory from x_0 and x_1 . For Turing machines [1], the reachability relation is known to be undecidable, since it is simply a reformulation of the halting problem. A two-stack pushdown automaton can trivially simulate a Turing machine, and a two-counter machine also has equivalent computational power. A reversible Turing machine was shown to be computationally universal by Bennett [2], and a reversible two-counter machine was shown to be universal by Morita [3]. The reachability relation is undecidable for all these machines.

As well as the halting problem, related problems are also undecidable, such as the *mortality problem*, under which a machine halts starting from any configuration (state and input). The mortality problem for Turing machines with an infinite input tape was shown to be undecidable by Hooper [4]. An elementary proof that the mortality problem for two-counter machines is undecidable was given by Blondel et al. [5].

The study of computability properties for continuous-space systems requires the reduction of problems either to a Turing machine, or to a model with similar computational power. Systems known to be able to simulate Turing machines include planar diffeomorphisms [6], planar piecewise-linear maps [7], piecewise-constant derivative systems [8] and high-dimensional saturated linear systems [9]. (It was shown that a saturated linear system with 866 variables can simulate a universal Turing machine.)

In the literature, most attention has been directed towards the system properties of stability and controllability. Results for saturated linear systems were obtained by Siegelmann and Sontag [9], [10] and a number of additional results have been obtained by Blondel, Tsitsiklis et al. [11], [12].

In [10], distinguishability of a state from zero was related to reachability analysis, and hence shown to be undecidable. Existing work on observability has focused on systems with saturated outputs, which arise naturally in many applications including neural networks. Observability for continuous-time linear systems with saturated outputs holds if and only if the Kalman rank condition plus an additional property hold [13], but the computability of this property was not discussed.

In this paper we study the observability problem for systems on a compact state space, with the evolution and output defined by continuous functions, since this is the case of most interest. We consider the cases that the set of possible initial states is either finite, or is the entire state space. We show that these observability problems are undecidable for the class of rational piecewise-affine systems. These systems are *effective* (i.e. the evolution of rational points can be exactly computed), have a *hybrid-space* character (i.e. the system evolution law depends on the position in phase space), and include the saturated linear systems. Unlike the restricted case of saturated linear systems, we can show that the observability problem for general piecewise-affine systems is undecidable even in dimension two.

The observability problem for a finite initial state set reduces to the halting problem by considering an appropriate observation function. When the initial state set is the entire state space, we reduce to the mortality problem. Since we consider systems on a compact state space defined by continuous maps, it is of crucial importance that we reduce to a class of machine with a compact configuration space. This includes Turing machines with an infinite input tape, but not Turing machines with a finite input or two-counter machines.

We first show that two appropriately-formulated observability problems for Turing machines are undecidable. Although we do not use these results directly in the sequel, they are of independent interest, and the proofs illustrate some of the techniques used when considering piecewise-affine systems.

Finally, we discuss the *distinguishability time* for two trajectories. For observable linear systems, all states can be distinguished in finite time. For observable hybrid systems, we may be able to distinguish two states in finite time, but may need to look at the trajectories on infinite time intervals.

A. Observability notions

We consider observability for systems without inputs.

Definition 1 (Distinguishability and Observability):

Two initial states x and x' of a system \mathcal{S} are *distinguishable*

if the outputs η and η' for the trajectories starting at x and x' are different. A system \mathcal{S} is *observable* if the output map $x \rightarrow \eta$, which gives the output for the trajectory starting at x , is injective.

For deterministic systems, observability amounts to determining the initial state. There is a corresponding notion of *reconstructibility*, which states that for any initial states x and x' there exists T such that if the corresponding observations η and η' are equal on the interval $[0, T]$, then $\xi(T) = \xi'(T)$, where ξ and ξ' are the trajectories starting at x and x' .

For systems with inputs, there are a number of different observability notions [14], corresponding to the system being observable for some predetermined input, for every input, or for an input given as a feedback based on the observations up to the current time.

II. OBSERVABILITY OF TURING MACHINES

We use a standard definition of Turing machine [1].

Definition 2 (Turing machine with output): Let Σ be a finite *input alphabet* and Υ a finite *output alphabet*.

