Comparison of Hybrid Systems and Fluid Stochastic Petri Nets *

Bruno Tuffin , Dong S. Chen and Kishor S. Trivedi†
Center for Advanced Computing and Communication
Department of Electrical and Computer Engineering
Duke University, Durham, NC 27708-0291, U.S.A.

Abstract. Hybrid Systems are models of interacting digital and continuous devices with applications in the control of aircraft, computers, or modern cars for instance. Concurrently, Fluid Stochastic Petri Nets (FSPNs) have been introduced as an extension of stochastic Petri nets to model physical systems that contain continuous fluid quantities controlled by discrete logic, as well as a means to deal with the possible state explosion of discrete models. Of course, an FSPN is by definition a hybrid system but in this paper, we compare a general hybrid systems formalism with FSPN formalisms. The conclusion of this paper, suplemented by some examples, is that fluid stochastic Petri nets can be expressed by the very general hybrid systems formalism. Conversely, a lot of usual hybrid systems can be expressed and solved using the fluid stochastic Petri nets theory, thus exploiting advantages they offer in dealing with concurrency and complexity issues.

Keywords: Modeling, Stochastic Petri Nets, Fluid Models, Hybrid Systems

1. Introduction

Stochastic Petri Nets (SPNs) of various type provide a powerful paradigm to analyze the performance and availability of complex systems (Ajmone Marsan et al., 1995; Ciardo et al., 1993; Peterson, 1981; Reisig, 1985). More recently, Fluid Stochastic Petri Nets (FSPNs) (Ciardo et al., 1999; Horton et al., 1998) have generalized SPNs to circumvent the state explosion as well as to deal more easily with the huge number of events likely to occur even in a short period of time (like in ATM networks for example). An extended literature has been developped to deal with these models. Note also that continuous Petri nets have been introduced by other authors (see (Alla and David, 1998; David, 1997) and the references included) to deal with the same problems.

On the other hand, Hybrid Systems (see for instance (Alur et al., 1991; Alur and Dill, 1994; Henzinger et al., 1995; Puri and Varaiya,

* Work supported in part by the National Science Foundation under Grants EEC-9980591 and EEC-9714965.

† Corresponding author. E-mail: kst@ee.duke.edu, Tel:+1 919 660 5269 , Fax:+1 919 660 5293.

1994; Varaiya, 1999)), also called Hybrid Automata, are models consisting of interacting digital and continuous devices, where the continuous variables of a plant are controlled and supervised by the digital controller. These models have been of interest in, for instance, aircraft control, manufacturing, communication networks, intelligent systems and so forth. The usual issues addressed by these models are stability, controllability, and observability.

By definition, an FSPN is a Hybrid System (HS), as it handles both fluid and discrete places. In this paper, we consider the general formalism of HS described in (Varaiya, 1999) and see if FSPNs fit this formalism. We also examine what kinds of HS can be represented by FSPNs. PNs have some advantages with respect to the usual HS models of control theory such as its ability to easily handle concurrency, to reduce the complexity of the model and to automatically generate the underlying stochastic process from the concise high level description. The aim of this paper is then to compare both formalisms, to see if either an FSPN can be represented by an HS (as described in (Varaiya, 1999)) or conversely if an HS can be expressed by an FSPN. Then we would be able to use the solution methods of one formalism on the other. Note that the question of comparing Petri nets and state automata models (both untimed) has been addressed in (Cassandras, 1993, page 104). The conclusion therein is that the best model depends on the particular application considered, and that the two modeling approaches are complementary.

This paper is organized as follows. We define FSPNs in Section 2 and hybrid systems in Section 3. Then we study in Section 4 if FSPNs fit the general formalism of hybrid systems introduced in Section 3 and we examine in Section 5 HS can be represented by FSPNs. These two Sections are complemented by examples. Finally, we conclude in Section 6.

2. Fluid Stochastic Petri Nets

An FSPN is defined by a 13-tuple (Ciardo et al., 1999; wm:, ; Horton et al., 1998)

$$(\mathcal{P}, \mathcal{T}, a, f, g, >, d, r, F, \omega, b, m^0, x^0)$$

where

- $\mathcal{P}$, the set of places, is partitioned into $\mathcal{P}_D$, the set of D discrete places, and $\mathcal{P}_C$, the set of C continuous (or fluid) places. Let (x, m) denote the marking of the FSPN, with vector x (resp. vector m) giving the content of the fluid (resp. discrete) places. Let $\mathcal{S}$ be the state space of the FSPN. We assume that the total number of tokens in discrete places is bounded.

- $\mathcal{T}$ is a set of transitions which can be partitioned in two subsets: $\mathcal{T}_T$, the set of timed transitions and $\mathcal{T}_I$, the set of immediate transitions.

- a describes the marking-dependent cardinality or fluid impulse of the input and output arcs connecting transitions and places. The function a is defined from $((\mathcal{P}_D \times \mathcal{T}) \cup (\mathcal{T} \times \mathcal{P}_D)) \times \mathcal{S}$ to $I\!N$ for discrete places and from $((\mathcal{P}_C \times \mathcal{T}) \cup (\mathcal{T} \times \mathcal{P}_C)) \times \mathcal{S}$ to $I\!R^+$ (set of non-negative reals) for fluid places.

