Hybrid Semantics of Stochastic Programs with Dynamic Reconfiguration
Supplementary Material

Luca Bortolussi
Dept. of Mathematics and Informatics,
University of Trieste, Italy.
luca@dmi.units.it

Alberto Policriti
Dept. of Mathematics and Informatics,
University of Udine, Italy.
Istituto di Genomica Applicata, Udine, Italy.
policriti@dimi.uniud.it

We begin by reviewing a technique to approximate the dynamics of stochastic programs—written in a stochastic process algebra—by a hybrid system, suitable to capture a mixed discrete/continuous evolution. In a nutshell, the discrete dynamics is kept stochastic while the continuous evolution is given in terms of ODEs, and the overall technique, therefore, naturally associates a Piecewise Deterministic Markov Process with a stochastic program.

The specific contribution in this work consists in an increase of the flexibility of the translation scheme, obtained by allowing a *dynamic* reconfiguration of the degree of discreteness/continuity of the semantics.

We also discuss the relationships of this approach with other hybrid simulation strategies for biochemical systems.

1 Piecewise Deterministic Markov Processes

In order to formalize the dynamical evolution of Transition Driven Stochastic Hybrid Automata, we will map them to a class of stochastic processes known as Piecewise Deterministic Markov Processes (PDMP, [1]). They basically are Stochastic Hybrid Automata with a continuous dynamics based on ODE and a discrete and stochastic dynamics given by a Markov jump process. In the following we recall their definition.

Definition 1.1. A PDMP is a tuple $(Q, D, \mathscr{X}, \lambda, R)$, such that:

- Q is a finite set of *modes* or *discrete states*. We consider a function $d : Q \to \mathbb{N}$, assigning to each mode the dimension of its continuous state space, i.e. the number of variables defined in that mode. For each $q \in Q$, let $D_q \subset \mathbb{R}^{d(q)}$ be an open set, the continuous domain of mode q. Its boundary is denoted by ∂D_q and its closure by $\overline{D}_q$.
- D is the *hybrid state space*, defined as the disjoint union of D_q sets, namely $D = \bigcup_{q\in Q}\{q\} \times D_q$. A point $y \in D$ is thus a pair $y = (q, \mathbf{x})$, $\mathbf{x} \in D_q$. A subset of D, instead, is the disjoint union of subsets of D_q: $A = \bigcup_{q\in Q}\{q\} \times A_q$. For each D_q, we consider the Borel σ-algebra $\mathscr{D}_q$ obtained by restricting the standard Borel σ-algebra of $\mathbb{R}^{d(q)}$, and we define on D the σ-algebra $\mathscr{D} = \{A \in D \mid A = \bigcup_{q\in Q}\{q\} \times A_q,\ A_q \in \mathscr{D}_q\}$.
- To each mode $q \in Q$ we associate a vector field $\mathscr{X}_q : D_q \to \mathbb{R}^{d(q)}$, which is assumed to be *locally Lipschitz continuous*. The flow of such vector field is indicated by $\phi_q(t, \mathbf{x_0})$, denoting the point reached at time t starting from $\mathbf{x_0} \in D_q$. We assume that the flow of $\mathscr{X}_q$ is such that solutions do not diverge in finite time.

Submitted to:
CompMod 2009

- $\lambda : D \to \mathbb{R}^+$ is the *jump rate* and it gives the hazard of executing a discrete transition. It is required to satisfy the following property:

$$\forall y_0 = (q, \mathbf{x_0}) \in D,\ \exists \varepsilon(y_0) > 0 : t \mapsto \lambda(q, \phi_q(t, \mathbf{x_0})) \text{ is integrable in } [0, \varepsilon(y_0)[. \tag{1}$$