A *Turing machine with output* \mathcal{M} over (Σ, Υ) is a tuple $\{Q, Q_{\text{init}}, Q_{\text{halt}}, \Gamma, \Lambda\}$ where Q is a finite *state set*, Q_{init} is the set of *initial states* and Q_{halt} is the set of *halting states*. The *transition function* Γ and *observation function* Λ are

$$\Gamma: Q \times \Sigma \rightarrow Q \times \Sigma \times \Delta, \quad \Lambda: Q \rightarrow \Upsilon \cup \{\varepsilon\},$$

where $\Delta = \{-1, 0, +1\}$.

A *configuration* of a Turing machine \mathcal{M} is an element of (q, w) of $Q \times \Sigma^{\mathbb{Z}}$. The element $w \in \Sigma^{\mathbb{Z}}$ is the *tape contents*.

The *successor function* τ of \mathcal{M} is a function $\tau: Q \times \Sigma^{\mathbb{Z}} \rightarrow Q \times \Sigma^{\mathbb{Z}}$ defined as follows. If the state is q , and the element w_0 of the tape is equal to a , then the transition $(q, a) \mapsto (q', a', \delta)$ is applied. The element w_0 is replaced by a' , the tape is shifted by the map σ^δ , where $(\sigma(w))_i = w_{i+1}$, and the new state is q' .

We say that a Turing machine has a *finite* tape if there is a special *blank symbol* $\sqcup \in \Sigma$ such that every symbol is blank except for a finite contiguous block of non-blank symbols, otherwise it has an *infinite* tape.

Definition 3: A (Turing) machine \mathcal{M} is *mortal* if it halts on every configuration, and *nilpotent* if there exists an integer N such that \mathcal{M} halts on every configuration after at most N steps.

The mortality problem for Turing machines with an infinite tape was shown to be undecidable by Hooper [4]. It is straightforward to show [12] that a Turing machine with infinite tape is mortal if and only if it is nilpotent. The mortality problem for Turing machines with a finite tape is a distinct problem; a Turing machine which moves right until it finds the first blank symbol on the tape is always mortal for a finite tape but not for an infinite tape, since the latter need not contain any blanks. In general, results on the mortality problem for one class of systems cannot be used to derive results on the mortality problem for other classes, even if each class can simulate the other.

For applications to dynamical systems, we will require mortality for Turing machines with an infinite tape. The decidability of the mortality problem for Turing machines with a finite tape is not known to the authors.

A. Tape observability

Consider the following *tape observability problem* for a Turing machine \mathcal{M} with a single initial state $q_{\text{init}} \in Q_{\text{init}}$. The observability problem is to determine the initial state of the tape from the output. We assume that the initial tape is finite, but can take any value. It is clear that there is no upper bound on the time needed to distinguish two initial tapes, since an output of length n can only distinguish between outputs of length n .

The following result reduces the tape observability problem to the mortality problem for Turing machines on a finite tape.

Theorem 4: Tape observability for Turing machines is undecidable if the mortality problem for a finite tape is undecidable.

Proof: Let \mathcal{M} be a Turing machine with alphabet Σ which has at least two non-blank symbols. Construct a Turing machine with output \mathcal{M}' as follows: \mathcal{M}' first makes a copy of its input. It then modifies this copy by replacing the first non-blank symbol with a blank. It finally runs \mathcal{M} on the modified copy of the input without disturbing the original input. If \mathcal{M} halts, then \mathcal{M}' outputs its original input. Otherwise, \mathcal{M}' outputs nothing.

Any input for \mathcal{M} can be obtained as the modified version of the input for \mathcal{M}' . Hence if \mathcal{M} is mortal, then \mathcal{M}' is mortal, and hence tape observable. If \mathcal{M} is immortal, then there is an input on which \mathcal{M} does not halt. There are at least two inputs for \mathcal{M}' which give this input for \mathcal{M} after blanking the first symbol. On these inputs, \mathcal{M}' does not halt, and hence these inputs are indistinguishable. ■

Note that the construction does not work for Turing machines with an infinite tape, since there is no way to effectively copy the initial tape.

B. State observability

Instead of considering the initial tape as the unknown configuration, we can instead consider the initial tape as a known parameter describing the system, and consider observability of the initial state. A Turing machine is *state observable* if the initial state $q_{\text{init}} \in Q_{\text{init}}$ can be determined from the initial tape contents and the output.

Theorem 5: State observability is undecidable for Turing machines with an infinite tape.