- $f : ((\mathcal{P}_C \times \mathcal{T}) \cup (\mathcal{T} \times \mathcal{P}_C)) \times \mathcal{S} \to I\!R^+$ is the marking-dependent fluid rate of the arcs connecting transitions and fluid places.

- $g : \mathcal{T} \times \mathcal{S} \to \{0, 1\}$ is the guard of each transition (if no guard function is defined, the default value is $g = 1$).

- $>$ defines the priority for the firing of each transition.

- d defines the firing time distribution for each transition. It can be marking dependent. Practically, each transition has its own clock $c(t)$ activated when the transition becomes enabled, sampled from distribution given by d and giving the time to spend before its firing.

- r is the static resampling policy for each transition when it becomes disabled by the firing of a competitive transition and later becomes enabled again. It can be for instance PRI (preemptive repeat identical), PRD (preemptive repeat different) or PRS (preemptive resume) (Gaver, 1962; Marie and Trivedi, 1987).

- F defines the resampling policy for each transition when another transition fires but the considered transition remains enabled.

- $\omega : \mathcal{T} \times \mathcal{S} \to I\!R^+_* = I\!R^+ - \{0\}$ is the weight function for the transitions.

- $b : \mathcal{P}_C \times \mathcal{S} \to I\!R^+_*$ is the bound on each continuous place.

- (x^0, m^0) is the initial marking, vector x^0 giving the marking of fluid places and the vector m^0 the marking of discrete places.

We say that a transition $t \in \mathcal{T}$ has concession in marking (x, m) if

$$\forall p \in \mathcal{P}_D, a_{p,t}(x, m) \leq m_p \text{ and } g_t(x, m) = 1.$$

An immediate transition which has concession in marking (x, m) is enabled in marking (x, m). On the other hand, a timed transition which has concession in marking (x, m) is enabled if no immediate or a higher priority timed transition has concession in (x, m). Let $\mathcal{E}(x, m)$ denote the set of enabled transitions in (x, m).

Next we describe the dynamics of an FSPN. Suppose the FSPN is in marking (or state) (x, m).

Then fluid continuously flows through the arcs of enabled transitions into fluid places. The potential rate of change of level in marking (x, m) is given by

$$\delta_i^{pot}(x, m) = \sum_{t \in \mathcal{E}(x,m)} f_{t,i}(m, x) - f_{i,t}(x, m).$$

The actual rate of change of level is then

$$\frac{dx_i}{dt} = \begin{cases} 0 & \text{if } (x_i = 0 \text{ and } \delta_i^{pot}(x, m) \leq 0) \text{ or} \\ & (x_i = b_i(m) \text{ and } \delta_i^{pot}(x, m) \geq 0) \\ \delta_i^{pot}(x, m) & \text{otherwise,} \end{cases}$$

where $b_i(m)$ is the fluid bound when the i^{th} continuous place in discrete marking is m.

After a time given by distribution d and priority $>$, a transition $t \in \mathcal{E}(x, m)$ may fire, leading to a new marking (x', m') given by

$$\begin{aligned} \forall p \in \mathcal{P}_D,\ m'_p &= m_p + a_{t,p}(x, m) - a_{p,t}(x, m) \\ \forall i \in \mathcal{P}_C,\ x'_i &= \min(b_i(m'), \max(0, x_i + a_{t,i}(x, m) - a_{i,t}(x, m))). \end{aligned} \quad (1)$$

Immediately after reaching this new marking, the set of enabled transitions and the firing times are reinitialized according to $>$, d, r and F. When two or more immediate transitions are enabled at the same time, transition u is chosen to fire with probability

$$\frac{\omega_u(x, m)}{\sum_{u'} \omega_{u'}(x, m)}.$$

Then again, fluid flows in fluid places, until a new transition fires, i.e., a new discrete event occurs, and so forth.

The FSPN defined above is the one of (Ciardo et al., 1999). In (Horton et al., 1998), the model involves several restrictions. First, fluid impulses in or from fluid places are forbidden. The next restriction is about the firing times of transitions. Only immediate and exponential firing times are allowed. Moreover, priorities are not considered and only PRD and PRS policies are respectively allowed for the resampling and affected relations. The general model described here is solved by simulation. If we restrict it like in (Horton et al., 1998), we can more easily derive the differential equations and solve the model by means of numerical analysis.

As pointed in (Ciardo et al., 1999), FSPNs (as well as SPNs) may exhibit some unstable behaviours. It can happen for instance when an immediate transition keeps adding a token (resp. a fluid impulse) in a discrete (resp. fluid) place and another immediate transition keeps removing it immediately (see (Ciardo et al., 1999) for more precise details on the possible unstable problems). These forbidden behaviours will not be addressed in the sequel as it can easily be proved that to a (forbidden) unstable behaviour of an FSPN correspond also a (forbidden) unstable behaviour of the equivalent hybrid system.

3. Hybrid systems

A hybrid system consists of two interacting components: one is continuous, $z \in \mathbb{R}^n$, and the other one is event-driven on a finite state space, $q \in \mathcal{Q}$ (Varaiya, 1999).