- $R : (D \cup \partial D) \times \mathscr{D} \to [0,1]$ is the *transition measure* or *reset kernel*. It maps each $y \in D \cup \partial D$ on a probability measure on $(D, \mathscr{D})$, and it must satisfy:

$$\text{for each } A \in \mathscr{D}, \text{ the function } y \mapsto R(y, A) \text{ is measurable;} \tag{2}$$

$$R(y, \{y\}) = 0, \text{ for each } y \in D \tag{3}$$

The idea of PDMP is that, within each mode q, the process evolves according to the differential equation given by the vector field $\mathscr{X}_q$. While in a mode, the process can jump spontaneously with hazard given by the rate function λ. Moreover, a jump is immediately performed whenever the boundary of the state space of the current mode is hit. In order to capture the evolution, we need to define the sequence of jump times and target states of the PDMP, given by random variables $T_1, Z_1, T_2, Z_2, \ldots$. Let

$$t_*(y) = \begin{cases} \inf\{t > 0 \mid \phi_q(t, \mathbf{x}) \in \partial D_q\} \\ \infty, \text{ if no such time exists,} \end{cases}$$

be the hitting time of the boundary ∂D_q starting from $y \in D$. We can define the survivor function of the first jump time T_1, given that the process started at $y = (q, \mathbf{x})$, by[1]

$$F(t, y) = \mathbb{P}(T_1 \geq t) = I_{t < t_*(y)} \exp\left(-\int_0^t \lambda(q, \phi_q(s, \mathbf{x}))\mathrm{d}s\right).$$

This defines the probability distribution of the first jump time T_1, which can be sampled using standard Montecarlo simulation techniques, i.e. by solving for t the equation $F(t, y) = U$, with U uniform random variable in $[0,1]$. Once the time of the first jump has been drawn, we can compute the target point Z_1 of the reset map by sampling from the distribution $R(y^-_{T_1}, \cdot)$, with $y^-_{T_1} = \phi_q(T_1, \mathbf{x})$. From Z_1, the process follows the solution of the vector field, until the next jump, determined by the same mechanism presented above.

A further requirement is that, letting $N_t = \sum_k I_{t > T_k}$ be the r.v. counting the number of jumps up to time t, it holds that N_t is finite with probability 1, i.e.

$$\forall t, \mathbb{E}N_t < \infty. \tag{4}$$

This condition is enforced to rule out pathologic behaviors like Zeno trajectories. Indeed, problems may arise by the interaction between the reset kernel and the hitting of boundaries, see [1] for further details.

1.1 Mapping TDSHA to PDMP

A TDSHA $\mathscr{T} = (Q, \mathbf{X}, \mathfrak{TC}, \mathfrak{TD}, \mathfrak{TS}, init)$ can be mapped quite straightforwardly into a PDMP $(Q, D, \mathscr{X}, \lambda, R)$. The recipe is the following:

[1] We remind to the reader that I_φ stands for the *indicator* function, whose value is 1 if condition φ holds and 0 otherwise.

- Q is the set of discrete modes, the dimension function $d : Q \to \mathbb{N}$ is given by $d(q) = n = |\mathbf{X}|$, and the domain within each mode is given by

$$D_q = \bigcap_{\delta \in \mathfrak{TD}} G_\delta^c,$$

  i.e. by intersecting the complement of each activation set G_δ. Note that D_q is open.
- The vector field in mode q in constructed from continuous transitions:

$$\mathscr{X}_q(\mathbf{x}) = \sum_{\tau \in \mathfrak{TC} \mid \mathbf{cmode}[\tau]=q} \mathbf{stoich}[\tau] \cdot \mathbf{rate}[\tau](\mathbf{x}).$$

- The rate function λ is defined as

$$\lambda(q,\mathbf{x}) = \sum_{\eta \in \mathfrak{TS} \mid \mathbf{e_1}[\eta]=q} \lambda(\eta, q, \mathbf{x}),$$

  where $\lambda(\eta, q, \mathbf{x}) = I_{\mathbf{guard}[\eta](\mathbf{x})} \mathbf{rate}[\eta](\mathbf{x})$.
- The transition measure R is defined according to whether we are in D_q or in ∂D_q. For $\mathbf{x} \in D_q$ and $A \in \mathscr{D}$, we put

$$R((q,\mathbf{x}),A) = \sum_{\eta \in \mathfrak{TS} \mid \mathbf{e_1}[\eta]=q} \frac{\lambda(\eta,q,\mathbf{x})}{\lambda(q,\mathbf{x})} \delta_{(\mathbf{e_2}[\eta], f(\mathbf{x}))}(A),$$

