Static Analysis for State-Space Reduction of Polygonal Hybrid Systems

Gordon Pace[1] and Gerardo Schneider[2]

[1] Dept. of Computer Science and AI, University of Malta, Msida, Malta
[2] Dept. of Informatics, University of Oslo, Oslo, Norway
gordon.pace@um.edu.mt, gerardo@ifi.uio.no

Abstract. Polygonal hybrid systems (SPDI) are a subclass of planar hybrid automata which can be represented by piecewise constant differential inclusions. The reachability problem as well as the computation of certain objects of the phase portrait, namely the viability, controllability and invariance kernels, for such systems is decidable. In this paper we show how to compute another object of an SPDI phase portrait, namely semi-separatrix curves and show how the phase portrait can be used for reducing the state-space for optimizing the reachability analysis.

1 Introduction

Hybrid systems combining discrete and continuous dynamics arise as mathematical models of various artificial and natural systems, and as approximations to complex continuous systems. They have been used in various domains, including avionics, robotics and bioinformatics. Reachability analysis has been the principal research question in the verification of hybrid systems, even if it is a well-known result that for most subclasses of hybrid systems most verification questions are undecidable. Various decidable subclasses have, subsequently, been identified, including timed [AD94] and rectangular automata [HKPV95], hybrid automata with linear vector fields [LPY01], piecewise constant derivative systems (PCDs) [MP93] and polygonal hybrid systems (SPDIs) [ASY01].

Compared to reachability verification, qualitative analysis of hybrid systems is a relatively neglected area [ALQ$^+$01b, DV95, MS00, SP02, SJSL00]. Typical qualitative questions include: 'Are there 'sink' regions where a trajectory can never leave once it enters the region?' and 'Are there regions in which every point in the region is reachable from every other?'. The collection of objects in a system satisfying these properties is called the *phase portrait* of the system.

Defining and constructing phase portraits of hybrid systems has been directly addressed for PCDs in [MS00], and for SPDIs in [ASY02]. Given a cycle on a SPDI, the *viability* kernel is the largest set of points in the cycle which may loop forever within the cycle. The *controllability* kernel is the largest set of strongly connected points in the cycle (such that any point in the set may be reached from any other). An *invariant set* is a set of points such that each point must keep rotating within the set forever, and the *invariance kernel* is the largest such set. Algorithms for computing these kernels have been presented in [ASY02, Sch04] and implemented in the tool set SPeeDI$^+$[PS].

E. Asarin and P. Bouyer (Eds.): FORMATS 2006, LNCS 4202, pp. 306–321, 2006.

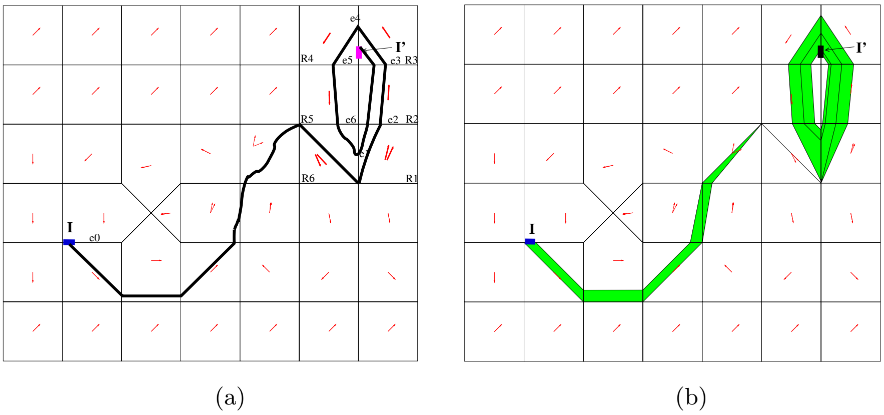


Fig. 1. (a) An SPDI and its trajectory segment; (b) Reachability analysis

The contribution of this paper is threefold. We start by introducing a new element of the phase portrait of SPDIs, *semi-separatrix curves*, and give an algorithm to compute them. Separatrices are convex polygons dissecting the plane into two mutually non-reachable subsets. We then show how the kernels can be used to answer reachability questions directly. We also show how semi-separatrices can be used to optimize the reachability algorithm for SPDIs by reducing the number of states of the SPDI graph. The optimization is based on topological properties of the plane (and in particular, those of SPDIs).

2 Theoretical Background

We summarize here the main definitions and results about SPDIs; for a more detailed description refer to [Sch02]. A (positive) *affine* function $f : \mathbb{R} \to \mathbb{R}$ is such that $f(x) = ax + b$ with $a > 0$. An *affine multivalued* function $F : \mathbb{R} \to 2^{\mathbb{R}}$, denoted $F = \langle f_l, f_u \rangle$, is defined by $F(x) = \langle f_l(x), f_u(x) \rangle$ where f_l and f_u are affine and $\langle \cdot, \cdot \rangle$ denotes an interval. For notational convenience, we do not make explicit whether intervals are open, closed, left-open or right-open, unless required for comprehension. For an interval $I = \langle l, u \rangle$ we have that $F(\langle l, u \rangle) = \langle f_l(l), f_u(u) \rangle$. The *inverse* of F is defined by $F^{-1}(x) = \{y \mid x \in F(y)\}$. The *universal inverse* of F is defined by $\tilde{F}^{-1}(I) = I'$ where I' is the greatest non-empty interval satisfying $\forall x \in I' \cdot F(x) \subseteq I$.

Clearly, $F^{-1} = \langle f_u^{-1}, f_l^{-1} \rangle$ and $\tilde{F}^{-1} = \langle f_l^{-1}, f_u^{-1} \rangle$, provided that $\langle f_l^{-1}, f_u^{-1} \rangle \neq \emptyset$.

A *truncated affine multivalued* function (TAMF) $\mathcal{F} : \mathbb{R} \to 2^{\mathbb{R}}$ is defined by an affine multivalued function F and intervals $S \subseteq \mathbb{R}^+$ and $J \subseteq \mathbb{R}^+$ as follows: $\mathcal{F}(x) = F(x) \cap J$ if $x \in S$, otherwise $\mathcal{F}(x) = \emptyset$. For convenience we write $\mathcal{F}(x) = F(\{x\} \cap S) \cap J$. For an interval I, $\mathcal{F}(I) = F(I \cap S) \cap J$ and $\mathcal{F}^{-1}(I) = F^{-1}(I \cap J) \cap S$. The *universal inverse* of $\mathcal{F}$ is defined by $\tilde{\mathcal{F}}^{-1}(I) = I'$ if and only if I' is the greatest non-empty interval such that for all $x \in I'$, $F(x) \subseteq I$ and

$F(x) = \mathcal{F}(x)$. We say that $\mathcal{F}$ is *normalized* if $S = \mathsf{Dom}(\mathcal{F}) = \{x \mid F(x) \cap J \neq \emptyset\}$ (thus, $S \subseteq F^{-1}(J)$) and $J = \mathsf{Im}(\mathcal{F}) = \mathcal{F}(S)$.