A general model of hybrid system often studied is the following: suppose that the system starts from state (z_0, q_0). Then, during a time interval $[t_0, t_1)$, just the continuous part of the state evolves according to the differential inclusion

$$\frac{dz}{dt}(t) \in F(z(t), q_0) \subset \mathbb{R}^n.$$

The time t_1 is defined as the first time for which a "guard" (or "enabling zone") $G(q, q') \subset \mathbb{R}^n$ verifies

$$z(t_1-) \in G(q_0, q_1)$$

for some q_1 (Of course $G(q, q') = \emptyset$ when it is not possible to go from q to q'). Then the discrete part of the state moves instantaneously from q_0 to q_1 and the continuous state is reset from $z(t_1-)$ to z_1 following the "reset" relation

$$(z(t_1-), z_1) \in R(q_0, q_1) \subset \mathbb{R}^n \times \mathbb{R}^n,$$

R being defined for each (q, q') pair. If another guard is satisfied following the reset, we can have another discrete transition (Varaiya, 1999).

Then we continue the system trajectory making the continuous part of state evolve according to $\frac{dz}{dt}(t) \in F(z(t), q_1)$ up to t_2 and replacing t_0, z_0, q_0, t_1, z_1 and q_1 by respectively t_1, z_1, q_1, t_2, z_2 and q_2, and so on. This system can evolve randomly as several differential inclusions may be possible and different guards and resets may be satisfied. We will see in Section 4 which random assumptions are needed to include FSPNs in this formalism.

Note that in *timed automata* (Alur and Dill, 1994), the components z_i of z represent clocks so that $\frac{dz_i}{dt}(t) = 1\ \forall t$.

In (Varaiya, 1999) and the references included, the main focus is to know if a hybrid system is decidable (i.e., if its reachability set at a time t can be determined by a finite algorithm). Some restrictions are given for which the answer is proven to be yes (see also (Henzinger et al., 1995; Puri and Varaiya, 1994)). Paper (Varaiya, 1999) also contains a description of a simulator of a network of hybrid systems that can interact in some ways. At first sight, this last point does not seem to be interesting for our comparison with FSPN formalism as an FSPN can be compared with a single hybrid system.

4. Can hybrid systems represent FSPNs?

Consider an FSPN consisting of D discrete places, C continuous places and T transitions. To represent this FSPN in the hybrid systems formalism, we need to include both the contents of fluid places and clocks of transitions in the continuous part of the state, so we set

$$z(t) = (x_1(t), \cdots, x_C(t), c_1(t), \cdots, c_T(t))$$

where $x_i(t)$, $1 \le i \le C$, is the content of the i^{th} continuous place and $c_i(t)$, $1 \le i \le C$, is the remaining time before (potential) firing for the i^{th} transition. Moreover, the discrete part q of the state (z, q) of the HS is given by the discrete marking m of the FSPN.

We now need to define appropriately the set F, the HS guard G and the reset relation R and to complete the modeling of FSPNs. Assume that the HS is in state $(z(0), q(0)) = (z_0, q_0)$ at $t = 0$ as just defined above (we assume here that the firing times of the enabled FSPN transitions are already sampled). In the interval of time $[0, t_1)$, the continuous state evolves according to $\frac{dz}{dt}(t) \in F(z(t), q_0),\ z(0) = z_0$, so that we need to have that

- the last T coordinates of each state of $F(z, q)$ (we replace $z(t)$ by z to simplify the notations) are equal to -1 for enabled transitions and 0 otherwise, this for all (z, q), i.e., each transition clock is going at the same speed as the global clock (and the rates are negative since we are keeping track of the remaining firing time).

- The differential inclusion for the first C coordinates of z is driven by the following relation

$$\frac{dz_i}{dt} = \begin{cases} 0 & \text{if } (z_i = 0 \text{ and } \delta_i^{pot}(x,q) \leq 0) \text{ or} \\ & (z_i = b_i(q) \text{ and } \delta_i^{pot}(x,q) \geq 0) \\ \delta_i^{pot}(x,q) & \text{otherwise} \end{cases}$$

  where x is the restriction of z to its C first coordinates and $\delta_i^{pot}(x,q)$ is defined in Section 2.

- Note that in this case $F(z,q)$ is composed of only one state (there is no randomness in this definition of F) and that the rate of change over time t may depend not only on the fluid and discrete markings, but also on the clocks of the transitions.

The time of first discrete state change t_1 from q_0 to q_1 is given by the HS guard G (take care not to mix up the term "guard" in hybrid systems theory with the term "guard" in Petri nets theory)

- We define G such that $z(t) \in G(q,q')$ (which means that the discrete state goes from q to q' when $z(t)$ reaches $G(q,q')$) if a transition fires. Then there are at most T possible guards from a discrete state q. Calling for simplicity $G(q,q')$ now $G(q,i)$ if it corresponds to a firing of transition i, we have $z(t_1-) \in G(q,i)$ if $c_i(t_1-) = 0+$, i.e., if the remaining firing time of transition i becomes zero.

- The possibility of dependence of G on the fluid places is the same as in (Ciardo et al., 1999), but the expression changes: $z(t) \in G(q,q')$(in hybrid systems) means for this dependence that the discrete transition is possible if the fluid levels are in some area (this is a guard in the Petri nets terminology, which is rather expressed in this case by $i \in G(x(t),q)$), and if the remaining firing time of the transition i from q to q' is $0+$. So the $G(q,i)$ are sets of type

$$X(q,i) \times \bar{\mathbb{R}}^+ \times \cdots \times \bar{\mathbb{R}}^+ \times \{0\} \times \bar{\mathbb{R}}^+ \times \cdots \times \bar{\mathbb{R}}^+$$

  where $\bar{\mathbb{R}}^+ = \mathbb{R}^+ \cup \{\infty\}$, the $\{0\}$ is placed after $i-1$ set $\mathbb{R}$ and $X(q,i)$ is the guard on fluid places when the discrete place is q ($X(q,i) = \emptyset$ when transition i is not allowed).