  where $\mathbf{reset}[\eta]$ is of the form $\mathbf{X}' = f(\mathbf{X}) \in D_{\mathbf{e_2}[\eta]}$ and $\delta_{(q,\mathbf{x})}(A)$ is the Dirac measure on the point $(q,\mathbf{x}) \in D$, assigning probability 1 to $(q,\mathbf{x})$ and 0 to the rest of the space.
  If $\mathbf{x} \in \partial D_q$ and $A \in \mathscr{D}$, then

$$R((q,\mathbf{x}),A) = \sum_{\delta \in \mathfrak{TD}(q,\mathbf{x})} \frac{\mathbf{priority}[\delta]}{\mathbf{priority}[\mathfrak{TD}(q,\mathbf{x})]} \delta_{(\mathbf{e_2}[\delta], f(\mathbf{x}))}(A),$$

  where $\mathfrak{TD}_q = \{\delta \in \mathfrak{TD} \mid \mathbf{e_1}[\delta] = q,\ \mathbf{guard}[\delta](\mathbf{x}) = true\}$ and $\mathbf{priority}[\mathfrak{TD}(q,\mathbf{x})] = \sum_{\delta \in \mathfrak{TD}(q,\mathbf{x})} \mathbf{priority}[\delta]$.

The rationale of such mapping is quite simple. The vector field is specified by adding the effect of continuous transitions (corresponding to different flows) on each variable. Instantaneous transitions are encoded by restricting the state space, so that when their guard becomes true, the PDMP hits the boundary and is forced to jump. Finally, rates of stochastic transitions are added together pointwise to determine the total jump rate (guards are dealt by introducing their indicator functions as factor for rates). The choice of the specific stochastic transition firing, instead, is done proportionally to its rate at the firing time. Similarly, the choice among different active transitions is performed proportionally to their priority.

The previous recipe, however, works under the following assumption:

Assumption 1. Resets of discrete jumps (either instantaneous or stochastic) cannot reach a point outside the hybrid state space, i.e. a point $(q,\mathbf{x}) \notin D_q$.

This guarantees that there cannot be sequences of jumps firing precisely at the same time. The "static" semantic of **sCCP**, presented in Section 4 of the paper, satisfy this assumption trivially, as there are no instantaneous transitions. This is no more true in the case of dynamic partitioning. In order to relax Assumption 1, we need to modify the mapping from TDSHA to PDMP, cf. below.

Remark 1.1. In the previous discussion we implicitly assumed that the PDMP derived at the end of this process satisfies the conditions (1)-(4) of PDMP. In order to make them hold we can require that the rate and reset functions of TDSHA are sufficiently smooth, which is generally the case. The most critical condition is hypothesis (4), which can be violated by subtle interactions among instantaneous transitions. However, TDSHA defined in Section 2 of the paper have no instantaneous transitions. In this case, a sufficient condition for (4) to hold is that rate functions are bounded [1]. Standard rate functions used in systems biology, however, are not bounded in $\mathbb{R}^n$ (think of a mass action rate). One way to enforce such condition is letting the system evolve in a bounded domain, sufficiently large that points outside it have no physical significance. Technically, we can do this by introducing suitable instantaneous transitions in the TDSHA obtained from an **sCCP** program. First, we need to introduce a new mode $\Delta \in Q$ in which the rate and the vector field are identically zero. Then the new instantaneous transitions will make the system jump from the boundary of the bounded domain into this new state Δ, with identity reset. Staten otherwise, we stop all trajectories leaving the allowed domain. Note that such trick is standard in PDMP, although generally used to stop the evolution at a finite time horizon (Δ is known as *cemetery point*).

We turn now to generalize the mapping from TDSHA to PDMP so to relax Assumption 1. The only differences w.r.t. the previous definition is in the reset kernel R.