Proof: Let \mathcal{M} be a Turing machine with a single initial state and a single halting state. Let $\Lambda(q) = \varepsilon$ except for the halting state q_{halt} for which $\Lambda(q_{\text{halt}}) = y$. Let Q' be a set disjoint from Q , and $i: Q \rightarrow Q'$ a bijection. We let \mathcal{M}' be a copy of \mathcal{M} with halting state $q'_{\text{halt}} = i(q_{\text{halt}}) \in Q'$ for which $\Lambda(q'_{\text{halt}}) = y'$. The transition function Γ' of \mathcal{M}' is given by $\Gamma'(i(q), a) = (i(\tilde{q}), \tilde{a}, \delta)$ if $\Gamma(q, a) = (\tilde{q}, \tilde{a}, \delta)$.

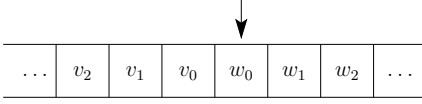


Fig. 1. Tape contents of a Turing machine.

Consider the *disjoint union* $\mathcal{M} \sqcup \mathcal{M}'$ of \mathcal{M} and \mathcal{M}' whose states are $Q \sqcup Q'$, initial states are $Q_{\text{init}} \sqcup Q'_{\text{init}}$, and halting states are $Q_{\text{halt}} \sqcup Q'_{\text{halt}}$. The transitions are given by $\Gamma \sqcup \Gamma'(q, a) = \Gamma(q, a)$ if $q \in Q$, and $\Gamma \sqcup \Gamma'(q', a) = \Gamma'(q', a)$ if $q' \in Q'$. An observation of a state of $\mathcal{M} \sqcup \mathcal{M}'$ is empty, except for one of the two halting states q_{halt} and q'_{halt} , in which case $\mathcal{M} \sqcup \mathcal{M}'$ reveals whether its state is in the original machine or the copy. If \mathcal{M} halts on an input w , then so does $\mathcal{M} \sqcup \mathcal{M}'$, and the halting state reveals the initial state. If \mathcal{M} does not halt, neither does $\mathcal{M} \sqcup \mathcal{M}'$, and no output is observed. Therefore, it is impossible to decide in general whether the initial state is q_{init} or q'_{init} . ■

In the construction, the halting states of $\mathcal{M} \sqcup \mathcal{M}'$ reveal aspects of the initial configuration which are not observed in other configurations. We therefore refer to halting states as *revealing states* for the machine with output.

III. SIMULATION OF TURING MACHINES BY PIECEWISE-AFFINE SYSTEMS

The standard simulation of a Turing machine relies on an encoding of the tape contents as a pair of numbers, which are rational for a finite tape.

A. Encoding of Turing machines

We now show how to simulate a Turing machine using a piecewise-affine map. This map will first be a partial function on a number of cells; later we can glue these together in an appropriate way. Without loss of generality, we restrict to Turing machines with a transition function $\Gamma : Q \times \Sigma \rightarrow Q \times \Sigma \times \Delta$ such that for every state q of the Turing machine, either $\Gamma(q, a) = (q', a, \delta)$ where q' does not depend on a , or $\Gamma(q, a) = (q', a', 0)$. The former move simply shifts the tape without reading or writing, whereas the latter changes the tape without shifting.

We can write the configuration of a Turing machine as a triple $(q, w, v) \in C = Q \times \Sigma^\omega \times \Sigma^\omega = \mathcal{C}$. The pair $(w, v) \in \Sigma^\omega \times \Sigma^\omega$ represents the tape contents $\dots, v_{-2}, v_{-1}, v_0 \cdot w_0, w_1, w_2, \dots$, with the tape head pointing to w_0 . Finite tapes can be represented by pairs of finite words $(w, v) \in \Sigma^* \times \Sigma^*$ with no trailing blanks. Note that Σ^* can be embedded in Σ^ω by padding with infinitely many blank symbols.

There is a natural topology on Σ^ω given by the metric $d(w, w') = 1/2^k$, where $w_i = w'_i$ for $i < k$ but $w_k \neq w'_k$. In this topology Σ^ω is compact, but Σ^* is not a closed subset of Σ^ω .