- Note here that some randomness can occur in an HS when several guards are satisfied at the same time (i.e., several transitions can fire at the same time in the Petri net). We choose one transition to fire according to the weight associated with each guard (corresponding to the weight of the transition to be fired).

After an HS guard is satisfied, i.e, a transition has fired, we look at the reset relation R which

- can put instantaneously a given amount of fluid (depending on q_0, q_1 and $z(t_1-)$ as the relation is $(z(t_1-), z_1) \in R(q_0, q_1)$) in some places, leading from $z_i(t_1-) = x_i(t_1-)$ to $z_i(t_i) = x_i(t_1) = x_{i,1}$ for the C first coordinates of $z(t)$ as in FSPNs when a transition has fired.

- This also reinitializes (randomly for the clocks of non-constant transitions) the remaining firing times, i.e., the last T coordinates of $z(t)$ depending on their discipline (note that this formalism can incorporate a large number of disciplines: preemptive (Gaver, 1962; Marie and Trivedi, 1987)...). The remaining firing time of the non-enabled transitions are set to infinity.

- The reason why we take for the clocks the remaining firing time (so the reverse time order) is that, if we had taken them in the increasing manner, the firing times should have been decided by the guard $G(q, q')$ but it is difficult to model this randomness in the current hybrid system formalism. Moreover, it allows us to make the firing times dependent of q_0, q_1 but also of $z(t_1-)$ (then of z_1), as the reset relation is then $(z(t_1-), z_1) \in R(q_0, q_1)$.

Another HS guard may be satisfied following the reset, it is typically the case of immediate transitions, and we process it as before.

Then we continue the trajectory after time t_1 in the same way as did at time 0.

As described in (Ciardo et al., 1999), the general behavior of FSPNs requires the user to solve systems of ordinary differential equations at each step. This can be costly so we use an FSPN discretized on the behavior of the original FSPN with respect to the continuous component x (consisting of the C first cordinates of z). These discretized FSPNs can be represented by linear hybrid systems, but the randomness on the firing of transitions is usually not included in hybrid systems packages.

EXAMPLE 1. *The FSPN model of Figure 1 is for a dual tank system analyzed completely in (*Ciardo et al., 1999*). It is a processing plant where a fluid flows with rate $r_{in} = 0.95$ in a tank "One" of maximal capacity $b_1 = 1.0$. The liquid from tank "One" is used by a processing station with a rate $r_{out} = 1.0 > r_{in}$. The station is subject to breakdowns and repairs (represented by transitions "Fail" and "Repair" and exponentially distributed with rate $\lambda = 0.1$ and $\mu = 1.0$). During a breakdown, the station is not fuelled and the liquid from the external source is no more directed to tank "One", but to an additional tank*