It can be proved [ASY01], that TAMFs are closed under composition.

Theorem 1. The composition of two TAMFs $\mathcal{F}_1(I) = F_1(I \cap S_1) \cap J_1$ and $\mathcal{F}_2(I) = F_2(I \cap S_2) \cap J_2$, is the TAMF $(\mathcal{F}_2 \circ \mathcal{F}_1)(I) = \mathcal{F}(I) = F(I \cap S) \cap J$, where $F = F_2 \circ F_1$, $S = S_1 \cap F_1^{-1}(J_1 \cap S_2)$ and $J = J_2 \cap F_2(J_1 \cap S_2)$. □

2.1 SPDIs

An *angle* $\angle_{\mathbf{a}}^{\mathbf{b}}$ on the plane, defined by two non-zero vectors $\mathbf{a}, \mathbf{b}$, is the set of all positive linear combinations $\mathbf{x} = \alpha\ \mathbf{a} + \beta\ \mathbf{b}$, with $\alpha, \beta \geq 0$, and $\alpha + \beta > 0$. We will assume that $\mathbf{b}$ is situated in the counter-clockwise direction from $\mathbf{a}$.

A *polygonal hybrid system*[1] (SPDI) is a finite partition $\mathbb{P}$ of the plane into convex polygonal sets, such that for each $P \in \mathbb{P}$ we have two vectors $\mathbf{a}_P$ and $\mathbf{b}_P$. Let $\phi(P) = \angle_{\mathbf{a}_P}^{\mathbf{b}_P}$. The SPDI is determined by $\dot{\mathbf{x}} \in \phi(P)$ for $\mathbf{x} \in P$.

Let $E(P)$ be the set of edges of P. We say that e is an *entry* of P if for all $\mathbf{x} \in e$ and for all $\mathbf{c} \in \phi(P)$, $\mathbf{x} + \mathbf{c}\epsilon \in P$ for some $\epsilon > 0$. We say that e is an *exit* of P if the same condition holds for some $\epsilon < 0$. We denote by $in(P) \subseteq E(P)$ the set of all entries of P and by $out(P) \subseteq E(P)$ the set of all exits of P.

Assumption 1. *All the edges in $E(P)$ are either entries or exits, that is, $E(P) = in(P) \cup out(P)$.*

Reachability for SPDIs is decidable provided the above assumption holds [ASY01]; without such assumption it is not know whether reachability is decidable.

A *trajectory segment* of an SPDI is a continuous function $\xi : [0, T] \to \mathbb{R}^2$ which is smooth everywhere except in a discrete set of points, and such that for all $t \in [0, T]$, if $\xi(t) \in P$ and $\dot{\xi}(t)$ is defined then $\dot{\xi}(t) \in \phi(P)$. The *signature*, denoted $\mathsf{Sig}(\xi)$, is the ordered sequence of edges traversed by the trajectory segment, that is, $e_1, e_2, \ldots$, where $\xi(t_i) \in e_i$ and $t_i < t_{i+1}$. If $T = \infty$, a trajectory segment is called a *trajectory.*

Example 1. Consider the SPDI illustrated in Fig. 1-(a). For sake of simplicity we will only show the dynamics associated to regions R_1 to R_6 in the picture. For each region R_i, $1 \leq i \leq 6$, there is a pair of vectors $(\mathbf{a}_i, \mathbf{b}_i)$, where: $\mathbf{a}_1 = (45, 100), \mathbf{b}_1 = (1, 4)$, $\mathbf{a}_2 = \mathbf{b}_2 = (1, 10)$, $\mathbf{a}_3 = \mathbf{b}_3 = (-2, 3)$, $\mathbf{a}_4 = \mathbf{b}_4 = (-2, -3)$, $\mathbf{a}_5 = \mathbf{b}_5 = (1, -15)$, $\mathbf{a}_6 = (1, -2), \mathbf{b}_6 = (1, -1)$. A trajectory segment starting on interval $I \subset e_0$ and finishing in interval $I' \subseteq e_4$ is depicted. ■

We say that a signature σ is *feasible* if and only if there exists a trajectory segment ξ with signature σ, i.e., $\mathsf{Sig}(\xi) = \sigma$. From this definition, it immediately follows that extending an unfeasible signature can never make it feasible:

[1] In the literature the names *polygonal differential inclusion* and *simple planar differential inclusion* have been used to describe the same systems.

Proposition 1. *If a signature σ is not feasible, then neither is any extension of the signature — for any signatures σ' and σ'', the signature $\sigma'\sigma\sigma''$ is not feasible.* □

Given an SPDI $\mathcal{S}$, let $\mathcal{E}$ be the set of edges of $\mathcal{S}$, then we can define a graph $\mathcal{G}_\mathcal{S}$ where nodes correspond to edges of $\mathcal{S}$ and such that there exists an arc from one node to another if there exists a trajectory segment from the first edge to the second one without traversing any other edge. More formally: Given an SPDI $\mathcal{S}$, the *underlying graph of* $\mathcal{S}$ (or simply the *graph of* $\mathcal{S}$), is a graph $\mathcal{G}_\mathcal{S} = (N_\mathcal{G}, A_\mathcal{G})$, with $N_\mathcal{G} = \mathcal{E}$ and $A_\mathcal{G} = \{(e, e') \mid \exists \xi, t \,.\, \xi(0) \in e \wedge \xi(t) \in e' \wedge \mathsf{Sig}(\xi) = ee'\}$. We say that a sequence $e_0 e_1 \ldots e_k$ of nodes in $\mathcal{G}_\mathcal{S}$ is a *path* whenever $(e_i, e_{i+1}) \in A_\mathcal{G}$ for $0 \leq i \leq k-1$.

The following lemma shows the relation between edge signatures in an SPDI and paths in its corresponding graph.

Lemma 1. *If ξ is a trajectory segment of $\mathcal{S}$ with edge signature $\mathsf{Sig}(\xi) = \sigma = e_0 \ldots e_p$, it follows that σ is a path in $\mathcal{G}_\mathcal{S}$.* □

Note that the converse of the above lemma is not true in general. It is possible to find a counter-example where there exists a path from node e to e', but no trajectory from edge e to edge e' in the SPDI.

2.2 Successors and Predecessors

Given an SPDI, we fix a one-dimensional coordinate system on each edge to represent points laying on edges [ASY01]. For notational convenience, we indistinctly use letter e to denote the edge or its one-dimensional representation. Accordingly, we write $\mathbf{x} \in e$ or $x \in e$, to mean "point $\mathbf{x}$ in edge e with coordinate x in the one-dimensional coordinate system of e". The same convention is applied to sets of points of e represented as intervals (e.g., $\mathbf{x} \in I$ or $x \in I$, where $I \subseteq e$) and to trajectories (e.g., "ξ starting in x" or "ξ starting in $\mathbf{x}$").