We encode the configuration of a Turing machine as a subset of $Q \times [0, 1] \times [0, 1]$ as follows. We assume Σ has

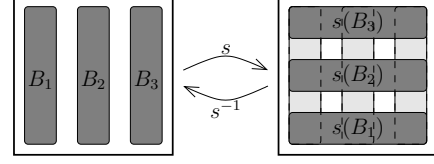


Fig. 2. The image of boxes under the shift map

elements with integer numbers between 1 and N , and encode a sequence $w \in \Sigma^\omega$ by

$$\theta(w) = \sum_{i=0}^{\infty} w_i / r^{i+1},$$

where $r > N + 1$ (say $r = N + 2$). The encoding of a state of a Turing machine is given by $v(q, w, v) = (q, \theta(w), \theta(v))$.

The function θ is continuous and injective. Its image $\theta(\Sigma^\omega)$ is a Cantor set contained in the interval $(0, 1)$. If all elements of w are eventually equal to some constant, then $\theta(w) \in \mathbb{Q}$, so whatever value is chosen to represent the blank symbol $_$, the encoding of a word $w \in \Sigma^*$ is always a rational. We define the *cell* $C(q) = \{q\} \times [0, 1] \times [0, 1]$, and the *box* $B(q, a)$ to be the convex hull of $v(q, aw, v)$ in $C(q)$.

To encode a Turing machine, we also need to simulate its transition by a map f . If $\Gamma(q, a) = (q', a', 0)$, then we can take $f : B(q, a) \rightarrow B(q', a')$ by

$$f(q, x_1, x_2) = (q', x_1 + (a' - a)/r, x_2),$$

which simply moves around boxes.

A shift of the tape can be simulated by the *shift map* s on $[0, 1] \times [0, 1]$ given by

$$s(x_1, x_2) = (rx_1 - \lfloor rx_1 \rfloor, (\lfloor rx_1 \rfloor + x_2)/r). \quad (1)$$

The map s is affine on each of the boxes B . If $\Gamma(q, a) = (q', a, 1)$, then we take $f : B(q, a) \rightarrow C(q')$ by

$$f(q, x_1, x_2) = (q', rx_1 - \lfloor rx_1 \rfloor, (\lfloor rx_1 \rfloor + x_2)/r) = (q', s(x_1, x_2))$$

The reverse-shift s^{-1} can also be given as a piecewise-affine map. Overall, the transitions are related by the commutation relation $f \circ v = v \circ \tau$.

B. Rational piecewise-affine maps

Our goal is to represent Turing machines by continuous rational piecewise-affine functions, since computations may be performed effectively on such functions, and continuity is a natural condition to require for functions on continuous spaces. Further, discontinuities are a well-known source of uncomputability; a discontinuous real-valued function is uncomputable using interval arithmetic [15]. For an overview of computability questions in analysis, see [16].

Definition 6 (Rational piecewise-affine maps): A continuous function $f : X \rightarrow Y$ is a *rational piecewise-affine map* if X is the union of finitely many sets X_i such that each X_i is given by a matrix inequality of the form $B_i x \leq b_i$, and $f : X_i \rightarrow Y$ is given by $f(x) = A_i x + a_i$. Here, the a_i

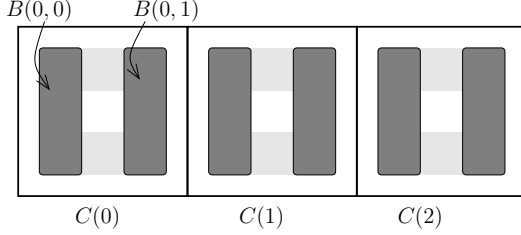


Fig. 3. State space of a piecewise-affine map simulating a Turing machine.

and b_i are vectors with rational coefficients, and the A_i and B_i are matrices with rational coefficients.

The following results show how to extend the simulating partial functions to total functions.

Theorem 7: Let X be a contractible (e.g. convex) polytope.

- 1) If $f : A \rightarrow X$ is a rational piecewise-affine map defined on $A \subset X$, then f extends to a rational piecewise-affine map defined on X .
- 2) If in addition f is defined on a finite disjoint union of topological discs which are disjoint from the boundary and is injective, then f can be extended to a rational piecewise-affine homeomorphism.