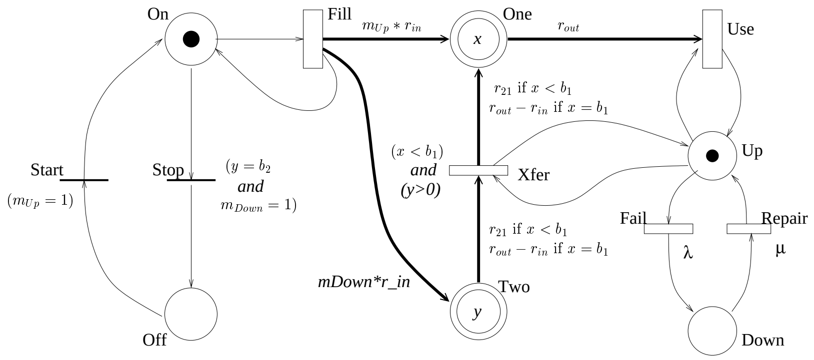

Figure 1. FSPN model of a dual tank

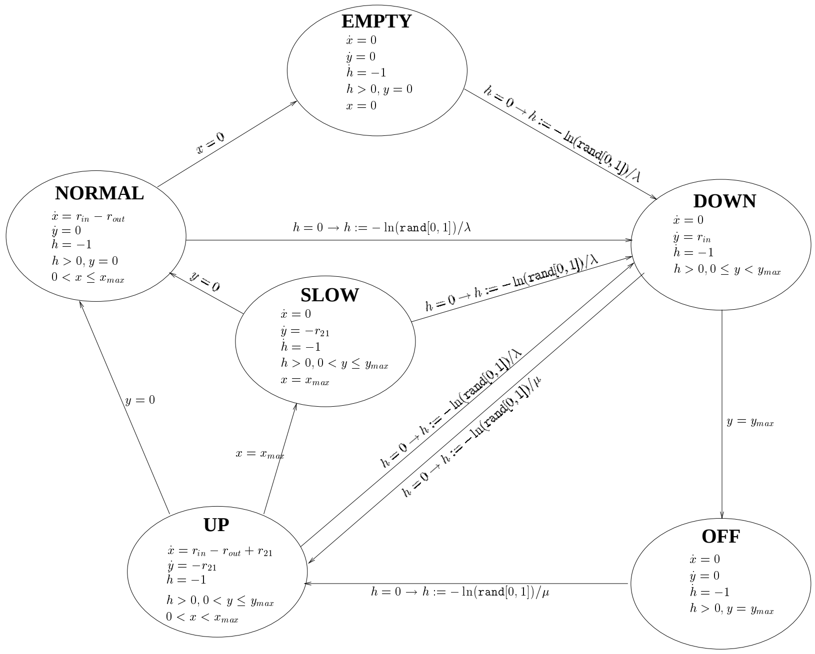

Figure 2. HS model of a dual tank

"Two" of capacity $b_2 = 2.0$*, to prevent the interruption of the flow from this external source. This interruption is then done when tank "Two" is full. When the processing station is repaired, the external flow is immediately switched to tank "One" which resumes its work. At the same time the liquid in tank "Two" is pumped into tank "One" with rate* $r_{21} = 0.1$ *(*$r_{21} + r_{in} > r_{out}$*). If tank One is full, the flow from tank "two" to tank "One" is slowed to* $r_{out} - r_{in}$*. We can compute*

*for instance the steady-state probability that the flow from the external source is interrupted. Using SPNP simulator (*Hirel et al., 2000*;* Ciardo et al., 1989*;* Ciardo et al., 1999*), using* 10^4 *independent replications, the* 95% *confidence interval is obtained:*

$$[4.3836e-02,\ 4.4172e-02]$$

*Now, we use a hybrid system to model this system. After state decomposition and combination, a model with three continuous variables is constructed in Figure 2. In this model, "Empty" is the state where both tanks are empty, "Normal" is the state where the processing station is up and tank two is empty, but not "One", "Up" the state where the processing station is up and tank two is not empty, "Slow" the state where the flow from tank "Two" to tank "One" is slowed, "Down" the state where the processing station is down and "Off" the state where the external flow is stopped. The time of repairs and failures of the processing station are represented by the real variable h, which is reset randomly (following an exponential distribution because when U is a uniform random variable between 0 and 1, $-(\ln U)/\lambda$ is exponentially distributed with rate λ (*Trivedi, 1982*)) and is decreased at rate 1.*

In conclusion, hybrid systems can represent FSPNs, but to do that, we have needed to consider the clocks as the remaining firing times to integrate the randomness of these times in reset relation R, which, unfortunately, is not the easiest case to deal with in hybrid systems. Moreover the randomness on the firing of transitions, even though included in the general formalism, is often not considered in hybrid systems.

5. Can FSPNs represent hybrid systems?

First, note that some of the practical hybrid systems are expressed in a less general way than the one in Section 3, and can sometimes easily be represented by an FSPN. Here, we try to deal with the generality of Section 3.

As the set $\mathcal{Q}$ is supposed finite, let the number of discrete places D be equal to $|\mathcal{Q}|$ and let each discrete place denote a discrete state $q \in \mathcal{Q}$. Of course, there will be a single (discrete) token in the set of all discrete places, indicating the hybrid system is in (discrete) state q when the token is in the place named q. Define $C = n$ fluid places (therefore $x(t) = z(t)$ and even the clocks are then considered as places). The fluid variables are assumed to be non negative. Note that this assumption can easily be weakened by adding a constant to each level so that they are all non negative. The bounds (b in FSPN formalism) of each fluid place will be implicit in the model.

There is a transition from place q to place $q' \in \mathcal{Q}$ if and only if $G(q,q') \neq \emptyset$. A transition from a state q to a state q' occurs in hybrid systems when $z(t) \in G(q,q')$, and the reset relation follows. It means that place q and all fluid places that are decreased in the reset relation R are input places for a transition. The output places are then q' and all the fluid places which are increased in the reset relation $R(q,q')$. By then, we have defined function a in FSPN notation. This transition is enabled following the guard G. It may depend on other fluid places (still following the guard).

Note that a transition involving a discrete place in entrance (resp. exit) involves always only one discrete place in exit (resp. entrance) and that the cardinality of this arc is always 1 (to ensure that there is always exactly one tokens in the set of discrete places).

The transitions are immediate, they fire as soon as the guard is verified. A transition τ fires if its guard (in FSPN formalism) G' is such that $\tau \in G'(x,q)$ which is translated here to $z(t) \in G(q,\tau)$ in the hybrid system formalism. Then the single (discrete) token moves to another place and the contents of fluid output places are re-initialized. Nevertheless, in FSPNs, this "reinitialization" R of the fluid places follows a deterministic rule, so that we can not then suppose it random here.

A last thing to incorporate from hybrid systems formalism to FSP-Ns one is the differential inclusion. This relation defines the marking dependent fluid rate (f in FSPN notations) of (new) arcs connecting continous places and transitions. In FSPNs, these rates are deterministic, so we can not let them be random as in hybrid systems. Moreover, it is difficult to represent these flows as they can be only either positive or negative (depending on the marking), therefore we assume they have a constant sign to be able to define the direction of the arc. So we put a fluid arc from a fluid place to a (new) transition if the flow has a rate of negative sign, with rate minus the i^{th} coordinate of the element of $F(z,q)$, and a fluid arc from a (new) transition to a fluid place if the flow has a rate of positive sign, with rate the i^{th} coordinate of the element of $F(z,q)$ (this not lead to a "well drawn" FSPN, but it seems to work).

The only randomness we introduce then is when two transitions are enabled at the same time, then we follow the same policy as when two guards are satisfied at the same time for the hybrid system.

