Hybrid Systems Modelling: Mixed Petri nets

C. Valentin-Roubinet

Abstract: - This paper presents a method to model and simulate Hybrid dynamics Systems in order to validate their supervision model by simulating it in closed loop with the process model. The qualitative aspects are modelled by Petri nets and the quantitative aspects dealing with activities durations by differential-algebraic equations. These two modelling tools are integrated into Mixed Petri nets, which encapsulate the structuration power of Petri nets and the continuous description power of differential-algebraic equations. It is then applied to two simple exemples to illustrate the concepts defined in the paper.

Key-Words: - hybrid dynamic systems, modeling, high level Petri nets, simulation, algebro-differntial equations, supervision, state space *IMACS/IEEE CSCC'99 Proceedings,* Pages:2601-2606

1 Introduction

Discontinuous processes, where continuous, discrete and sequential aspects are deeply linked belong to the class of Hybrid Dynamical Systems (HDS) [1]. These kind of processes are widely developed in a lot of industries as: chemical, pharmaceutical, gas, petrol or food industries. There are very complex because they include both continuous and discrete features. Often, schedules of conditions to switch from one configuration to another are not obvious and require a global knowledge of the process states.

The plant must satisfy several specific requirements imposed by its designers and general technical, economic, and social conditions in the presence of ever-changing external disturbances:

* Safety: pressures, temperatures, concentration of chemicals should always be within allowable limits,
* Quality and Quantity: the customer requirements should be respected,
* Operational constraints: storages should not overflow or go dry, temperatures should not exceed an upper limit since the reaction may go to uncontrolled, ...
* Environmental constraints: limit temperature and concentration of chemicals in the effluents should be respected,
* Economics: energy, capital, raw materials, storage and human labor should be minimized,
* •

Therefore, to validate the supervision design of such complex systems with numerous constraints, it is necessary to simulate the behaviour of the supervised system. In that context, we need an adapted methodology to model such complex systems able to describe continuous features, discrete features, events, and links between them.

Various methods have been proposed, with different level of interaction between continuous features, events and discrete states. Their common point of view is that the continuous evolutions are conditioned by events which change the dynamics of the process and that the discrete state of the process depends on the continuous variables behaviour (threshold crossing by a continuous variable or its derivative, ...)

Usually, an interface synchronizes both aspects. Its role is to convert critical evolutions of continuous variables into sequences of events or into conditions and vice versa, events or discrete states into continuous states or specific dynamics. The state space is then partitioned into a finite number of regions. Events are generated each time a boundary is crossed. In the literature, these concepts are not always defined using these notions, but they always occur. They are given various names: generator/actuator [9], phase invariant or system invariant [8], zero marking of continuous places of a hybrid Petri net [2], event generator [6], quantiser/injection [4], ... It must be pointed out that the continuous part does not necessarily models the controled process and the discrete part the supervision. Both aspects may be mixed into either parts. Abrupt dynamic changes may happen within the process itself, without external influence (they will be named autonomous) or controled by the supervision part (they will be named controled). As well, abrupt autonomous or controled state jumps may happen within the process.

We propose a generic method to model and simulate supervised hybrid dynamic systems in order to validate the supervision model in closed loop with the process model.

To reach this goal, we developed a modelling tool [5], [7], well adapted to the characteristics of the hybrid dynamic systems: it must be able to represent sequences of activities, duration of activities, choices, synchronizations and shared resources descriptions. Petri nets fuffil these requirements very well. We need to represent continuous evolutions as well. Differential Algebraic Equations (DAE) fullfil these requirements very well. Therefore, a tool encapsulating these two basic tools satisfy our needs. It is Petri nets interpreted with DAE. We name this tool Mixed Petri nets (MPN). It belongs to the class of High level Petri nets [3]. It is defined in section 2 into details. Section 3 illustrates the power of this model.

2 Modelling

We assume that the reader is familiar with basic concepts and definitions of ordinary Petri nets [2].

Definition 1: a Mixed Petri net (MPN) is a 5-uplet MPN = {N, CP, HI, M0, Var0}.

* N is an ordinary Petri Net with n places and m transitions,
* CP is the model Continuous Part, PC = {Var, ADE},
* HI is the Hybrid Interface, internal interpretation of the model. HI = {AEP, AEM, CJ, TC, TE}
* M0 is the initial marking of the net (initial state of the discrete event part).
* IS is the Initial State of discrete, continuous or hybrid variables of Var.