Suppose to be in a point $(q,\mathbf{x})$, $\mathbf{x} \notin D_q$, where $\mathbf{x}$ is not necessarily a point of the border of D_q. Define the set $\mathfrak{TD}(q,\mathbf{x}) = \{\delta \in \mathfrak{TD} \mid \mathbf{e_1}[\delta] = q \wedge \mathbf{x} \in G_\delta\}$. Clearly, $\mathbf{x} \notin D_q$ implies $\mathfrak{TD}(q,\mathbf{x}) \neq \emptyset$.
We will define now the set $\mathfrak{T}(q,\mathbf{x})$ of executable sequences of instantaneous transitions. First, however, we need some notation. Given a discrete transition $\delta \in \mathfrak{TD}$, we indicate with $r(\delta,\mathbf{x})$ the point to which $\mathbf{x}$ is mapped by $\mathbf{reset}[\delta]$. Moreover, let $\cdot$ denote the concatenation of sequences, defined for sequences and set of sequences in the standard way.

Definition 1.2. The set $\mathfrak{T}(q,\mathbf{x})$ of executable sequences of instantaneous transitions from the point $(q,\mathbf{x})$ is defined by

$$\mathfrak{T}(q,\mathbf{x}) = \bigcup_{\delta \in \mathfrak{TD}(q,\mathbf{x})} \delta \cdot \mathfrak{T}(\mathbf{e_2}[\delta], r(\delta,\mathbf{x})).$$

Note that this recursive definition may generate infinite sequences of transitions, as the following example shows:

Example 1.1. Consider a TDSHA with one mode q and one variable X and two instantaneous transitions, $\mathfrak{TD} = \{\delta_1, \delta_2\}$, with $G_{\delta_1} = [1,2[$, $G_{\delta_2} = [2,\infty[$, $r(\delta_1,x) = x+1$, and $r(\delta_2,x) = x-1$. Then $\mathfrak{T}(q,1)$ contains the single sequence of infinite length $\delta_1\delta_2\delta_1\delta_2\ldots$.

However, we consider such behaviors as pathological, hence we rule them out.

Definition 1.3. The set $\mathfrak{TD}$ of instantaneous transitions is *well-behaved* if

$$\forall (q,\mathbf{x}) \in D^c, \forall \alpha \in \mathfrak{T}(q,\mathbf{x}), |\alpha| < \infty.$$

This is the condition we will require on $\mathfrak{TD}$ for the mapping to PDMP to work.
The following proposition is straightforward, due to finiteness of $\mathfrak{TD}$:

Proposition 1.1. *If $\mathfrak{TD}$ is well-behaved, then*

$$\forall (q,\mathbf{x}) |\mathfrak{T}(q,\mathbf{x})| < \infty.$$

If $|\mathfrak{T}(q,\mathbf{x})| < \infty$, then we can define on it a probability measure p as follows:

1. for $\delta \in \mathfrak{TD}(q,\mathbf{x})$, $p(\delta) = \frac{\mathbf{priority}[\delta]}{\mathbf{priority}[\mathfrak{TD}(q,\mathbf{x})]}$;

2. for $\alpha = \delta \cdot \alpha'$, $p(\alpha) = p(\delta)p(\alpha')$.

The fact that p is a probability measure on $\mathfrak{T}(q,\mathbf{x})$ follows from the fact that $\sum_{\delta \in \mathfrak{T}\mathfrak{D}(q,\mathbf{x})} p(\delta) = 1$.

We can now define the reset kernel in a point $\mathbf{x} \in D_q^c$, for a well behaved set of instantaneous transitions. Let $\alpha \in \mathfrak{T}(q,\mathbf{x})$, $\alpha = \delta_1 \cdots \delta_k$. Note that $\mathbf{e_1}[\delta_i] = \mathbf{e_2}[\delta_{i-1}]$, and set $\mathbf{e_2}[\alpha] = \mathbf{e_2}[\delta_k]$. Then, define the point $\bar{r}(\alpha,\mathbf{x})$ reached by α starting from $\mathbf{x}$ recursively as:

$$\bar{r}(\delta,\mathbf{x}) = r(\delta,\mathbf{x}); \quad \bar{r}(\delta_1 \cdots \delta_k,\mathbf{x}) = r(\delta_k, \bar{r}(\delta_1 \cdots \delta_{k-1},\mathbf{x})).$$

The reset kernel in the point $(q,\mathbf{x})$ is thus

$$R((q,\mathbf{x}),A) = \sum_{\alpha \in \mathfrak{T}(q,\mathbf{x})} p(\alpha)\delta_{(\mathbf{e_2}[\alpha],\bar{r}(\alpha,\mathbf{x}))}(A).$$

Therefore, each possible sequence α of discrete transitions is chosen according to its probability $p(\alpha)$.