As we can see, representing hybrid systems by FSPNs is difficult in general. Nevertheless, for some hybrid systems, this is possible. First, this is the case when our restrictions on the randomness of the hybrid system in this section apply, which is the case in control theory, where the continuous dynamics is often represented by ordinary differential equations instead of differential inclusions. Moreover, some particular hybrid systems are represented by more specific models for which an FSPN representation may be easier.

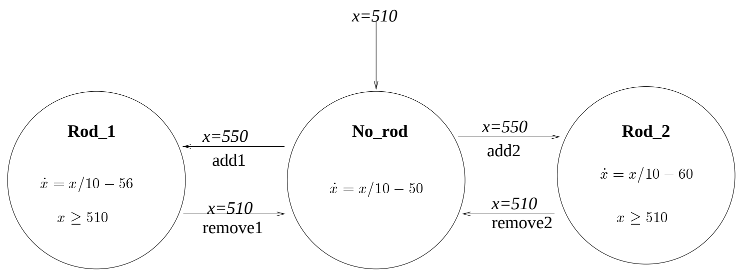

Figure 3. Reactor temperature control example

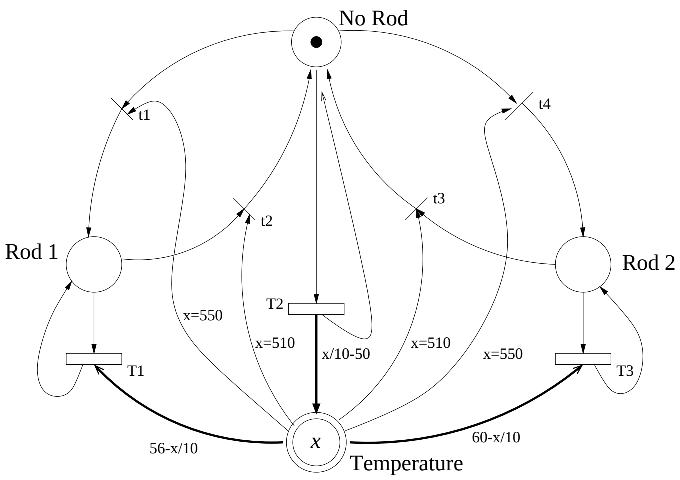

Figure 4. FSPN model of reactor temperature control example

EXAMPLE 2. *We first take the reactor temperature control system from (*Henzinger and Ho, 1995*) as example. The system consists of a reactor core and two control rods that control the temperature of the reactor core. The reactor core is modeled by the hybrid system in Figure 3. The temperature of the reactor core is represented by the variable x. Initially, the core temperature is 510 degree and both control rods are not in the reactor core. In this case, the core temperature rises at a rate equal to $x/10 - 50$. If the core temperature reaches 550 degrees, one of two control rods can be put into the reactor core to dampen the reaction.*

If control rod 1 is put in, the core temperature falls at a rate equal to $56 - x/10$. *Control rod 2 has a stronger effect; if it is put in, the core temperature falls at a rate that equals to* $60 - x/10$. *Either control rod is removed once the core temperature falls back to 510 degrees.*

In Figure 4, an FSPN model is constructed for this system. Three discrete places are used to represent the three states of the system. A fluid place is used to represent the temperature x. *The control actions are modeled by the guards depending on the fluid place. In this system, rod 1 and rod 2 are selected randomly (with the same probability) when a control action is needed. Therefore, the behavior of this system should be better represented by the probability distribution of all states. Simulating by SPNP, we can for example compute the proportion of time each rod is being used in the interval* $[0, 10^3]$. *The* 95% *confidence intervals (using* 10^4 *independent replications) are*

$$[5.9699e-01,\ 5.9772e-01]$$

for no rod,

$$[2.9377e-01,\ 2.9533e-01]$$

for rod one and

$$[1.0768e-01,\ 1.0852e-01]$$

*for rod two. This model can also be made more complex (as in (*Henzinger and Ho, 1995*)) by imposing that when a rod is removed, it can not be put back into the core for a given time. This can be done by creating two new fluid places giving the time elapsed since the rods have been removed.*

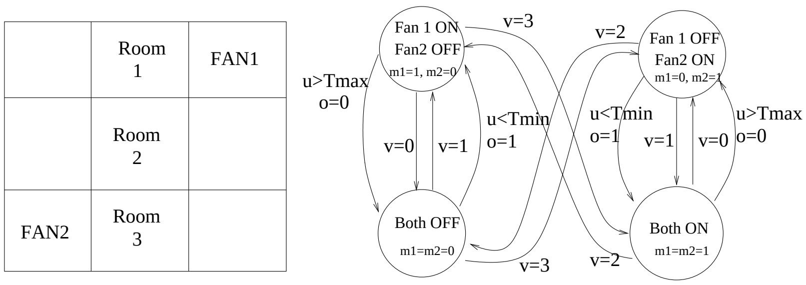

Figure 5. Model of a heating system for three rooms

EXAMPLE 3. *This example, formulated with* control network, *another variant of hybrid system, is taken from (*Bushnell et al., 1998*). This model is about temperature control in three rooms. There are two fans*