Definition 2: CP is the continuous part of the model, CP = {Var, ADE}:

* Var is a set of continuous or hybrid state variables, Var_i is the vector of active variables associated with place P_i,
* ADE is the set of Algebro-Differential Equations modelling the different system behaviors.

Definition 3: HI is the model Hybrid Interface. It models the continuous to discrete or discrete to continuous interactions. HI = {AEP, AEM, CJ, TC, TE}

* AEP is a n elements set. Each element of AEP is a set, AEP_i of Active Equations associated to the Place P_i. If place P_i is not safe, $AEP_i = \emptyset$. AEP_i defines a simple link between a unique place P_i and a set of equations defining the evolution of the sub-system which configuration is represented by the place P_i.
* AEM is a set of active equations associated to a marking M_i with at least two marked places ($\|M_i\|>1$). AEM defines multiple links between a situation described by the marking of several places and a set of equations defining the evolution of the sub-systems represented by these places.
* CJ is a n elements set. Each element, Cj_i, is a Var_i size vector. Component i is the controlled jump applied on variable i when a mark is added to place P_i.
* TC is a m elements set. Each element of TC, TC_j is the Condition associated to the transition T_j. TCj: Var $\rightarrow$ {0, 1}. It is a boolean function.
* TE is a m element set. Each element of TE, TE_j is an events pair associated with transition T_j. This pair is composed of an input event TEi_j and of an output event, TEo_j. ε is the always occuring event. η is the never occuring event. These events are internal events. They are needed to synchronized several parts of the model.

These model definitions clearly define the different parts of the model: the discrete part is represented by {N, M0}, the continuous part by {ADE, Var, Var0} and the hybrid interface by {AEP, AEM, CJ, TC, TE}. Evolution rules given after, define clearly how each element influences the system behavior. Continuous variables are modified according to discrete state (configuration), which defines the process dynamics (AEP, AEM), and the discrete state depends on the continuous variables position into the state space in comparison with thresholds.

Theorem 1: Each MPN marking is associated with a set of equations. Its homogeneity is guaranted by the cardinal equality between $\bigcup_{i \in \{1..n\}} Var_i$ and $\left(\bigcup_{i \in \{1..n\}} ADEP_i\right) \bigcup ADEM$.

Like in ordinary Petri nets, marking evolution is due to transition firing. Added to place marking, continuous elements are involved into firing conditions. A place P_i is an input place of T_j iff $Pre(P_i,T_j) > 0$. The set of all input places (Pre-set) of T_j is denoted by $°T_j$. A place P_i is an output place of T_j iff $Post(P_i,T_j) > 0$. The set of all output places (Post-set) of T_j is denoted by $T_j°$.

Definition 4: A transition T_j is net-enabled (net-firable) at a marking M iff $\forall P_i \in °T_j$, $M(P_i) \geq Pre(P_i, T_j)$.
This is the same definition as a transition enabled in an ordinary Petri net.

Definition 5: A transition T_j is boundary-net-enabled (boundary-net-firable) at a marking M iff it is net-enabled and if:

* $TC_j = 1$,
* TEi_j occurs.

Definition 6: When a transition t_j is fired, MPN marking being M', and variable state being Var':

* $\forall p_i \in P$, $M(p_i) = M'(p_i) + Pre(p_i,t_j) - Post(p_i,t_j)$, thus $Pre(P_i, T_j)$ marks are removed from t_j input places and $Post(p_i,t_j)$ are added to t output places.
* Integration of algebro-differential equations AEP_i associated with t_j input places is stopped.
* Integration of algebro-differential equations AEP_k associated with t_j output places is started.
* Integration of algebro-differential equations AEM associated with M making is started.
* TEc_j occurs.
* active variables associated with pi make a controlled jump defined by Cji: $Var_i \rightarrow Var_j$

Definition 7: algebro-differential equations are continuously integrated during the time between two transition firings. The variables state is known all over the time.

Definition 8: $\forall p_i \in P$ / $(M(p_i) = 1) \wedge (EADPi \neq \emptyset)$, $\exists \tau_a \in R$ and $\exists t_j \in p_i\bullet$ / $TC_j(\tau_a) = 1$. Algebro-differential equation integration is stopped at $t = \tau_a$ and transition t_j validation is evaluated.

When a safe place Pi is marked, the set AEP of equations associated with it is active. It means that the variables of Var concerned with these equations follow the dynamics described by the active equations. The initial state of the concerned continuous or hybrid variables is equal to the last state reached before the last firing. There may be state discontinuity (CJ) or derivative discontinuity (transition firing).