Now, let $P \in \mathbb{P}$, $e \in in(P)$ and $e' \in out(P)$. For $I \subseteq e$, $\mathsf{Succ}_{e,e'}(I)$ is the set of all points in e' reachable from some point in I by a trajectory segment $\xi : [0, t] \to \mathbb{R}^2$ in P (i.e., $\xi(0) \in I \wedge \xi(t) \in e' \wedge \mathsf{Sig}(\xi) = ee'$). $\mathsf{Succ}_{e,e'}$ is a TAMF [ASY01].

Example 2. Let $e_1, \ldots, e_6$ be as in Fig. 1-(a), where all the edges have local coordinates over $[0, 10]$, and $I = [l, u]$. We assume a one-dimensional coordinate system. We show only the first and last edge-to-edge TAMF of the cycle:

$$\begin{array}{lll} F_{e_1 e_2}(I) = \left[\frac{l}{4}, \frac{9}{20}u\right], & S_1 = [0, 10], & J_1 = \left[0, \frac{9}{2}\right] \\ F_{e_6 e_1}(I) = [l, 2u], & S_6 = [0, 10], & J_6 = [0, 10] \end{array}$$

with $\mathsf{Succ}_{e_i e_{i+1}}(I) = F_{e_i e_{i+1}}(I \cap S_i) \cap J_i$, for $1 \leq i \leq 6$; S_i and J_i are computed as shown in Theorem 1. ■

Given a sequence $w = e_1, e_2, \ldots, e_n$, since TAMFs are closed under composition, the successor of I along w, defined as $\mathsf{Succ}_w(I) = \mathsf{Succ}_{e_{n-1}, e_n} \circ \ldots \circ \mathsf{Succ}_{e_1, e_2}(I)$, is a TAMF.

Example 3. Let $\sigma = e_1 \cdots e_6 e_1$. We have that $\mathsf{Succ}_\sigma(I) = F(I \cap S_\sigma) \cap J_\sigma$, where: $F(I) = [\frac{l}{4} + \frac{1}{3}, \frac{9}{10}u + \frac{2}{3}]$, with $S_\sigma = [0, 10]$ and $J_\sigma = [\frac{1}{3}, \frac{29}{3}]$. ■

For $I \subseteq e'$, $\mathsf{Pre}_{e,e'}(I)$ is the set of points in e that can reach a point in I by a trajectory segment in P. The $\forall$-*predecessor* $\widetilde{\mathsf{Pre}}(I)$ is defined in a similar way to $\mathsf{Pre}(I)$ using the universal inverse instead of just the inverse: For $I \subseteq e'$, $\widetilde{\mathsf{Pre}}_{ee'}(I)$ is the set of points in e such that *any* successor of such points are in I by a trajectory segment in P. Both definitions can be extended straightforwardly to signatures $\sigma = e_1 \cdots e_n$: $\mathsf{Pre}_\sigma(I)$ and $\widetilde{\mathsf{Pre}}_\sigma(I)$. The successor operator thus has two "inverse" operators.

2.3 Qualitative Analysis of Simple Edge-Cycles

Let $\sigma = e_1 \cdots e_k e_1$ be a simple edge-cycle, i.e., $e_i \neq e_j$ for all $1 \leq i \neq j \leq k$. Let $\mathsf{Succ}_\sigma(I) = F(I \cap S_\sigma) \cap J_\sigma$ with $F = \langle f_l, f_u \rangle$ (we suppose that this representation is normalized). We denote by $\mathcal{D}_\sigma$ the one-dimensional discrete-time dynamical system defined by Succ_σ, that is $x_{n+1} \in \mathsf{Succ}_\sigma(x_n)$.

Assumption 2. *None of the two functions f_l, f_u is the identity.*

Without the above assumption the results are still valid but need a special treatment making the presentation more complicated.

Let l^* and u^* be the fixpoints[2] of f_l and f_u, respectively, and $S_\sigma \cap J_\sigma = \langle L, U \rangle$. A simple cycle is of one of the following types [ASY01]: STAY, the cycle is not abandoned neither by the leftmost nor the rightmost trajectory, that is, $L \leq l^* \leq u^* \leq U$; DIE, the rightmost trajectory exits the cycle through the left (consequently the leftmost one also exits) or the leftmost trajectory exits the cycle through the right (consequently the rightmost one also exits), that is, $u^* < L \vee l^* > U$; EXIT-BOTH, the leftmost trajectory exits the cycle through the left and the rightmost one through the right, that is, $l^* < L \wedge u^* > U$; EXIT-LEFT, the leftmost trajectory exits the cycle (through the left) but the rightmost one stays inside, that is, $l^* < L \leq u^* \leq U$; EXIT-RIGHT, the rightmost trajectory exits the cycle (through the right) but the leftmost one stays inside, that is, $L \leq l^* \leq U < u^*$.

Example 4. Let $\sigma = e_1 \cdots e_6 e_1$. Then, $S_\sigma \cap J_\sigma = \langle L, U \rangle = [\frac{1}{3}, \frac{29}{3}]$. The fixpoints from Example 3 are $\frac{1}{3} < l^* = \frac{11}{25} < u^* = \frac{20}{3} < \frac{29}{3}$. Thus, σ is a STAY. ■

Any trajectory that enters a cycle of type DIE will eventually quit it after a finite number of turns. If the cycle is of type STAY, all trajectories that happen to enter it will keep turning inside it forever. In all other cases, some trajectories will turn for a while and then exit, and others will continue turning forever. This information is crucial for proving decidability of the reachability problem.

Example 5. Consider the SPDI of Fig. 1-(a). Fig. 1-(b) shows part of the reach set of the interval $[8, 10] \subset e_0$, answering positively to the reachability question:

[2] The fixpoint x^* is the solution of $f(x^*) = x^*$, where $f(\cdot)$ is positive affine.

Is $[1, 2] \subset e_4$ reachable from $[8, 10] \subset e_0$? Fig. 1-(b) has been automatically generated by the SPeeDI toolbox we have developed for reachability analysis of SPDIs [APSY02]. ■

2.4 Kernels

We present now how to compute the invariance, controllability and viability kernels of an SPDI. Proofs are omitted but for further details, refer to [ASY02] and [Sch04]. In the following, for σ a cyclic signature, we define $K_\sigma \subseteq \mathbb{R}^2$ as follows: $K_\sigma = \bigcup_{i=1}^{k}(\mathit{int}(P_i) \cup e_i)$ where P_i is such that $e_{i-1} \in \mathit{in}(P_i)$, $e_i \in \mathit{out}(P_i)$ and $\mathit{int}(P_i)$ is P_i's interior.

Viability Kernel. We now recall the definition of *viability kernel* [Aub01]. A trajectory ξ is *viable* in K if $\xi(t) \in K$ for all $t \geq 0$. K is a *viability domain* if for every $\mathbf{x} \in K$, there exists at least one trajectory ξ, with $\xi(0) = \mathbf{x}$, which is viable in K. The *viability kernel* of K, denoted $\mathsf{Viab}(K)$, is the largest viability domain contained in K.