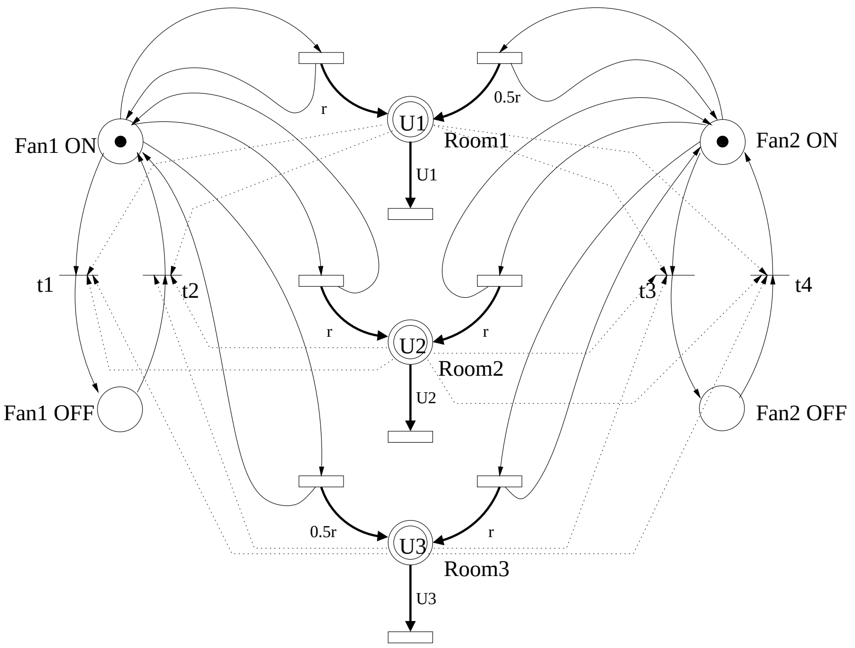

Figure 6. FSPN model for the heating system

on the corners to supply heat for the room. Each fan has different effects on different rooms. Their effects can be modeled by the following differential equations:

$$\dot{u}_1 = -u_1 + rm_1 + 0.5rm_2 \tag{2}$$

$$\dot{u}_2 = -u_2 + rm_1 + rm_2 \tag{3}$$

$$\dot{u}_3 = -u_3 + 0.5rm_1 + rm_2 \tag{4}$$

where u_i *represents the temperature of each room and* m_i *represents the on/off state of each fan (*$m_i = 0$ *if the fan* i *is off and* $m_i = 1$ *if the fan* i *is on).* r *is the heat rate of the fan.*

Figure 5 shows the arrangements of rooms and fans. Also, a finite automaton model is shown for room 1 (u_1 *is then called* u*), where* v *represent a discrete input message read from the network, while* o *represent the discrete output. The continuous state* u *is governed by (2). To completely represent the whole system, we need two more similar automatons. As we transfer this model into FSPN representation (Figure 6), we capture all the characteristics in this model but with a more compact graph. This is due to the representative ability of Petri net. In the Petri net model, we draw some dashed lines to represent possible control strategy for this sytem. Changing of these guards can* *change the system behavior and achieve certain control objective. In addition, the switching between ON and OFF of the two fans can be governed by certain stochastic process, which can easily be represented using FSPN. For instance, suppose that each switching between between ON and OFF of the twofans follows a random distribution between 0 and* 0.2, *and that* $r = 4.0$. *Assume that fan one can be switched on (resp. off) if (as in (*Bushnell et al., 1998*)) the temperature in room one or two hits or is under 0.4 (resp. if the temperature in room one or two hits or is above 1.0). The same control strategy, but considering rooms two and three, is used for fan two. Using SPNP simulator and* 500 *independent replications, the proportion of time each fan is on is given by the following* 95% *confidence intervals:*

$$\textit{Fan 1}\ [1.1309e-01,\ 1.1337e-01],$$

$$\textit{Fan 2}\ [1.1309e-01,\ 1.1336e-01].$$

6. Conclusion

FSPNs and Hybrid Systems are two powerful modeling formalisms using both discrete and continuous variables. In this paper, we have shown that the FSPNs can be expressed as Hybrid Systems, and that the converse property is not always valid (even if it can be the case for a major class of systems). But representing Hybrid Systems by FSPNs has some advantages. Table 6 summarizes the different characteristics of the formalisms. The next issue is the use of the solution methods of one formalism to solve some problems usually formulated in the other formalism. Then we can use the Hybrid Systems tools (for instance HYTECH (Henzinger and Ho, 1995)) to apply the Hybrid systems question such as reachability, liveness or safety to FSPNs and Petri nets tools such as, for instance, SPNP (Ciardo et al., 1989) to solve reliability, performance or performability problems.

References

'On the Simulation of Stochastic Petri Nets'. Computer Science Department. College of Willian and Mary.

Ajmone Marsan, M., G. Balbo, G. Conte, S. Donatelli, and G. Franceschinis: 1995, *Modelling with Generalized Stochastic Petri Nets*. John Wiley & Sons.

Alla, H. and R. David: 1998, 'A modelling and analysis tool for discrete event systems: continuous Petri net'. *Performance Evaluation* **33**, 175–199.

Table I. Hybrid Systems vs FSPNs.