Proof: [Sketch] To prove Pt. 1, we simply triangulate the complement of A , and extend f in a piecewise-linear way by giving its values on the vertices of the simplices. To prove Pt. 2, we see that since f is isotopic to the identity on A , so by the isotopy extension theorem (see [17] Thm. 8.1.3) f extends to a homeomorphism on X . This can be triangulated by taking small enough perturbations. (Alternatively, we can construct the original isotopy by a suitable piecewise-constant derivative vector field, and extend within the class of piecewise-constant vector fields.) ■

In particular, any Turing machine can be simulated by a rational piecewise-affine map, and any reversible Turing machine by a rational piecewise-affine homeomorphism.

The following result shows that any orbit which remains in the boxes $B(Q, \Sigma)$ for all time is shadowed by the image of some point in the Cantor set $v(Q, \Sigma^\omega, \Sigma^\omega)$.

Lemma 8 (Shadowing lemma): Suppose $x \in X$ is such that there exist sequences (q_n) and (a_n) such that $f^n(x) \in B(q_n, a_n)$ for all n . Then there exists a configuration $(q, w, v) \in \mathcal{C}$ such that $f^n(v(q, w, v)) \in B(q_n, a_n)$ for all n .

Proof: For each m , let R_m be the set of points y such that $f^n(y) \in B(q_n, a_n)$ for all $n \leq m$. Then R_m is a rectangle, since R_0 is a rectangle, and f maps rectangles in some $B(q, a)$ to rectangles. We claim that the vertices of R_m are points of $v(\mathcal{C})$ for all m . This is true for $m = 0$, since $R_0 = B(q_0, a_0)$ and the vertices of R_0 are points of $v(\mathcal{C})$. For any m we have $f^m(R_m) = f^m(R_{m-1}) \cap B(q_m, a_m)$, so if the vertices of $f^m(R_{m-1})$ are points of $v(\mathcal{C})$, then so are those of $f^m(R_m)$. The claim follows by induction since $v(\mathcal{C})$ is invariant under f , and f is locally injective on each $B(q, a)$. Hence each R_m is a nonempty compact set containing a

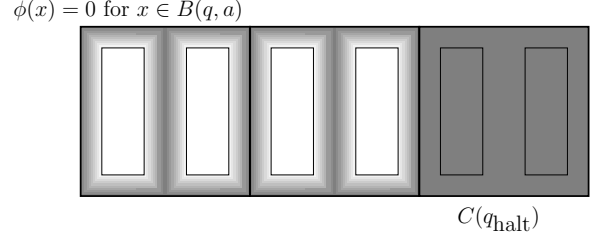


Fig. 4. A function $\phi(x)$ which is positive except on boxes $B(q, a)$ for $q \notin Q_{\text{halt}}$.

point of $v(\mathcal{C})$, and so $R_m \cap v(\mathcal{C})$ is a decreasing sequence of nonempty compact sets. Therefore there is a point in $\bigcap_{n=0}^{\infty} R_n \cap v(\mathcal{C})$, and this point is the required image of (q, w, v) . ■

IV. OBSERVABILITY OF PIECEWISE-AFFINE SYSTEMS

We wish to consider continuous systems consisting of continuous functions with continuous output. We also wish to consider various kinds of sets of initial conditions. In particular, we would like to take *all* possible initial conditions, or initial conditions in some compact set. We define output functions of a similar form to

$$h(x) = (\phi(x), x_1, \dots, x_k, x_{k+1}\phi(x), \dots, x_n\phi(x))$$

where $\phi : X \rightarrow \mathbb{R}$ and $x \in \mathbb{R}^n$, so that the i th coordinate of x is observable if and only if $i \leq k$ or $\phi(x) \neq 0$. An example of the function ϕ is given in Fig. 4. We often take $\phi(x) = 0$ on the boxes $B(q, a)$ for $q \notin Q_{\text{halt}}$, and $\phi(x) > 0$ otherwise.

A. Continuous rational piecewise-affine systems

Definition 9 (Continuous piecewise-affine systems): A pair (f, h) is a continuous rational piecewise-affine system if $f : X \rightarrow X$ and $h : X \rightarrow Y$ are rational piecewise-affine maps.

Theorem 10: Observability is undecidable for continuous rational piecewise-affine systems with finitely many possible initial states.