For $I \subseteq e_1$ we define $\overline{\mathsf{Pre}}_\sigma(I)$ to be the set of all $\mathbf{x} \in \mathbb{R}^2$ for which there exists a trajectory segment ξ starting in $\mathbf{x}$, that reaches some point in I, such that $\mathsf{Sig}(\xi)$ is a suffix of $e_2 \dots e_k e_1$. It is easy to see that $\overline{\mathsf{Pre}}_\sigma(I)$ is a polygonal subset of the plane which can be calculated using the following procedure. We start by defining $\overline{\mathsf{Pre}}_e(I) = \{\mathbf{x} \mid \exists \xi : [0, t] \to \mathbb{R}^2, t > 0 \,.\, \xi(0) = \mathbf{x} \wedge \xi(t) \in I \wedge \mathsf{Sig}(\xi) = e\}$ and apply this operation k times: $\overline{\mathsf{Pre}}_\sigma(I) = \bigcup_{i=1}^{k} \overline{\mathsf{Pre}}_{e_i}(I_i)$ with $I_1 = I$, $I_k = \mathsf{Pre}_{e_k, e_1}(I_1)$ and $I_i = \mathsf{Pre}_{e_i, e_{i+1}}(I_{i+1})$, for $2 \leq i \leq k - 1$.

The following result provides a non-iterative algorithmic procedure for computing the viability kernel of K_σ on an SPDI:

Theorem 2. *If σ is DIE,* $\mathsf{Viab}(K_\sigma) = \emptyset$*, otherwise* $\mathsf{Viab}(K_\sigma) = \overline{\mathsf{Pre}}_\sigma(S_\sigma)$. □

Example 6. Fig. 2-(a) shows all the viability kernels of the SPDI given in Example 1. There are 4 cycles with viability kernels — in the picture two of the kernels are overlapping. ■

Controllability Kernel. We say K is *controllable* if for any two points $\mathbf{x}$ and $\mathbf{y}$ in K there exists a trajectory segment ξ starting in $\mathbf{x}$ that reaches an arbitrarily small neighborhood of $\mathbf{y}$ without leaving K. More formally: A set K is controllable if $\forall \mathbf{x}, \mathbf{y} \in K, \forall \delta > 0, \exists \xi : [0, t] \to \mathbb{R}^2, t > 0 \,.\, (\xi(0) = \mathbf{x} \wedge |\xi(t) - \mathbf{y}| < \delta \wedge \forall t' \in [0, t] \,.\, \xi(t') \in K)$. The *controllability kernel* of K, denoted $\mathsf{Cntr}(K)$, is the largest controllable subset of K.

For a given cyclic signature σ, we define $\mathcal{C}_\mathcal{D}(\sigma)$ as follows:

$$\mathcal{C}_\mathcal{D}(\sigma) = \begin{cases} \langle L, U \rangle & \text{if } \sigma \text{ is EXIT-BOTH} \\ \langle L, u^* \rangle & \text{if } \sigma \text{ is EXIT-LEFT} \\ \langle l^*, U \rangle & \text{if } \sigma \text{ is EXIT-RIGHT} \\ \langle l^*, u^* \rangle & \text{if } \sigma \text{ is STAY} \\ \emptyset & \text{if } \sigma \text{ is DIE} \end{cases} \qquad (1)$$

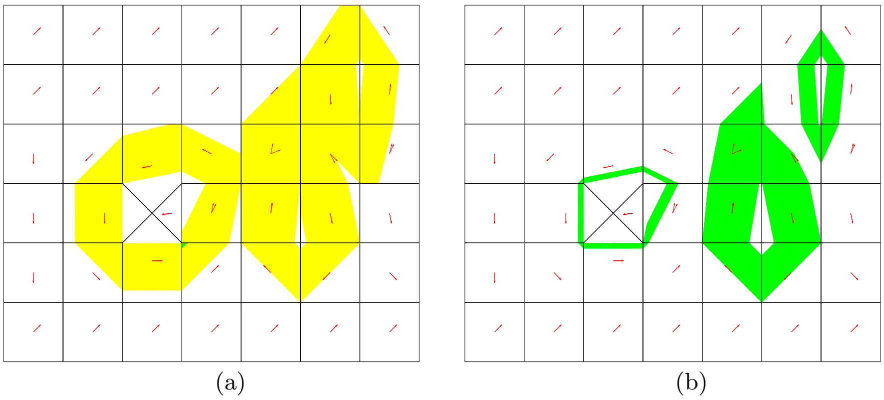


Fig. 2. (a) Viability kernels; (b) Controllability kernels

For $I \subseteq e_1$ let us define $\overline{\mathsf{Succ}}_\sigma(I)$ as the set of all points $\mathbf{y} \in \mathbb{R}^2$ for which there exists a trajectory segment ξ starting in some point $x \in I$, that reaches $\mathbf{y}$, such that $\mathsf{Sig}(\xi)$ is a prefix of $e_1 \dots e_k$. The successor $\overline{\mathsf{Succ}}_\sigma(I)$ is a polygonal subset of the plane which can be computed similarly to $\overline{\mathsf{Pre}}_\sigma(I)$. Define $\mathcal{C}(\sigma) = (\overline{\mathsf{Succ}}_\sigma \cap \overline{\mathsf{Pre}}_\sigma)(\mathcal{C}_\mathcal{D}(\sigma))$. We compute the controllability kernel of K_σ as follows:

Theorem 3. $\mathsf{Cntr}(K_\sigma) = \mathcal{C}(\sigma)$. □

Example 7. Fig. 2-(b) shows all the controllability kernels of the SPDI given in Example 1. There are 4 cycles with controllability kernels — in the picture two of the kernels are overlapping. ■

The following result which relates controllability and viability kernels, states that the viability kernel of a given cycle is the local basin of attraction of the corresponding controllability kernel.

Proposition 2. *Any viable trajectory in* K_σ *converges to* $\mathsf{Cntr}(K_\sigma)$. □

Let $\mathsf{Cntr}^l(K_\sigma)$ be the closed curve obtained by taking the leftmost trajectory and $\mathsf{Cntr}^u(K_\sigma)$ be the closed curve obtained by taking the rightmost trajectory which can remain inside the controllability kernel. In other words, $\mathsf{Cntr}^l(K_\sigma)$ and $\mathsf{Cntr}^u(K_\sigma)$ are the two polygons defining the controllability kernel.