We still need to deal with jumps of stochastic transitions leading out of the hybrid state space. The idea is simple: if we jump out of the allowed region, we immediately apply a sequence of discrete transitions. Formally, let $\eta \in \mathfrak{T}\mathfrak{S}(q,\mathbf{x})$, where $\mathfrak{T}\mathfrak{S}(q,\mathbf{x}) = \{\eta \in \mathfrak{T}\mathfrak{S} \mid \mathbf{e_1}[\eta] = q \wedge \mathbf{guard}[\eta](\mathbf{x})\}$, and suppose $r(\eta,\mathbf{x}) \notin D$. Let $\mathbf{e_2}[\eta] = q_\eta$ and $r(\eta,\mathbf{x}) = \mathbf{x}_\eta$, and consider $\mathfrak{T}\mathfrak{D}(q_\eta,\mathbf{x}_\eta) \neq \emptyset$ (as $\mathbf{x}_\eta \notin D_{q_\eta}$). We can now simply modify the reset kernel $R((q,\mathbf{x}),A)$ by replacing $\delta_{q_\eta,\mathbf{x}_\eta}(A)$ with $R((q_\eta,\mathbf{x}_\eta),A)$. Therefore $R((q,\mathbf{x}),A)$ becomes

$$\begin{aligned} R((q,\mathbf{x}),A) \quad &= \quad \sum_{\eta \in \mathfrak{T}\mathfrak{S}(q,\mathbf{x}) \mid (\mathbf{e_2}[\eta], r(\eta,\mathbf{x})) \in D} \frac{\lambda(\eta,q,\mathbf{x})}{\lambda(q,\mathbf{x})} \delta_{(\mathbf{e_2}[\eta],r(\eta,\mathbf{x}))}(A) \\ &+ \quad \sum_{\eta \in \mathfrak{T}\mathfrak{S}(q,\mathbf{x}) \mid (\mathbf{e_2}[\eta], r(\eta,\mathbf{x})) \notin D} \frac{\lambda(\eta,q,\mathbf{x})}{\lambda(q,\mathbf{x})} R((q_\eta,\mathbf{x}_\eta),A). \end{aligned}$$

Consider now the TDSHA constructed according to the dynamic partitioning scheme introduced in Section 5 of the paper. It is easy to see that such TDSHA may not satisfy Assumption 1 (for instance, a stochastic jump may trigger a reconfiguration of the partitioning). However, it is easy to show that the set of instantaneous transitions is well behaved. Consider an **sCCP** action e, initially dealt as discrete. According to the discussion of Section 5, the action will be moved in the set of continuously-approximated actions as soon as a certain function $f_e(\mathbf{x})$ reaches the value ε from below. The function depends only on **sCCP** store variables $\mathbf{X}$, which are not reset by the instantaneous transition. Hence, soon after having switched e from discrete to continuous, $f_e(\mathbf{x})$ will still be equal to ε. Now, the condition for bringing e back in the discrete set is $f_e(\mathbf{x}) \leq -\varepsilon$, which is obviously false after the switching. Hence, if an **sCCP**-edge changes status, it cannot be immediately changed back to its original status. Hence it is not possible to execute an infinite sequence of instantaneous transitions. Thus the following theorem holds.

Theorem 1.1. *Let $\mathscr{A} = (A, \mathscr{D}, \mathbf{X}, init)$ be an **sCCP** program and let $\mathscr{T}(C, cont, disc) = (Q, \mathbf{Y}, \mathfrak{T}\mathfrak{C}, \mathfrak{T}\mathfrak{D}, \mathfrak{T}\mathfrak{S}, init)$ be the TDSHA with dynamic partitioning with policies given by predicates cont and disc. Then its set of discrete transitions $\mathfrak{T}\mathfrak{D}$ is well-behaved.*

References

[1] M.H.A. Davis (1993): *Markov Models and Optimization*. Chapman & Hall.