Proof: Let f simulate a pair of identical Turing machines $\mathcal{M}, \mathcal{M}'$ as in the proof of Thm. 5. Let $X_{\text{init}} = \{x_0, x'_0\}$ be a set consisting of two initial states, corresponding to the initial states of each Turing machine. Consider an output $h : X \rightarrow \mathbb{R}$ where $h(x) > 0$ if $x \in B(q_{\text{halt}}, a)$, $h(x) < 0$ for $x \in B(q'_{\text{halt}}, a)$, and $h(x) = 0$ if $x \in C(q)$ if $q \notin \{q_{\text{halt}}, q'_{\text{halt}}\}$. Then deciding observability of (f, h) is equivalent to solving the halting problem for \mathcal{M} . If \mathcal{M} halts, then x_0 reaches $B(q, a)$ for $q \in Q_{\text{halt}}$ giving positive output, and x'_0 reaches $B(q', a)$ for $q' \in Q'_{\text{halt}}$ giving negative output. If \mathcal{M} does not halt, then no output is detected for either x_0 and x'_0 , and the system is unobservable on its initial state. ■

Theorem 11: Observability is undecidable for continuous rational piecewise-affine systems with any possible initial states.

Proof: Let \mathcal{M} be a Turing machine with M states and halting states Q_{halt} , let $\tilde{X} = [0, M] \times [0, 1]$ and let

\tilde{f} be a rational piecewise-affine map simulating \mathcal{M} on \tilde{X} , let $X = \tilde{X} \times [0, 1]$, let $f(x_1, x_2, x_3) = (\tilde{f}(x_1, x_2), x_3)$ and let $h : X \rightarrow X \times [0, 1] \times [0, 1]$ be given by $h(x_1, x_2, x_3) = (x_1, x_2, \phi(x_1, x_2)x_3)$ where $\phi = 0$ on $B_{q,a}$ for $q \notin Q_{\text{halt}}$, and $\phi \in (0, 1]$ otherwise.

Then the initial x_1, x_2 coordinates are known, so the initial state can be determined as soon as the x_3 coordinate is known. This occurs if ϕ ever becomes nonzero. This can only not occur if $f^n(x) \in B(Q, \Sigma)$ for all iterates, and further that the machine never halts. By the shadowing lemma, it suffices to consider points in $v(\mathcal{C})$, and then it is clear that the system is unobservable if and only if \mathcal{M} is mortal on an infinite tape'. Hence observability is undecidable, since mortality for Turing machines with an infinite tape is undecidable. ■

B. Piecewise-affine planar homeomorphisms

A reversible Turing machine is simulated by homeomorphisms of the plane. Since for a reversible system, knowing the current state gives full knowledge of the initial state, we can consider observability for planar systems and take $h(x) = (\phi(x), x_1\phi(x), x_2\phi(x))$. Observability is then equivalent to having every state entering the set where $\phi(x) \neq 0$ (though we need a dummy fixed state with $\phi(x) = 0$ so as not to allow a state staying in the set where $\phi = 0$ to be distinguished from every other state). Although [2] shows that any Turing machine is simulated by a reversible Turing machine, this is not enough to show that the mortality problem is undecidable. We obtain the following results.

Theorem 12:

- 1) Observability is undecidable for continuous reversible piecewise-affine planar systems with finitely many initial states.
- 2) If the mortality problem for reversible Turing machines with an infinite tape is undecidable, then so is observability for continuous rational piecewise-affine planar homeomorphisms.

C. Discontinuous piecewise-affine systems

Up to now we have considered continuous functions. However in many situations it is natural to model hybrid systems by discontinuous piecewise-affine functions. On the boundaries of partition elements, we take the functions to be multivalued; this mild form of nondeterminism does not cause any serious problems.

A multivalued function $f : X \rightarrow X$ is a *rational piecewise-affine function* if that X is a union of polytopes of the form $Bx \leq b$ on which $f(x) = Ax + b$. The function f is multivalued on a common boundary of two polytopes. A *piecewise-affine hybrid system* is specified by a rational piecewise-affine function f on a (possibly disconnected) union of polytopes, and a continuous observation function h . Note that the system as defined may be *nondeterministic*; when the intersection of polytopes is reached, the behaviour is undetermined. However, this does not cause difficulties in determining system observability.

Definition 13 (Discrete-state observability): A piecewise-affine hybrid system is *discrete-state observable* if it is possible to determine the initial discrete state from the output.

By reduction to the state observability problem for Turing machines, we obtain the following result for systems.