A non-empty controllability kernel $\mathsf{Cntr}(K_\sigma)$ of a given cyclic signature σ partitions the plane into three disjoint subsets: (1) the controllability kernel itself, (2) the set of points limited by $\mathsf{Cntr}^l(K_\sigma)$ (and not including $\mathsf{Cntr}^l(K_\sigma)$) and (3) the set of points limited by $\mathsf{Cntr}^u(K_\sigma)$ (and not including $\mathsf{Cntr}^u(K_\sigma)$). We define the *inner* of $\mathsf{Cntr}(K_\sigma)$ (denoted by $\mathsf{Cntr}_{in}(K_\sigma)$) to be the subset defined by (2) above if the cycle is counter-clockwise or to be the subset defined by (3) if it is clockwise. The *outer* of $\mathsf{Cntr}(K_\sigma)$ (denoted by $\mathsf{Cntr}_{out}(K_\sigma)$) is defined to be the subset which is not the inner nor the controllability itself. Note that an edge in the SPDI may intersect a controllability kernel. In such cases, we can generate a different SPDI, with the same dynamics but with the edge split into parts, such that each part is completely inside, on or outside the kernel. Although the signatures will obviously change, it is easy to prove that the behaviour of the SPDI remains identical to the original. In the rest of the paper, we will assume that all edges are either completely inside, on or completely outside the kernels. We note that in practice splitting is not necessary since we can just consider parts of edges.

Proposition 3. *Given two edges e and e', one lying completely inside a controllability kernel, and the other outside or on the same controllability kernel, such that ee' is feasible, then there exists a point on the controllability kernel, which is reachable from e and from which e' is reachable.* □

Invariance Kernel. In general, an *invariant set* is a set of points such that for any point in the set, every trajectory starting in such point remains in the set forever and the *invariance kernel* is the largest of such sets. In particular, for an SPDI, given a cyclic signature, an *invariant set* is a set of points which keep rotating in the cycle forever and the *invariance kernel* is the largest of such sets. More formally: A set K is said to be *invariant* if for any $x \in K$ there exists at least one trajectory starting in it and every trajectory starting in x is viable in K. Given a set K, its largest invariant subset is called the *invariance kernel* of K and is denoted by $\mathsf{Inv}(K)$. We need some preliminary definitions before showing how to compute the kernel. The *extended* *∀-predecessor* of an output edge e of a region R is the set of points in R such that every trajectory segment starting in such point reaches e without traversing any other edge. More formally, let R be a region and e be an edge in $out(R)$, then the *e-extended ∀-predecessor* of I, $\widetilde{\overline{\mathsf{Pre}}}_e(I)$ is defined as: $\widetilde{\overline{\mathsf{Pre}}}_e(I) = \{\mathbf{x} \mid \forall\xi\ .\ (\xi(0) = \mathbf{x} \Rightarrow \exists t \geq 0\ .\ (\xi(t) \in I \wedge \mathsf{Sig}(\xi[0,t]) = e))\}$. It is easy to see that $\widetilde{\overline{\mathsf{Pre}}}_\sigma(I)$ is a polygonal subset of the plane which can be calculated using a similar procedure as for $\overline{\mathsf{Pre}}_\sigma(I)$. We compute the invariance kernel of K_σ as follows:

Theorem 4. *If σ is STAY then* $\mathsf{Inv}(K_\sigma) = \widetilde{\overline{\mathsf{Pre}}}_\sigma(\widetilde{\mathsf{Pre}}_\sigma(J_\sigma))$*, otherwise it is $\emptyset$.* □

Example 8. Fig. 3-(a) shows the unique invariance kernel of the SPDI given in Example 1. ■

An interesting property of invariance kernels is that the limits are included in the invariance kernel, i.e. $[l^*, u^*] \subseteq \mathsf{Inv}(K_\sigma)$. In other words:

Proposition 4. *The set delimited by the polygons defined by the interval $[l^*, u^*]$ is an invariance set of STAY cycles.* □

The following result relates controllability and invariance kernels.

Proposition 5. *If σ is STAY then* $\mathsf{Cntr}(K_\sigma) \subseteq \mathsf{Inv}(K_\sigma)$. □

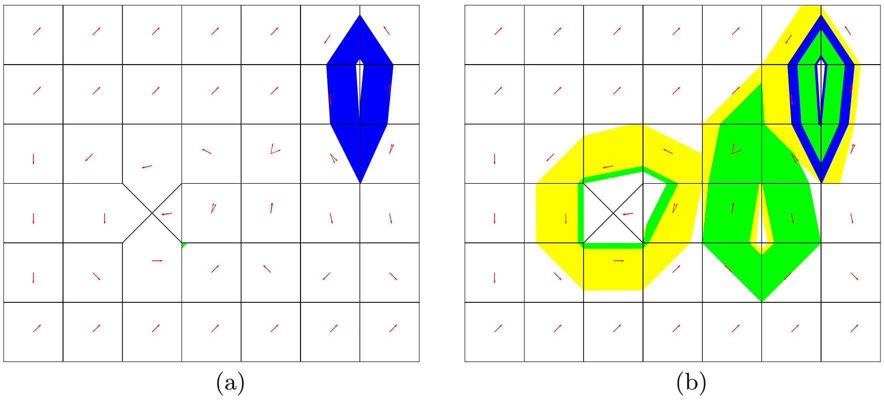


Fig. 3. (a) Invariance kernel; (b) All the kernels

Example 9. Fig. 3-(b) shows the viability, controllability and invariance kernels of the SPDI given in Example 1. For any point in the viability kernel of a cycle there exists a trajectory which will converge to its controllability kernel (proposition 2). It is possible to see in the picture that $\mathsf{Cntr}(\cdot) \subset \mathsf{Inv}(.)$ (proposition 5). All the above pictures has been obtained with the toolbox SPeeDI^{+} [PS]. ■

In a similar way as for the controllability kernel, we define $\mathsf{Inv}^{l}(K_\sigma)$ and $\mathsf{Inv}^{u}(K_\sigma)$.

3 Semi-separatrix Curves

In this section we define the notion of *separatrix curves*, which are curves dissecting the plane into two mutually non-reachable subsets, and *semi-separatrix curves* which can only be crossed in one direction. All the proofs of this and forthcoming sections may be found in [PS06]. We start by defining these notions independently of SPDIs.

Definition 1. *Let $K \subseteq \mathbb{R}^2$. A* separatrix *in K is a closed curve γ partitioning K into three sets K_A, K_B and γ itself, such that K_A, K_B and γ are pairwise disjoint, $K = K_A \cup K_B \cup \gamma$ and the following conditions hold: (1) For any point $\mathbf{x}_0 \in K_A$ and trajectory ξ, with $\xi(0) = \mathbf{x}_0$, there is no t such that $\xi(t) \in K_B$; and (2) For any point $\mathbf{x}_0 \in K_B$ and trajectory ξ, with $\xi(0) = \mathbf{x}_0$, there is no t such that $\xi(t) \in K_A$. If only one of the above conditions holds then we say that the curve is a* semi-separatrix. *If only condition 1 holds, then we say that K_A is the* inner *of γ (written γ_{in}) and K_B is the* outer *of γ (written γ_{out}). If only condition 2 holds, K_B is the* inner *and K_A is the* outer *of γ.*

Notice that, as in the case of the controllability kernel, an edge of the SPDI may be split into two by a semi-separatrix — part inside, and part outside. As before, we can split the edge into parts, such that each part is completely inside, or completely outside the semi-separatrix.