The simulation model is composed of a continuous part described by DAE, representing all the possible dynamics of the process, the *process model*, a discrete event part representing all the possible configurations of the process and how to switch from one to another, the *supervision model*, and interfaces between the two. To avoid state explosion, the supervision model is decomposed into several Mixed Petri nets, each describing one physical phenomenon within the process. The global configuration of the process at any time is then represented by the marking of the supervision model. Therefore, if pp is the number of physical phenomena, $\| M_i \| \geq pp$.

At each configuration of the process corresponds a specific dynamic and a specific region of the state space. Therefore, the state space can be partitioned into exactly as many regions as configurations. The *process interface* detects when one of the state variable crosses a boundary of the active region. Then, the supervision model calculates the configuration changes. The *supervision interface* associates to each discrete state of the supervision model (mixed Petri nets marking) the corresponding dynamics of the process model.

3 Exemples

To illustrate this modelling methode, the exemple presented figure 1 is explained into details. It is made of two tanks, which levels x1 and x2 are continuously observable. The two control actions available are opening or closing ON/OFF valve V0. The goal is to control level in tank 2 such as $x2refmin \leq x2 \leq x2refmax$, behavioral constraints on the tanks being respected: $x1min \leq x1 \leq x1max$ and $x2min \leq x2 \leq x2max$.

Flow rate Q0 into input valve V0 is such as $Q0 = q0*u$.

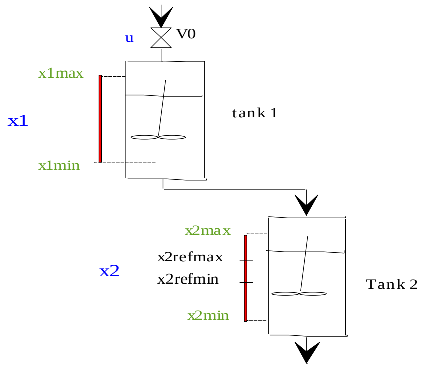


Fig. 1: two cylinder tanks process

Algebro-differential equations of this system are the following:

$$\begin{cases} \dot{x}1 = \dfrac{-1}{R1S1}x1 + \dfrac{q0}{S1}u & (1) \\ \dot{x}2 = \dfrac{1}{R1S2}x1 + \dfrac{-1}{R2S2}x2 & (2) \\ \text{u} = 0 & (3) \\ \text{u} = 1 & (4) \end{cases}$$

With:

$$Var = \begin{bmatrix} x1 \\ x2 \\ u \end{bmatrix}, \ Var0 = \begin{bmatrix} 0{,}2 \\ 0{,}35 \\ 0 \end{bmatrix}, \ VD_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \ VD_2 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix},$$

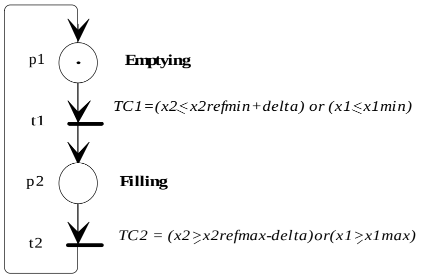
$$TE = \begin{bmatrix} (\varepsilon,\eta) \\ (\varepsilon,\eta) \end{bmatrix},$$

$$TC = \begin{bmatrix} (x2 \le x2ref\ \min + delta) \vee (x1 \le x1\min) \\ (x2 \ge x2ref\ \max - delta) \vee (x1 \ge x1\max) \end{bmatrix},$$

$M_0 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$, delta choosen by the designer.

The discrete part is made of only one net, therefore, there is no synchronization by means of events. As well, there is no state jump in this simple exemple (there could be one if an object were introduced into the tank to have a chemical treatment).

The supervision model could be the following. Only valve V0 is controled, thus, x1 is a first order model and x2 a second order.



delta=0.01

Fig. 2: supervision model

3.1 Discrete part to continuous part actions

Places	$EADP_i$	Variables actives, Var_i
p1	(1), (2), (3)	x1, x2, u
p2	(1), (2), (4)	x1, x2, u

Table1: simple links between places and equations

MPN marking gives the continuous evolutions of the system. If place p1 contains 1 mark, equations (1) and (2) are continuously integrated and equation (3) is added to the model. If place p2 contains 1 mark, equations (1) and (2) are continuously integrated and equation (4) is added to the model. In that particular case, $Var_1 = Var_2 = Var$ because the system is simple, but it is not always the case. As well, the mixed Petri net is made of only one kind of place. They model an action from discrete part to continuous part when $EADP_i \neq \emptyset$ et $Var_i \neq \emptyset$.

