Piecewise-constant derivative systems and their algorithmic properties

Eugene Asarin
Institute for Information Transmission Problems, Moscow
Université Paris 12

June 6, 1994

[summary by Frédéric Chyzak]

Abstract

E. Asarin deals with simple differential systems: dynamical systems with piecewise-constant derivative—in short, PCD systems. Recently, it has been proved that the reachability problem is decidable in a 2-dimensional space [2]. Asarin and Maler proved in [1] that every Turing machine can be simulated by a 3-dimensional PCD system. Thus, the reachability problem is proved to be undecidable in more than two dimensions. Connections with automata theory and first order logic are also given. This summary is based on [1].

1. PCD systems and transition systems

PCD systems are special cases of differential systems; their solutions are trajectories in a continuous space. Besides, transition systems lead to trajectories in a discrete set. This section recalls results detailed in [1] that prove how transition systems can be simulated by PCD systems.

Definition 1. Let $\mathcal{E}$ be $\mathbb{R}^d$ seen as a d-dimensional Euclidean space and f be a vector field defined from a subset of $\mathcal{E}$ to $\mathcal{E}$. The *dynamical system* on $\mathcal{E}$ with respect to f is the differential system ruled by the equation

$$\frac{d^+x}{dt} = f(x),$$

where $\frac{d^+}{dt}$ is the right derivative and where x is a functional unknown in the time t. A *PCD system* is a dynamical system defined by a piecewise-constant vector field taking a finite number of values. A *trajectory* of a dynamical system is a solution of it.

The definition of a transition system formally looks like the previous one.

Definition 2. Let Q be a set whose elements are called *states* and δ a subset of $Q \times Q$. The *transition system* on Q with respect to δ is the system in the unknown $\sigma : \mathbb{N} \mapsto Q$ satisfying $(\sigma_n, \sigma_{n+1}) \in \delta$ for all $n \in \mathbb{N}$. A transition system is *deterministic* if δ reduces to a function.

All transition systems under consideration in this summary are deterministic though the following results also hold with non deterministic ones.

Let $\xi(t)$ be a solution of a PCD system defined on an interval $[0, r] \subseteq \mathbb{R}^+$. As described in [1], this trajectory can be discretised both in space and time and associated to a state sequence:

(1) first, each subset of $\mathcal{E}$ on which f is constant is associated to a state of Q: this is the discretisation in space;

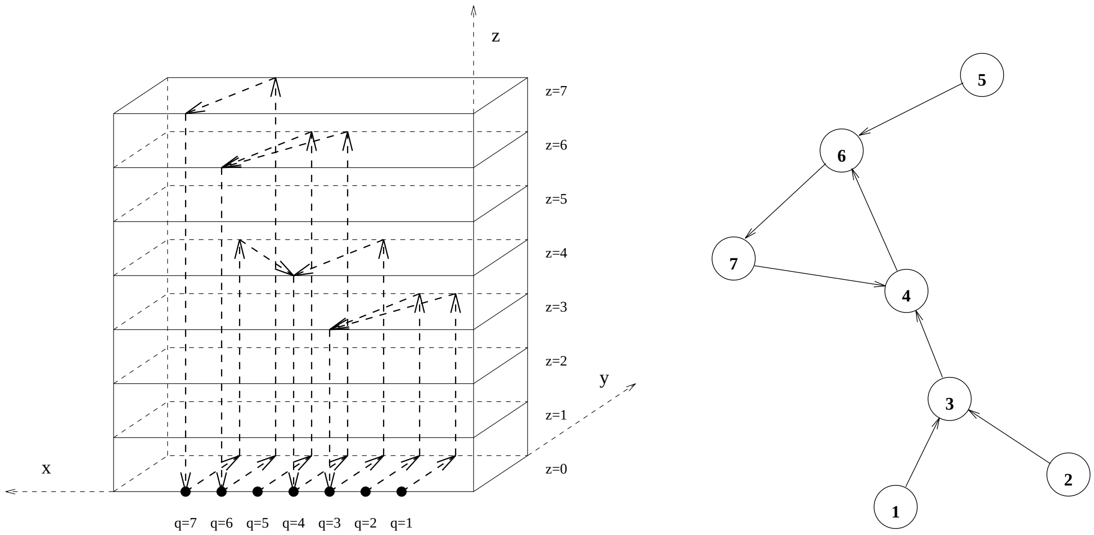

FIGURE 1. A 7-state automaton and its corresponding dynamical system

(2) then, the interval $[0, r]$ on which ξ is defined is divided into successive sub-intervals $[r_n, r_{n+1}]$ on which ξ is constant, so that $\bigcup_{n=0,\ldots,N} [\, r_n, r_{n+1} \,[\; = [\, 0, r\,[$, where $N \in \mathbb{N} \cup \{\infty\}$: this is the discretisation in time;

(3) finally, the associated state sequence $(\sigma_n)_{n=0,\ldots,N}$ is defined by letting σ_n be the state corresponding to $f([r_n, r_{n+1}])$.