The above notions are extended to SPDIs straightforwardly. The set of all the separatrices of an SPDI $\mathcal{S}$ is denoted by $\mathsf{Sep}(\mathcal{S})$, or simply Sep.

Now, let $\sigma = e_1 \ldots e_n e_1$ be a simple cycle, $\angle_{\mathbf{a}_i}^{\mathbf{b}_i}$ $(1 \leq i \leq n)$ be the dynamics of the regions for which e_i is an entry edge and $I = [l, u]$ an interval on edge e_1. Remember that $\mathsf{Succ}_{e_1 e_2}(I) = F(I \cap S_1) \cap J_1$, where $F(x) = [a_1 x + b_1, a_2 x + b_2]$. Let $\mathbf{l}$ be the vector corresponding to the point on e_1 with local coordinates l and $\mathbf{l}'$ be the vector corresponding to the point on e_2 with local coordinates $F(l)$ (similarly, we define $\mathbf{u}$ and $\mathbf{u}'$ for $F(u)$). We define first $\overline{\mathsf{Succ}}_{e_1}^{\mathbf{b}_1}(I) = \{\mathbf{l} + \alpha(\mathbf{l}' - \mathbf{l}) \mid 0 < \alpha < 1\}$ and $\overline{\mathsf{Succ}}_{e_1}^{\mathbf{a}_1}(I) = \{\mathbf{u} + \alpha(\mathbf{u}' - \mathbf{u}) \mid 0 < \alpha < 1\}$. We extend these definitions in a straight way to any (cyclic) signature $\sigma = e_1 \ldots e_n e_1$, denoting them by $\overline{\mathsf{Succ}}_{\sigma}^{\mathbf{b}}(I)$ and $\overline{\mathsf{Succ}}_{\sigma}^{\mathbf{a}}(I)$, respectively; we can compute them similarly as for $\overline{\mathsf{Pre}}$. Whenever applied to the fixpoint $I^* = [l^*, u^*]$, we denote $\overline{\mathsf{Succ}}_{\sigma}^{\mathbf{b}}(I^*)$ and $\overline{\mathsf{Succ}}_{\sigma}^{\mathbf{a}}(I^*)$ by ξ_σ^l and ξ_σ^u respectively. Intuitively, ξ_σ^l (ξ_σ^u) denotes the piece-wise affine closed curve defined by the leftmost (rightmost) fixpoint l^* (u^*).

We show now how to identify semi-separatrices for simple cycles.

Theorem 5. *Given an SPDI, let σ be a simple cycle, then the following hold:*

1. *If σ is EXIT-RIGHT then ξ_σ^l is a semi-separatrix curve (filtering trajectories from "left" to "right");*
2. *If σ is EXIT-LEFT then ξ_σ^u is a semi-separatrix curve (filtering trajectories from "right" to "left");*
3. *If σ is STAY, then the two polygons defining the invariance kernel ($\mathsf{Inv}^l(K_\sigma)$ and $\mathsf{Inv}^u(K_\sigma)$), are semi-separatrices.* □

In the case of STAY cycles, ξ_σ^l and ξ_σ^u are also semi-separatrices. Notice that in the above result, computing a semi-separatrix depends only on one simple cycle, and the corresponding algorithm is then reduced to find simple cycles in the SPDI and checking whether it is STAY, EXIT-RIGHT or EXIT-LEFT. DIE cycles induce an infinite number of semi-separatrices and are not treated in this setting.

Example 10. Fig. 4 shows all the semi-separatrices of the SPDI given in Example 1, obtained as shown in Theorem 5. The small arrows traversing the semi-separatrices show the inner and outer of each semi-separatrix: a trajectory may traverse the semi-separatrix following the direction of the arrow, but not vice-versa. ■

The following two results relate feasible signatures and semi-separatrices.

Proposition 6. *If, for some semi-separatrix γ, $e \in \gamma_{in}$ and $e' \in \gamma_{out}$, then the signature ee' is not feasible.* □

Proposition 7. *If, for some semi-separatrix γ, and signature σ (of at least length 2), then, if head(σ) $\in \gamma_{in}$ and last(σ) $\in \gamma_{out}$, σ is not feasible.* □

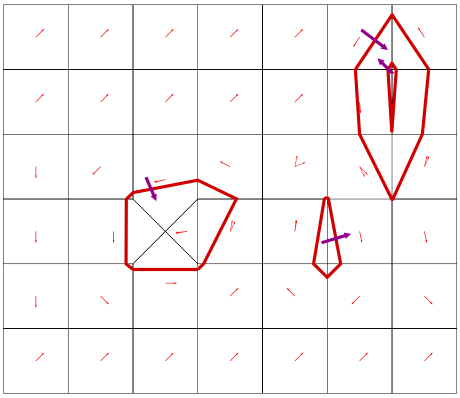


Fig. 4. Semi-separatrices

4 State-Space Reduction Using Semi-separatrices

Semi-separatrices partition the state space into two parts[3] – once one crosses such a border, all states outside the region can be ignored. We present a technique, which, given an SPDI and a reachability question, enables us to discard portions of the state space based on this information. The approach is based on identifying *inert* states (edges in the SPDI) not playing a role in the reachability analysis.

Definition 2. *Given an SPDI $\mathcal{S}$, a semi-separatrix $\gamma \in \mathsf{Sep}$, a source edge $e0$ and a destination edge $e1$, an edge e is said to be* inert *if it lies outside the semi-separatrix while $e0$ lies inside, or it lies inside, while $e1$ lies outside:*

$$inert^{\gamma}_{e0 \to e1} = \{e : \mathcal{E} \mid e0 \in \gamma_{in} \wedge e \in \gamma_{out}\} \cup \{e : \mathcal{E} \mid e1 \in \gamma_{out} \wedge e \in \gamma_{in}\}.$$

We can prove that these inert edges can never appear in a feasible signature:

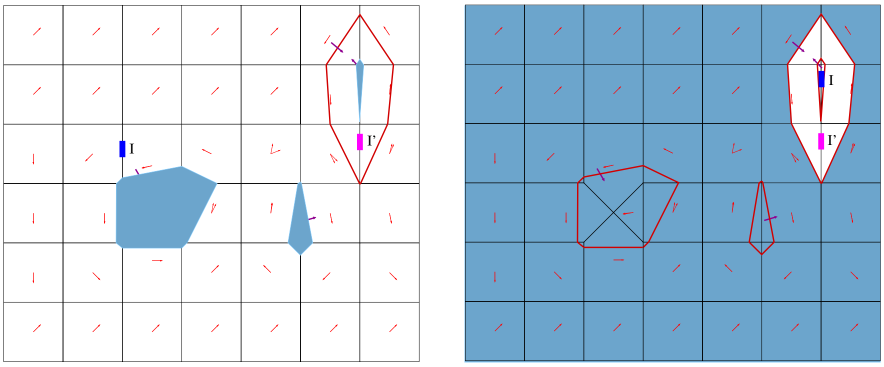
Lemma 2. *Given an SPDI $\mathcal{S}$, a semi-separatrix γ, a source edge $e0$ and a destination edge $e1$, and a feasible signature $e0\sigma e1$ in $\mathcal{S}$. No inert edge from $inert^{\gamma}_{e0 \to e1}$ may appear in $e0\sigma e1$.* □

Given an SPDI, we can reduce the state space by discarding inert edges.