3.2 Continuous part to discrete part actions

In that particular example, no equations are associated with a marking, because the model is made of only one MPN. But a set of equations AEP_i is associated with each place.

These active equations describe the continuous evolutions associated with each configuration given by a special place of the mixed petri net. TC_i calculation and firing times are realised from these equations two.

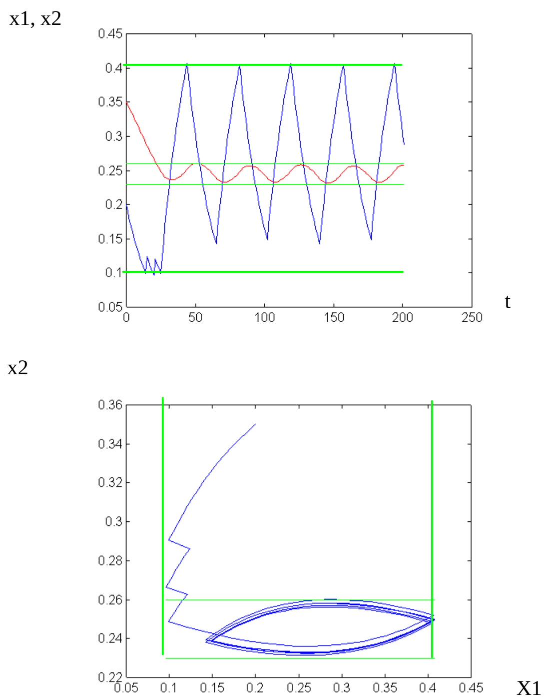


Fig. 3: a/ timed evolution (x1 blue, x2 red), b/state space x2=f'(x1), Green, switching lines.

As soon as the levels stay inside the switching lines, the proposed supervision is accepted and is stable.

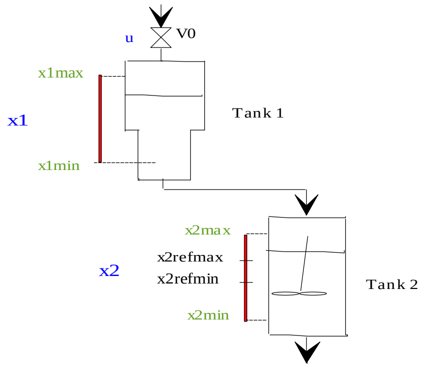


Fig. 4. process with one cylinder tank and one two-levels-cylinder tank

Algebro-differential equations of this system are the following:

$$\begin{cases} \dot{x}1 = \dfrac{-1}{R1S1}x1 + \dfrac{q0}{S1}u & (5) \\ \dot{x}1 = \dfrac{-1}{R1S'1}x1 + \dfrac{q0}{S'1}u & (6) \\ \dot{x}2 = \dfrac{1}{R1S2}x1 + \dfrac{-1}{R2S2}x2 & (7) \\ u = 0 & (8) \\ u = 1 & (9) \end{cases}$$

The supervision model could be the following. The discrete part of the model is divided in two, because two differnt physical phenomena occur inside the system: V0 supervision and tank 1 diameter change;

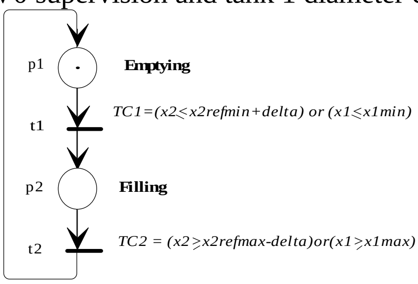


Fig. 5: controlled dynamic change: V0 supervision

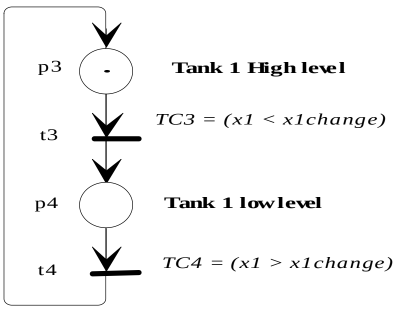


Fig 6: autonomous dynamic change: tank shape

3.3 Discrete part to continuous part actions

Places	$EADP_i$	Variables actives, Var_i
p1	(7), (8)	x2, u
p2	(7), (9)	x2, u
P3	(5),	x1
P4	(6)	x1

