A Petri Nets approach for hybrid systems modeling

Mircea Adrian Drighiciu, Anca Petrisor and Marius Popescu

***Abstract*—** This paper focuses on the modeling of hybrid systems with autonomous commutation of the model generated by a hysteresis phenomenon through a particular Petri Nets structures, called Modified Petri Nets (MPN). The main goal of this approach is to get a formal description language for such hybrid systems, which combines the advantages of a graphical description with the possibility of a transparent visualization, simulation and analysis. Hence, several enhancements were proposed. The first of them combines the classical discrete Petri Net approach and the concept of continuous Petri Nets, having as result a class called Hybrid Petri Nets (HPN). In the second enhancement, the aspect of the system complexity was approached by introducing object oriented concepts, like encapsulation and information hiding. In this way, the resulting Hybrid Object Nets (HON) combines the advantages of Hybrid Petri Nets with those of the object-oriented paradigm. The proposed concepts are illustrated with a case study, which refers to a classical temperature control process in a room, using a thermostat with anticipative resistance.

***Keywords*—**Dynamical systems, hybrid systems, Petri Nets, switched systems.

I. INTRODUCTION

A dynamical system is generally considered a hybrid structure if it is difficult to deal with it either as a purely continuous-variable system or as a purely discrete-event system without ignoring important phenomena that result from the combination of continuous and discrete movements of this system. This situation is due to the fact that the theories of continuous and discrete systems have been elaborated completely separately until recently. Hybrid systems pose the problem of bridging the gap between both theories. This has been done until now not only by considering a combination of continuous and discrete subsystems but also by investigating different extensions of either continuous or discrete systems.

This paper was presented at 7th WSEAS International Conference on SYSTEM SCIENCE and SIMULATION in ENGINEERING (ICOSSSE'08), Venice, Italy, November 2008, receiving a recommendation for development and publication in NAUN Journals. This work was supported in part by Romanian Ministry of National Education under CNCSIS research grant, A type, Code 606, No. 27 GR - Researches on Petri Nets for flexible robotized structures control.

Mircea Adrian Drighiciu is associate professor at University of Craiova, Faculty of Electromechanical, Environmental and Informatics Engineering, Electromechanical Department, 107 Decebal Bd., Craiova, Romania (e-mail: adrighiciu@em.ucv.ro).

Anca Petrisor is engineering researcher at University of Craiova, Faculty of Electromechanical, Environmental and Informatics Engineering, Electromechanical Department, 107 Decebal Bd., Craiova, Romania (e-mail: apetrisor@em.ucv.ro).

Marius Popescu is associate professor at University of Craiova, Faculty of Electromechanical, Environmental and Informatics Engineering, Electromechanical Department, 107 Decebal Bd., Craiova, Romania (e-mail: mrpopescu@em.ucv.ro).

Hybrid dynamical systems generate variables or signals, that are mixed signals consisting of combinations of continuous or discrete value or time signals, and through them interaction with other systems and the environment occurs. More specifically, some of these signals take values from a continuous set (e.g. the set of real numbers) and others take values from a discrete, typically finite set (e.g. the set of symbols $\{\alpha, \beta, \gamma\}$). Furthermore, these continuous or discrete-valued signals depend on independent variables such as time, which may also be continuous or discrete. Another distinction that could be made is that some of the signals could be time-driven while others could be event-driven in an asynchronous manner [1], [2], [3], [16].

A hybrid system is a dynamical system that cannot be represented and analyzed with sufficient precision either by the methods of the continuous systems theory or by the methods of the discrete systems theory. It is known that continuous systems theory assumes that the system under consideration can be described by some differential equation:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), t), \quad \boldsymbol{x}(0) = \boldsymbol{x}_0 \tag{1}$$

$$\boldsymbol{y}(t) = \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{u}(t), t) \tag{2}$$

where $\boldsymbol{x} \in \mathrm{R}^n$ is the state vector, $\boldsymbol{u} \in \mathrm{R}^m$ the input vector and $\boldsymbol{y} \in \mathrm{R}^r$ the output vector. $\boldsymbol{x}_0$ denotes the initial state. More generally, (1) can be replaced by a set of difference and algebraic equations, which then is called a differential-algebraic system (DAE system) [16], [17], [20], [22], [23].

The key assumption of continuous systems theory concerns the fact that the functions $\boldsymbol{f}$ and $\boldsymbol{g}$ satisfy a Lipschitz condition. With respect to the state **x** this smoothness assumption means for the function $\boldsymbol{f}$ that a constant L has to exist for which the inequality:

$$\|\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) - \boldsymbol{f}(\hat{\boldsymbol{x}}, \boldsymbol{u}, t)\| \leq L \cdot \|\boldsymbol{x} - \hat{\boldsymbol{x}}\| \tag{3}$$

holds for all $\boldsymbol{x}$, $\hat{\boldsymbol{x}}$, $\boldsymbol{u}$ and t, where $\|\cdot\|$ symbolises a vector norm. A similar condition should be satisfied with respect to $\boldsymbol{u}$. Under this assumption, uniqueness and existence results can be derived for the solution of the differential equation (1). Furthermore, many analysis methods assume the property (3).