Definition 3. *Given an SPDI $\mathcal{S}$, a semi-separatrix γ, a source edge $e0$ and a destination edge $e1$, we define the reduced SPDI $\mathcal{S}^{\gamma}_{e0 \to e1}$ to be the same as $\mathcal{S}$ but without the inert edges.*

Clearly, the resulting SPDI is not bigger than the original one. Finally, we prove that checking reachability on the reduced SPDI is equivalent to checking reachability on the original SPDI:

[3] Here, we do not consider the semi-separatrix itself.



(a) (b)

Fig. 5. Reduction using semi-separatrices

Theorem 6. *Given an SPDI $\mathcal{S}$, a semi-separatrix γ, and edges $e0$ and $e1$, then, $e1$ is reachable from $e0$ in $\mathcal{S}$ if and only if $e1$ is reachable from $e0$ in $\mathcal{S}^{\gamma}_{e0\to e1}$.* □

We have shown, that once semi-separatrices are identified, given a reachability question, we can reduce the size of the SPDI to be verified by removing inert edges of all the known semi-separatrices.

Example 11. The shaded areas of Fig. 5 (a) and (b) are examples of subsets of the SPDI edges of the reachability graph, eliminated by the reduction presented in this section applied to all semi-separatrices, when answering reachability questions (in this case to the question: Is I' reachable from I?). ■

This result enables us to verify SPDIs much more efficiently. It is important to note that model-checking an SPDI requires identification of simple loops, which means that the calculation of the semi-separatrices is not more expensive than the initial pass of the model-checking algorithm. Furthermore, we can perform this analysis only once for an SPDI and store the information to be used in any reachability analysis on that SPDI. Reduction, however, can only be applied once we know the source and destination states.

5 State-Space Reduction Using Kernels

5.1 State-Space Reduction Using Kernels

We have already shown that any invariant set is essentially a pair of semi-separatrices, and since the invariance kernel is an invariant set, we can use the results from section 4 to abstract an SPDI using invariance kernels. We now turn our attention to state space reduction using controllability kernels:

Definition 4. *Given an SPDI $\mathcal{S}$, a loop σ, a source edge $e0$ and a destination edge $e1$, an edge e is said to be* redundant *if it lies on the opposite side of a controllability kernel as both $e0$ and $e1$:*

$$\begin{aligned} \mathit{redundant}^{\sigma}_{e0\to e1} = & \{e : \mathcal{E} \mid \{e0,\ e1\} \subseteq \mathsf{Cntr}_{in}(\sigma) \cup \mathsf{Cntr}(\sigma) \wedge e \in \mathsf{Cntr}_{out}(\sigma)\} \\ & \cup\ \{e : \mathcal{E} \mid \{e0,\ e1\} \subseteq \mathsf{Cntr}_{out}(\sigma) \cup \mathsf{Cntr}(\sigma) \wedge e \in \mathsf{Cntr}_{in}(\sigma)\}. \end{aligned}$$

We can prove that we can do without these edges to check feasibility:

Lemma 3. *Given an SPDI $\mathcal{S}$, a loop σ, a source edge $e0$, a destination edge $e1$, and a feasible signature $e0\sigma e1$ then there exists a feasible signature $e0\sigma' e1$ such that σ' contains no redundant edge from $\mathit{redundant}^{\sigma}_{e0\to e1}$.* □

Given an SPDI, we can reduce the state space by discarding redundant edges.

Definition 5. *Given an SPDI $\mathcal{S}$, a loop σ, a source edge $e0$ and a destination edge $e1$, we define the reduced SPDI $\mathcal{S}^{\sigma}_{e0\to e1}$ to be the same as $\mathcal{S}$ but without redundant edges.*

Clearly, the resulting SPDI is smaller than the original one. Finally, based on proposition 3, we prove that reachability on the reduced SPDI is equivalent to reachability on the original one:

Theorem 7. *Given an SPDI $\mathcal{S}$, a loop σ, a source edge $e0$ and a destination edge $e1$, then, $e1$ is reachable from $e0$ in $\mathcal{S}$ if and only if $e1$ is reachable from $e0$ in $\mathcal{S}^{\sigma}_{e0\to e1}$.* □

Given a loop which has a controllability kernel, we can thus reduce the state space to explore. In practice, we apply this state space reduction for each controllability kernel in the SPDI. Once a loop in the SPDI is identified, it is straightforward to apply the reduction algorithm.

5.2 Immediate Answers to Reachability Questions

By definition of the controllability kernel, any two points inside it are mutually reachable. This can be used to answer reachability questions in which both the source and destination edge lie (possibly partially) within the same controllability kernel. Using proposition 2, we know that any point in the viability kernel of a loop can eventually reach the controllability kernel of the same loop, which allows us to relax the condition about the source edge to just check whether it (partially) lies within the viability kernel. Finally, we note that the union of non-disjoint controllability sets is itself a controllability set which allows us to extend the result to work for a collection of loops whose controllability kernels form a strongly connected set.

Definition 6. *We extend viability and controllability kernels for a set of loops Σ by taking the union of the kernels of the individual loops, with $\mathsf{Viab}(K_{\Sigma})$ being the union of all viability kernels of loops in Σ, and similarly $\mathsf{Cntr}(K_{\Sigma})$.*

Definition 7. *Two loops σ and σ' are said to be compatible ($\sigma \leftrightsquigarrow \sigma'$) if their controllability kernels overlap: $\mathsf{Cntr}(K_{\sigma}) \cap \mathsf{Cntr}(K_{\sigma'}) \neq \emptyset$.*

We extend the notion of compatibility to a set of loops Σ to mean that all loops in the set are transitively compatible: $\forall \sigma,\ \sigma' \in \Sigma \cdot \sigma \leftrightsquigarrow^{*} \sigma'$.