2. Realisation of finite and pushdown automata

In his talk, Asarin focuses on a simple subclass of PCD systems for which the vector field under consideration is constant on *convex polyhedra*: the class of polyhedral-PCD systems. (A convex polyhedron is an intersection of a finite number of indifferently open or closed half-spaces of $\mathcal{E}$.)

The following result proves that PCD systems are at least as powerful as finite automata.

THEOREM 1. *Any finite automaton can be simulated by a 3-dimensional PCD system.*

Rather than a proof, an example of application of this theorem is given in Fig. 1. Note that a state of the automaton is represented by a point of the x-axis, and that the pipes used to lead from one state to another are 1-dimensional. One gets easily convinced that this construction is general.

The results concerning automata with infinite number of states are based on stack machines.

DEFINITION 3. Given an alphabet $\Sigma = \{0, \ldots, k-1\}$, a *stack* S is an element of the set Σ^ω of infinite words on Σ. Such a word is denoted by $s_0 s_1 \cdots$ where $s_n \in \Sigma$ for all $n \in \mathbb{N}$.

For simplicity, only the case $k = 2$ will be dealt with.

The set Σ^ω of possible stacks is certainly not denumerable. Actually, stacks have to be interpreted as binary representations of numbers of $[0, 1]$. Two particular operations can be processed on them.

(1) the PUSH function defined from $\Sigma \times \Sigma^\omega$ to Σ^ω by $\text{PUSH}(s, s_0 s_1 \cdots) = s s_0 s_1 \cdots$;

(2) and the converse POP function defined from Σ^ω to $\Sigma \times \Sigma^\omega$ by $\text{POP}(s_0 s_1 \cdots) = (s_0, s_1 s_2 \cdots)$.

Stack machines can now be defined. As obvious on the following definition, their sets of states $Q \times \Sigma^\omega$ are certainly not denumerable.

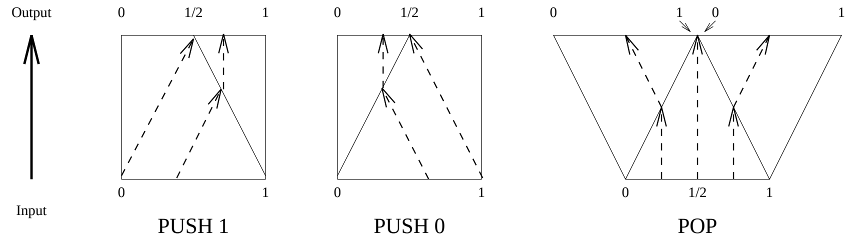

Figure 2. push and pop by PCD systems

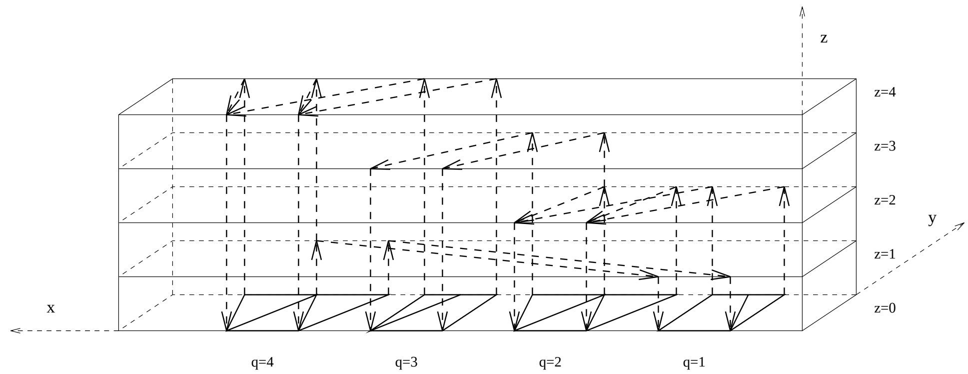

Figure 3

Definition 4. A *pushdown automaton* is a transition system $(Q \times \Sigma^\omega, \delta)$ for some finite set $Q = \{q_1, \ldots, q_n\}$ and some δ defined using transitions of either following forms:

(1) the push forms: $\delta(q_i, S) = (q_j, \text{PUSH}(v, S))$, defined for any $v \in \{0, \ldots, k-1\}$;
(2) the pop form: $\delta(q_i, S) = (q_{j_v}, S')$, if $(v, S') = \text{POP}(S)$ and the q_{j_v}'s are elements of Q.

The following result makes use of the continuous nature of $\mathcal{E}$ to prove a result for pushdown automata (PDA) that is similar to the one for finite automata.