Table 2: simple links between places and equations

The example 2 illustrated a less obvious case. If place p1 contains 1 mark, equations (7) and (8) are continuously integrated and calculates the evolution of level x2 when the system is emptying (u=0). If place p2 contains 1 mark, equations (7) and (9) are continuously integrated and calculates the evolution of level x2 when the system is filling (u=1). In that case, $Var_1 \neq Var_2 \neq Var$. If p3 contains 1 mark, equation (5) is continuously integrated and calculates the evolution of level x1 when this level is higher than x1change. If p4 contains 1 mark, equation (6) is continuously integrated and calculates the evolution of level x1 when this level is less than x1change.

3.4 Continuous part to discrete part actions

In this second example either, no equations are associated with a marking.

These active equations describe the continuous evolutions associated with each configuration given by a special place of the mixed petri net. TC_i calculation and firing times are realised from these equations two.

4 Advandages and drawbacks

When the process behaviour can be represented by a knowledge model, based on Differential Algebraic Equations (DAE), this modelling and simulation method is very well suited. The configuration switches are represented with Petri nets and the duration of activities is calculated very precisely from the DA equations. In this example, there is neither sequences nor synchronization between subprocesses but only choices. Mixed Petri nets (MPN) would be also well adapted to these more complex discrete event aspects (for example, multiproducts multibatchs systems).

The modelling is quite closed to reality and need few simplifying hypothesis. The qualitative aspects are modelled by PN and the quantitative aspects dealing with activities durations by DAE. These two modelling tools are integrated into Mixed Petri nets, which encapsulate the structuration power of Petri nets and the continuous description power of differential-algebraic equations.

If the process behaviour can not be described with a knowledge model because of its complexity, this method can be used only if the activities can be represented only as a duration. In other words, in that case, the process can be modelled by this way only if it can be looked at as a discrete event process.

5 Conclusion

Mixed Petri nets have been defined in this paper. This model is aimed to model hybrid dynamic systems in order to validate their supervision by simulation. The evolution rules are presented and illustrated on two exemples pointing out different aspects of the model. Simulation results are given for the first example.

References:

[1] M. S. BRANICKY, V.S. BORKAR, S.K. MITTER, A unified framework for Hybrid Systems, *In 11^e Conférence Internationale sur l'Analyse et l'Optimisation des systèmes: Systèmes à événements discrets*, *DES'94*, Sophia Antipolis, France ; june 1994.

[2] R. DAVID, H. ALLA, "Petri nets for modeling of dynamic systems - A survey", Automatica, vol. 30, n° 2, pages 175-202.

[3] K. JENSEN, G. ROZENBERG, *High-Level Petri Nets*, Springer-Verlag, Berlin.

[4] J. LUNZE, B. NIXDORF, H. RICHTER, Hybrid Modelling of continuous-variable systems with application to supervisory control, ECC'97, Bruxelles, Belgique.

[5] C. VALENTIN-ROUBINET, Modeling of Hybrid Systems : DAE supervised by Petri Nets. The example of a gas storage, *Proccedings of the Symposium ADPM'98*, Reims, March 1998.

[6] R. VALETTE, H. PINGAUD, A. PAGES, D. ANDREU, J.C. PASCAL, Modeling, simulation and control of event-driven operations in process systems, *INRIA-IEEE Symposium on Emerging Technologies and Factory Automation*, vol.3, p. 119-128, Paris, France.

[7] D. VIBERT, C VALENTIN-ROUBINET, E. NIEL, A modelling Method to take into account fluctuations of continuous variables in a class of Hybrid systems, *European Control Conference*, Bruxelles, july 1997.

[8] R. ALUR, C. COURCOUBETIS, T. HENZINGER, P. HO, X. NICOLLIN, A. OLIVERO, J. SIFAKIS, S. YOVINE, The Algorithmic Analysis of Hybrid Systems, *In 11^e Conférence Internationale sur l'Analyse et l'Optimisation des systèmes: Systèmes à événements discrets*, *DES'94*, Sophia Antipolis, France.

[9] J.P.ANTSAKLIS, M.D.LEMMON, J.A.STIVER, Hybrid System modeling and Event Identification, Technical Report of the ISIS group ISIS-93-002, University Notre Dame, Notre Dame, IN.