	Hybrid Systems	FSPNs
discrete part	discrete states $q \in Q$	discrete places $\mathcal{P}_D$
continuous part	continuous variables $x \in I\!R^n$	continuous places $\mathcal{P}_C$
continuous evolution law	differential inclusions $\dot{x} \in F(q,x)$	marking-dependent fluid rate f
discrete transition	"guard" $G(q,q') \subset I\!R^n$ reset relation $R(q,q') \subset I\!R^n \times I\!R^n$	guard function : $\mathcal{T} \times \mathcal{S} \to \{0,1\}$ firing of transitions
Initial conditions	$(x(t_0) = x_0, q_0)$,	(x^0, m^0)
Description	"hidden" in F, R, G	$a, f, g, >, d, r, F, \omega, b$
Non-deterministic behaviour	differential inclusion multiple guard selection multiple reset selection	stochastic firing time of transitions, probabilistic choice of transitions firing at the same time through weight function ω.
Application fields	supervisory control, process control, robotics modeling and control	reliability modeling, performance and performability modeling
Focus	verification, validation of: reachability, safety, liveness	Stochastic properties
Solution methods	Machine reasoning, ODE simulation	numerical solution of PDES Discrete event simulation
Advantages	Great Generality	Complexity, concurrency, randomness issue

Alur, R., C. Courcoubetis, T. Henzinger, and P. Ho: 1991, 'Hybrid Automata: An Algorithmic Approach to the Specification and Verification of Hybrid Systems'. In: A. P. R. Robert L. Grossman, Anil Nerode and H. Rischel (eds.): *Hybrid Systems*, Vol. 736 of *Lecture Notes in Computer Science*. pp. 209–229.

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

Bushnell, L., O. Beldiman, and G. Walsh: 1998, 'An Equivalence Between a Control Network and a Switched Hybrid System'. In: T. A. Henzinger and S. Sastry (eds.): *Hybrid Systems: Computation and Control*, Vol. 1386 of *Lecture Notes in Computer Science*. pp. 64–79.

Cassandras, C.: 1993, *Discrete Event Systems: Modeling and Performance Analysis.* Aksen Associates Incorporated Publishers and Irwin.

Ciardo, G., A. Blakemore, P. Chimento, J. Muppala, and K. Trivedi: 1993, 'Automated Generation and Analysis of Markov Reward Models using Stochatic Reward Nets'. In: C. Meyer and R. Plemmons (eds.): *Linear Algebra, Markov Chains and Queuing Models*, Vol. 48 of *IMA Volumes in Mathematics and its Applications*. Springer-Verlag, Heidelberg, pp. 145–191.

Ciardo, G., J. Muppala, and K. Trivedi: 1989, 'SPNP: Stochastic Petri net Package'. In: *Proc. Third International Workshop on Petri Nets and Performance Models, PNPM'89*. pp. 142–151.

Ciardo, G., D. Nicol, and K. Trivedi: 1999, 'Discrete-Event Simulation of Fluid Stochastic Petri-Nets'. *IEEE Transactions on Software Engineering* **25**(2), 207–217.

David, R.: 1997, 'Modeling Hybrid Systems Using Continuous and Hybrid Petri Nets'. In: *Proc. Seventh International Workshop on Petri Nets and Performance Models, PNPM'97*. pp. 142–151.

Gaver, D.: 1962, 'A waiting line with interrupted service, including priorities'. *Journal of Royal Statistical Society* **B24**, 73–90.

Henzinger, T. and P. Ho: 1995, 'HYTECH: The Cornell HYbrid TECHnology Tool'. In: P. Antsaklis, W. Kohn, A. Nerode, and S. Sastry (eds.): *Hybrid Systems II*, Vol. 999 of *Lecture Notes in Computer Science*. pp. 265–293.

Henzinger, T., P. Kopke, and P. Varaiya: 1995, 'What's decidable about hybrid automata?'. In: *Proc. 27th Symposium on the Theory of Computing*. pp. 373–382.

Hirel, C., B. Tuffin, and K. Trivedi: 2000, 'SPNP Version 6.0'. In: B. Haverkort, H. Bohnenkamp, and C. Smith (eds.): *Computer performance evaluation: Modelling tools and techniques; 11th International Conference; TOOLS 2000, Schaumburg, Il., USA*, Vol. 1786 of *Lecture Notes in Computer Science*. Springer Verlag, pp. 354–357.

Horton, G., V. Kulkarni, D. Nicol, and K. Trivedi: 1998, 'Fluid Stochastic Petri nets: Theory, Application and Solution'. *European Journal of Operational Research* **105**, 184–201.

Marie, R. and K. Trivedi: 1987, 'A Note On the Effect of Preemptive Policies on the Stability of a Priority Queue'. *Information Processing Letters* **24**(6), 397–401.

Peterson, J.: 1981, *Petri nets and the Modeling of Systems*. Prentice-Hall, EnglewoodCliffs, NJ.

Puri, A. and P. Varaiya: 1994, 'Decidability of hybrid systems with rectangular inclusions'. In: D. Dill (ed.): *Computer Aided Verification, CAV'94*, Vol. 818 of *Lecture Notes in Computer Science*. pp. 95–104.

Reisig, W.: 1985, *Petri Nets: An Introduction*. Springer-Verlag.

Trivedi, K. S.: 1982, *Probability and Statistics with Reliability, Queuing, and Computer Science Applications*. Englewood Cliffs: Prentice-Hall, Inc.

Varaiya, P.: 1999, 'Design, Simulation and Implementation of Hybrid Systems'. In: S. Donatelli and J. Kleijn (eds.): *Application and Theory of Petri Nets 1999*, Vol. 1639 of *Lecture Notes in Computer Science*. pp. 1–5.