Theorem 2. *Any pushdown automaton can be simulated by a 3-dimensional PCD system.*

Once again, only a sketch of the proof is given. The push and pop transitions are simulated by corresponding "ports" (see Fig. 2).

The pipes are now 2-dimensional strips of $\mathcal{E} = \mathbb{R}^3$, the second dimension encoding a stack. Figure 3 gives the PCD system encoding the PDA defined by:

$$q_1 : S := \text{PUSH}(1, S); \text{GOTO } q_2;$$
$$q_2 : (v, S) := \text{POP}(S); \text{IF } v = 1 \text{ GOTO } q_2, \text{ELSE GOTO } q_3;$$
$$q_3 : S := \text{PUSH}(0, S); \text{GOTO } q_4;$$
$$q_4 : (v, S) := \text{POP}(S); \text{IF } v = 1\text{GOTO } q_1, \text{ELSE GOTO } q_4;$$

3. Reachability problem and realisation of a Turing machine

For a fixed dynamical system $(\mathcal{E}, f)$, given any two points x and x' of $\mathcal{E}$, the reachability problem is to decide the existence of a trajectory ξ and of a $t \in \mathbb{R} - \{0\}$ such that $\xi(0) = x$ and $\xi(t) = x'$.

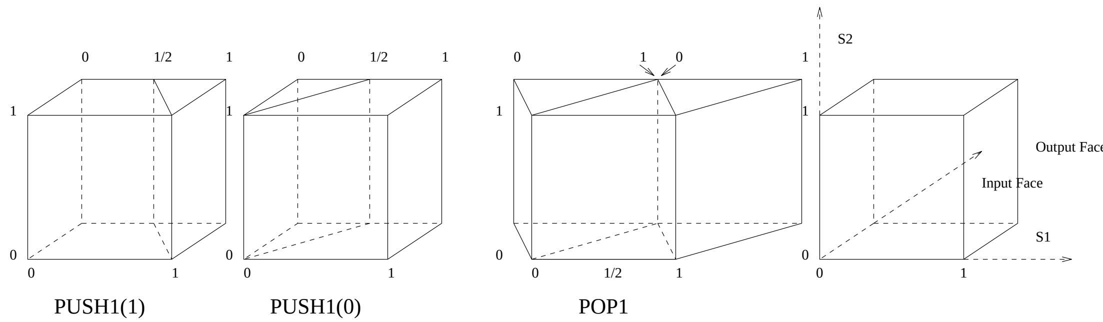

FIGURE 4. PUSH and POP with two stacks

The reachability problem for the class of polyhedral-PCD systems was proved to be decidable in the case of two dimensions by Maler and Pnueli in [2]. The last result about automata can be easily generalised to prove that it becomes undecidable in higher dimensions.

THEOREM 3. *Any pushdown automaton with 2 stacks can be simulated with a 4-dimensional PCD system.*

The idea of the proof is that adding a third dimension to the pipes makes it possible to encode simultaneously two stacks as a point of $[0,1]^2$. Of course, this additional dimension also increments the dimension of $\mathcal{E}$. The "ports" have two be replaced by those of Fig. 4 that push on and pop off the first stack, as well as with corresponding ones for the second stack.

Since any Turing machine can be realised by a pushdown automaton with two stacks, the following corollary holds.

COROLLARY 1. *Every Turing machine can be simulated with a 4-dimensional PCD system.*

Finally, since the halting problem of a Turing machine is equivalent to a reachability problem of a PCD system, the following corollary also holds.

COROLLARY 2. *The reachability problem is undecidable in the case of three (or more) dimensions.*

4. PCD systems and first order logic

Since PCD systems are at least as expressive as Turing machines, any given predicate P on integers $n_1, \ldots, n_k$ can be implemented as a PCD system with the k-tuple $n_1, \ldots, n_k$ in input: the reachability of a special state q_{true} is equivalent to the provability of $P(n_1, \ldots, n_k)$, the reachability of a special state q_{false} is equivalent to the provability of $\neg P(n_1, \ldots, n_k)$.

The more general following result will not be proved here.

THEOREM 4. *Any predicate on a tuple of integers with m changes of quantifiers can be computed with a PCD system on a space of dimension $10m$.*

Bibliography

[1] Asarin (Eugene) and Maler (Oded). – On some relations between dynamical systems and transition systems. In *Automata, Languages and Programming. Lecture Notes in Computer Science*, vol. 820. – Springer Verlag, 1994. Proceedings of the 21st International Colloquium, ICALP 94, Jerusalem, Israel.

[2] Maler (O.) and Pnueli (A.). – Reachability analysis of planar multi-linear systems. In Courcoubetis (C.) (editor), *Computer Aided Verification: Proceedings. Lecture Notes in Computer Science*, vol. 697, pp. 194–209. – Springer Verlag, 1993. Proceedings of the 5th International conference, CAV'93, Elounda, Greece.