In our paper, we are interested on a class of hybrid dynamical systems with commutation. An abrupt change of the vector field $\boldsymbol{f}$ if the state $\boldsymbol{x}$ reaches a given bound is called switching. Formally, the system can be represented by two or more different vector fields $\boldsymbol{f}_q$ together with conditions that describe the validity of these vector fields, for example by:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) \tag{4}$$

with:

$$f = \begin{cases} \boldsymbol{f}_1(\boldsymbol{x}) \text{ for } \boldsymbol{h}(\boldsymbol{x}) \leq 0 \\ \boldsymbol{f}_2(x) \text{ for } \boldsymbol{h}(\boldsymbol{x}) > 0 \end{cases} \tag{5}$$

If the system is currently described by the vector field $\boldsymbol{f}_1$ and the state reaches the border $\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{0}$ of the region of validity of this vector field, the vector field switches to $\boldsymbol{f}_2$ which is valid until the border described by $\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{0}$ is reached from the other side.

The same arguments apply to systems with controlled switching where the vector field also changes abruptly in response to an input command u. Here, the notion of switching is used for systems with piecewise constant input, where for a given time interval the input is fixed to some value $\overline{\boldsymbol{u}}$ and, hence, the vector field is fixed to

$$\boldsymbol{f}(\boldsymbol{x}, \overline{\boldsymbol{u}}) = \overline{\boldsymbol{f}}(\boldsymbol{x}) \tag{6}$$

This kind of switching is nothing else than a change in the vector field due to a given input, the system with piecewise constant input being considered as an autonomous system with switching dynamics, which does not imply that the system exhibits hybrid phenomena [16], [17], [20], [23].

In order to do a unitary conception of the hybrid systems representation, different approaches of modeling are used and at present there is already an abundance of such models. They can be characterized and described along several dimensions. In broad terms, approaches differ with respect to the emphasis on or the complexity of the continuous and discrete dynamics, and on whether they emphasize analysis and synthesis results or analysis only or simulation only.

In fact, the modeling of hybrid systems needs a combination of description methods for discrete systems and for continuous systems. The classical timed Petri Nets approach with its discrete state space is well suited for the field of discrete systems, but not for continuous systems. For the field of continuous systems, the continuous Petri Nets approach [5], [6], [7], [10], [12], [13], [14], [19] is useful because it offers a continuous sate space. The combination of a discrete and a continuous state space is a main condition for the hybrid structures modeling. Frequently in hybrid systems, the event-driven dynamics were studied separately from the time-driven dynamics via automata or Petri nets models or via differential or difference equations.

Hence, if it is possible to describe the behavior of continuous systems with continuous Petri Nets and then to combine these models with the discrete world of common timed Petri Nets, we would be able to model the complex behavior of hybrid dynamical systems using a single graphical or analytical formalism [6], [8], [9], [10], [16], [21].

II. Reviews on Hybrid Petri Nets

In a discrete Petri Net (PN), the marking of a place may correspond either to the Boolean state of a device (for example a motor is turned on or off), or to an integer (for example the number of parts in a conveyor input buffer). A general analysis method is to compute the set of reachable states and deduce the different properties of the system. But when a Petri net contains a large number of tokens, the number of reachable states explodes and this is a practical limitation of the use of Petri nets. This observation led us to define continuous and hybrid Petri nets.

An autonomous HPN is a sextuple Q = {P, T, **Pre**, **Post**, $\mathbf{m}_0$, *h*} such that: P = {P_1, P_2, …P_n} is a finite, not empty, set of places; T = {T_1, T_2, …, T_m} is a finite, not empty, set of transitions; $P \cap T = \varnothing$ (P and T are disjointed); *h*, called "hybrid function" indicates for every node whether it is a discrete node or a continuous node; **Pre** : P x T $\rightarrow R^+$ or N^+, is the input incidence mapping; **Post** : P x T $\rightarrow R^+$ or N^+, is the output incidence mapping and $\mathbf{m}_0$: P $\rightarrow R^+$ or N^+ - the initial marking of the net [5], [6].

The basic model of a non-autonomous HPN consists in a combination between non – autonomous discrete and continuous sub-models. So, generally speaking, the discrete transition in a non-autonomous HPN may be fired as the transition in a discrete PN (i.e. they may be synchronized, or timed with constant or stochastic timings). Similarly, the continuous transitions in a non-autonomous HPN may be fired with a flow rate, as transitions in a continuous PN (they can be synchronized, or maximal speeds may be constant, or function of time, or function of the marking) [12], [15].

Informally, there are two parts in a hybrid Petri net, a discrete part and a continuous part, and these parts are interconnected thanks to arc linking a discrete node (place or transition) to a continuous node (transition or place), (Fig.1). In some cases, one part can influence the behavior of the other part without changing its own marking. In other cases, the firing of a D – transition can modify both the discrete and the continuous marking.