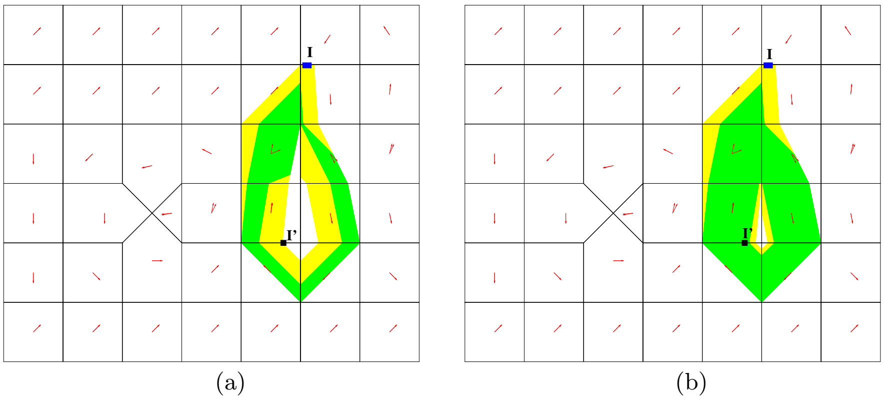


Fig. 6. Answering reachability using kernels

Based on proposition 2, we can prove the following:

Theorem 8. *Given a source edge e_{src} and a destination edge e_{dst}, if for some compatible set of loops Σ, $e_{src} \cap \mathsf{Viab}(K_\Sigma) \neq \emptyset$ and $e_{dst} \cap \mathsf{Cntr}(K_\Sigma) \neq \emptyset$, then e_{dst} is reachable from e_{src}.* □

Example 12. Fig. 6-(a) shows a viability and a controllability kernel of a cycle and two intervals I and I'. Whether I' is reachable from I cannot be answered immediately in this case, but Fig. 6-(b) shows the overlapping of the viability and controllability kernels depicted in Fig. 6-(a) with the kernels of an inner cycle. I' thus lies in a compatible controllability kernel, and we can immediately conclude (by theorem 8) that I' is reachable from I. ■

In practice, we propose to use these theorems to enable answering certain reachability questions without having to explore the complete state space. It can also be used to reduce reachability questions to (possibly) simpler ones by trying to reach a viability kernel rather than a particular edge. As in the case of semi-separatrices, a preliminary analysis of an SPDI's kernels be used in all subsequent reachability queries. SPeeDI [APSY02] starts by calculating and caching all loops in the given SPDI, and can thus easily identify maximal compatible sets of loops. Combining this technique with the semi-separatrix reduction technique we envisage substantial gains.

6 Concluding Remarks

We have introduced the concept of semi-separatrices for polygonal hybrid systems, and presented non-iterative algorithms to calculate them.

Using semi-separatrices and kernels, we presented techniques to improve reachability analysis on SPDIs. In all cases, the techniques require the identification and analysis of loops in the SPDI. When multiple reachability questions are to be asked about the same SPDI, this information can be gathered once to avoid repeated analysis. We note that most of this information is still required in reachability analysis, and thus no extra work is required to perform the optimization presented in this paper. The results presented all depend on checking whether an edge lies within a given polygon which can be efficiently checked using standard geometrical techniques frequently used in computer graphics such as using the odd-parity test [FvDFH96]. Sometimes, using kernel information, we can answer reachability questions without any further analysis. In other cases, we use semi-separatrices and controllability kernels to reduce the size of the SPDI.

Our work is obviously restricted to planar systems, which enables us to compute these kernels exactly. In higher dimensions and hybrid systems with higher complexity, calculation of kernels is not computable. Other work is thus based on calculations of approximations of these kernels (e.g., [ALQ$^+$01b, ALQ$^+$01a, SP02]). We are not aware of any work using kernels and semi-separatrices to reduce the state-space of the reachability graph as presented in this paper.

We have built a toolset SPeeDI [APSY02] for the analysis of SPDIs. We have recently extended this toolset to SPeeDI$^+$ [PS] which calculates kernels of SPDIs. We are currently exploring the implementation of the optimizations presented in this paper to improve the efficiency of SPeeDI$^+$. We are also investigating other applications of these kernels in the model-checking of SPDIs.

References

[AD94] R. Alur and D.L. Dill. A theory of timed automata. *Theoretical Computer Science*, 126:183–235, 1994.

[ALQ$^+$01a] J.-P. Aubin, J. Lygeros, M. Quincampoix, S. Sastry, and N. Seube. Towards a viability theory for hybrid systems. In *European Control Conference*, 2001.

[ALQ$^+$01b] J.-P. Aubin, J. Lygeros, M. Quincampoix, S. Sastry, and N. Seube. Viability and invariance kernels of impulse differential inclusions. In *Conference on Decision and Control*, volume 40 of *IEEE*, December 2001.

[APSY02] E. Asarin, G. Pace, G. Schneider, and S. Yovine. SPeeDI: a verification tool for polygonal hybrid systems. In *CAV'2002*, volume LNCS 2404, 2002.

[ASY01] E. Asarin, G. Schneider, and S. Yovine. On the decidability of reachability for planar differential inclusions. In *HSCC'2001*, volume LNCS 2034, 2001.

[ASY02] E. Asarin, G. Schneider, and S. Yovine. Towards computing phase portraits of polygonal differential inclusions. In *HSCC'02*. LNCS 2289, 2002.

[Aub01] J.-P. Aubin. The substratum of impulse and hybrid control systems. In *HSCC'01*, volume 2034 of *LNCS*, pages 105–118. Springer, 2001.

[DV95] A. Deshpande and P. Varaiya. Viable control of hybrid systems. In *Hybrid Systems II*, number 999 in LNCS, pages 128–147, 1995.

[FvDFH96] J. D. Foley, A. van Dam, S. K. Feiner, and J. F. Hughes. *Computer graphics (2nd ed. in C): principles and practice.* Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1996.

[HKPV95] T.A. Henzinger, P.W. Kopke, A. Puri, and P. Varaiya. What's decidable about hybrid automata? In *STOC'95*, pages 373–382. ACM Press, 1995.

[LPY01] G. Lafferriere, G. Pappas, and S. Yovine. Symbolic reachability computation of families of linear vector fields. *Journal of Symbolic Computation*, 32(3):231–253, September 2001.

[MP93] O. Maler and A. Pnueli. Reachability analysis of planar multi-linear systems. In *CAV'93*, pages 194–209. LNCS 697, Springer Verlag, July 1993.

[MS00] A. Matveev and A. Savkin. *Qualitative theory of hybrid dynamical systems*. Birkhäuser Boston, 2000.

[PS] G. Pace and G. Schneider. SPeeDI^{+}. `http:\\www.cs.um.edu.mt\speedi`.

[PS06] G. Pace and G. Schneider. Static analysis of SPDIs for state-space reduction. Technical Report 336, Department of Informatics, University of Oslo, April 2006.

[Sch02] G. Schneider. *Algorithmic Analysis of Polygonal Hybrid Systems*. PhD thesis, VERIMAG – UJF, Grenoble, France, July 2002.

[Sch04] G. Schneider. Computing invariance kernels of polygonal hybrid systems. *Nordic Journal of Computing*, 11(2):194–210, 2004.

[SJSL00] S. Simić, K. Johansson, S. Sastry, and J. Lygeros. Towards a geometric theory of hybrid systems. In *HSCC'00*, number 1790 in LNCS, 2000.

[SP02] P. Saint-Pierre. Hybrid kernels and capture basins for impulse constrained systems. In *HSCC'02*, volume 2289 of *LNCS*. Springer-Verlag, 2002.