Theorem 14: Discrete-state observability is undecidable for discontinuous piecewise-affine hybrid systems.

V. FINITE- AND INFINITE-TIME OBSERVABILITY

A system is said to be *observable in time T* if every pair of trajectories can be distinguished by the output over the time interval $[0, T]$. A system can be

- Observable in *infinitesimal* time if it is observable in time T for any $T > 0$,
- Observable in *finite* time if it is observable in time T for some $T < \infty$.
- Observable in *infinite* time if any pair of trajectories can be distinguished on $[0, \infty)$.

A linear system is observable if and only if it is observable in infinitesimal time. For Turing machines with outputs, state observability must be in finite time, since there are only finitely many possible trajectories which need to be distinguished, and input observability must be in infinite time, since infinitely long inputs are allowed.

For discrete-time and hybrid systems, it is straightforward to construct examples which are observable in infinite time.

Example 15: Consider a Turing machine which reads its input from left to right and halts on finding the first zero. Then the piecewise-affine continuous simulator is observable in infinite time, but not in finite time, since there does not exist finite N such that \mathcal{M} halts in time N .

Example 16: Consider the continuous piecewise-affine system with state space $X = [0, 2]$ and output space $Y = [0, 1]$. Let

$$f(x) = \begin{cases} x+1 & \text{if } x \leq 1, \\ 2 & \text{if } x \geq 1; \end{cases} \quad h(x) = \begin{cases} 0 & \text{if } x \leq 1 \\ x-1, & \text{if } x \geq 1. \end{cases} \quad (2)$$

Then if $x \leq 1$, $h(x) = 0$ and $h(f(x)) = x$, whereas if $x \geq 1$, $h(x) = x-1$ and $h(f(x)) = 1$. Clearly, the outputs differ after one step for any two initial states. Hence the system is observable in time 1.

Example 17: Consider the continuous piecewise-affine system with state space $X = [0, 2]$ and output space $Y = [0, 1]$. Let

$$f(x) = \begin{cases} 2x & \text{if } x \leq 1, \\ 2 & \text{if } x \geq 1; \end{cases} \quad h(x) = \begin{cases} 0 & \text{if } x \leq 1, \\ x-1 & \text{if } x \geq 1. \end{cases} \quad (3)$$

Note that f and h are continuous at $x = 1$.

If $x_0 > 0$, then $f^n(x_0) > 1$ for some least n , the output is $(0, 0, \dots, 0, 2^n x_0 - 1, 1, 1, \dots)$ and $x_0 = (h(f^n(x_0)) + 1)/2^n$. If $x_0 = 0$, then the output is $(0, 0, \dots)$. Hence the output for any two initial points is different, but since there are points whose output consists of arbitrarily long initial sequences

of zeroes, it is not possible to distinguish $x_0 = 0$ from $x_0 > 0$ on any finite time interval.

VI. CONCLUSIONS AND FUTURE WORKS

A. Conclusions

In this paper we have shown how to extend results on undecidability of reachability to results on observability. Two observability problems for Turing machines were shown to be undecidable. These results translate immediately to undecidability results for classes of systems which can simulate Turing machines with suitable initial state sets. To obtain undecidability of observability for systems in which any initial state is allowed, it is necessary to take more care over the construction to ensure that the hybrid system is not trivially unobservable. Examples of piecewise-affine hybrid systems were given to show that observability may be in finite-time or even infinite time.

B. Future Works

These results show that for even relatively simple classes of system, observability is not decidable. However, observability is still an important system property which requires study. A number of approaches are possible. Sub-classes of system for which observability is decidable are known to exist, such as a result of Vidal et al. [18] for linear switched systems. Sufficient conditions for observability for general systems which are decidable may be given, such as observability in finite time [10] or observability after a single discrete event for piecewise-affine systems [19]. An alternative approach is to consider approximate system properties, in the same spirit as discrete-state observability, which may be decidable.

There are also links with information theory, since observability of a system providing an encoding can be viewed as decidability of the encoding; see [20]. The relationship between observability and realisations of piecewise-affine hybrid systems [21] is also an important area for further work.

VII. ACKNOWLEDGEMENTS

The authors gratefully acknowledge the financial support of the European Commission through the project Control and Computation (IST-2001-33520) of the Program Information Societies and Technologies.

REFERENCES

- [1] M. Sipser, *Introduction to the Theory of Computation*. Brooks Cole, 1996.
- [2] C. H. Bennett, "Logical reversibility of computation," *IBM J. Res. Develop.*, vol. 17, pp. 525–532, 1973.
- [3] K. Morita, "Universality of a reversible two-counter machine," *Theoret. Comput. Sci.*, vol. 168, no. 2, pp. 303–320, 1996.
- [4] P. K. Hooper, "The undecidability of the Turing machine immortality problem," *J. Symbolic Logic*, vol. 31, pp. 219–234, 1966.
- [5] V. D. Blondel, O. Bournez, P. Koiran, C. H. Papadimitriou, and J. N. Tsitsiklis, "Deciding stability and mortality of piecewise affine dynamical systems," *Theoret. Comput. Sci.*, vol. 255, no. 1-2, pp. 687–696, 2001.
- [6] C. Moore, "Generalized shifts: unpredictability and undecidability in dynamical systems," *Nonlinearity*, vol. 4, no. 2, pp. 199–230, 1991.
- [7] P. Koiran, M. Cosnard, and M. Garzon, "Computability with low-dimensional dynamical systems," *Theoret. Comput. Sci.*, vol. 132, no. 1–2, pp. 113–128, 1994.
- [8] E. Asarin, O. Maler, and A. Pnueli, "Reachability analysis of dynamical systems having piecewise-constant derivatives," *Theoret. Comput. Sci.*, vol. 138, no. 1, pp. 35–65, 1995.
- [9] H. T. Siegelmann and E. D. Sontag, "On the computational power of neural nets," *J. Comput. System Sci.*, vol. 50, no. 1, pp. 132–150, 1995.
- [10] E. D. Sontag, "From linear to nonlinear: Some complexity questions," in *Proceedings of the 34th IEEE Conference on Decision and Control*. New York: IEEE Press, 1995, pp. 2916–2920.
- [11] V. D. Blondel and J. N. Tsitsiklis, "A survey of computational complexity results in systems and control," *Automatica*, vol. 36, no. 9, pp. 1249–1274, 2000.
- [12] V. D. Blondel, O. Bournez, P. Koiran, and J. N. Tsitsiklis, "The stability of saturated linear dynamical systems is undecidable," *J. Comput. System Sci.*, vol. 62, no. 3, pp. 442–462, 2001.
- [13] R. Koplon, E. Sontag, and M. Hautus, "Observability of linear systems with saturated outputs," *Linear Algebra Appl.*, vol. 205–206, pp. 909–936, 1994.
- [14] E. D. Sontag, "On the observability of polynomial systems, i: Finite-time problems," *SIAM J. Control Optim.*, vol. 17, no. 1, pp. 139–151, 1979.
- [15] A. Grzegorzczak, "Computable functionals," *Fund. Math.*, vol. 42, pp. 168–202, 1955.
- [16] K. Weihrauch, *Computable analysis - An introduction*, ser. Texts in Theoretical Computer Science. Berlin: Springer-Verlag, 2000.
- [17] M. W. Hirsch, *Differential Topology*, ser. Graduate Texts In Mathematics. Springer-Verlag, 1976, no. 33.
- [18] R. Vidal, A. Chiuso, S. Soatto, and S. Sastry, "Observability of linear hybrid systems," in *Hybrid Systems: Computation and Control*, ser. Lecture Notes in Computer Science, O. Maler and A. Pnueli, Eds., no. 2623. Berlin Heidelberg New York: Springer-Verlag, 2003, pp. 527–539.
- [19] P. Collins and J. H. van Schuppen, "Observability of piecewise-affine hybrid systems," in *Hybrid Systems: Computation and Control*, ser. Lecture Notes in Computer Science, no. 2993. Berlin Heidelberg New York: Springer-Verlag, 2004, pp. 265–279.
- [20] T. M. Cover and J. A. Thomas, *Elements of information theory*, ser. Wiley Series in Telecommunications. New York: John Wiley & Sons, Inc., 1991.
- [21] L. Habets and J. H. van Schuppen, "Reduction of affine systems on polytopes," in *Proceedings of the International Symposium on the Mathematical Theory of Networks and Systems, Univ. of Notre Dame, Notre Dame, IN, U.S.A., August 2002, 2002*.