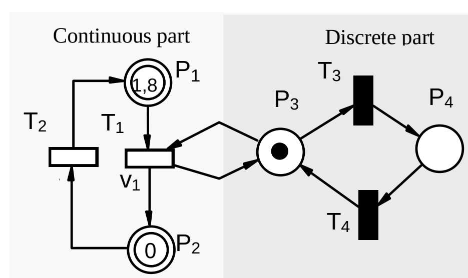

Fig.1 An Hybrid Petri Net model

Continuous places of the model are P_1 and P_2; the continuous transitions are T_1 and T_2, the discrete places – P_3 and P_4, and the discrete transitions – T_3 and T_4. Starting from the initial marking (initial conditions) of the model and firing T_1 with a constant flow (firing speed – v_1) 0.1 another marking (state of the system) in reached - (1.7, 0.1, 1, 0). A marking quantity 0.1 has been removed from P_1 and P_3, which are input transitions, and the same quantity has been added to P_2 and P_3 - output transitions.

The HPN, as were defined are not sufficient to model hybrid systems because they do not cover the possibility of modeling the dynamic behavior of continuous systems. Since continuous Petri Nets allow the modeling of real values, they can be used to model continuous state variables only. Therefore, a powerful extension of basic formalism, called MPN allows introducing several enhancements:

- the firing speed of continuous transitions can be given as a function of token quantities, opening the possibility of modeling the behavior of continuous dynamics, due to fact that the values of this function can become positive as well as negative;
- the token quantity of continuous places can take as well positive as negative values for modeling positive as well as negative continuous system variables, whereas HPN only allow positive values.

The combination of discrete and continuous subnets allows the modeling of hybrid systems according the same description language and rules [4], [5], [7].

Usually, a continuous system is described by its input, output and by its system behavior. Using a MPN model (Fig.2), the input and output variables are each described with a continuous place. The transition T_1 is always active and the system behavior is described with the firing speed function (flow rate) $v = f(u, y)$ depending on P_1 and P_2 marking. Moreover, due to $P_1 - T_1$ test arc (dotted line represented) which not allows the modification of the P_1 marking quantity, during the dynamic behavior of the model the token quantity of P_1 is not influenced [8], [14], [16].

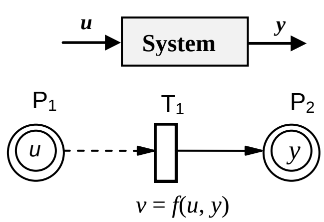

Fig.2 Continuous basic element of MPN model

If a single input arc is directed to P_2, we get:

$$v = \frac{dy(t)}{dt} = \dot{y}(t) \tag{7}$$

For a general place P_j with the marking m_j and i input arcs we get:

$$\dot{m}_j(t) = \frac{dm_j(t)}{dt} = \sum_i v_i \tag{8}$$

In the model behavior, continuous input and output transitions supply their part to increase or to decrease m_j. We can model different basic elements in this way. Even non-linear coupled subsystems can be described since v may be a non-linear function.

The main property of hybrid systems is the interaction between discrete and continuous subsystems. This interaction is also the main reason that classical methods of description are not sufficient in order to model such systems. So, the use of MPN allows to representing various interactions between continuous and discrete subnets.

Hence, the discrete control of continuous processes is described by discrete control places. They can influence the flow of continuous transitions in accordance with the firing rule (Fig.3).

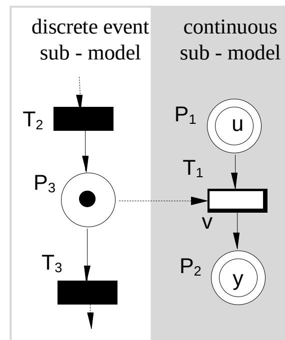

Fig.3 Discrete control of a continuous process

If the discrete place P_3 is marked, the transition T_1 is validated and it is active, with v flow rate, otherwise (when the P_3 is not marked) it is deactivated and the continuous process stops.

III. HYBRID OBJECT NETS

One of the obvious problems regarding to the modeling of hybrid systems is that HPN – but also the most mathematical, textual or graphical methods - are currently usable for small structures, with a simple topology, in which the behavioral properties of the model can be easily verified. Most often, the models of complex systems are unwieldy, large, difficult to understand or difficult to modify. Therefore a hierarchical concept - derived from the object-oriented paradigm - to organize and to synthesize the whole model is necessary.

In order to solve these handling problems arising from the system complexity, in [8], [9], [10], [11] is proposed a object oriented paradigm for the analysis of HPN, resulting in a new method to describe both continuous and discrete event systems with reduced effort. The main purpose of this approach was to encapsulate subnets within object frames which can interact with each other using defined interfaces only. In this way, one of the important advantages of this concept is the ability to describe a larger system, with a complex topology by the decomposition into interacting objects. Due to the properties of the synthesized objects, the modification of the whole system model could be easier achieved. The object-oriented concept unites the advantages of the modules and hierarchies and adds useful concepts like *reuse*, *encapsulation* and *information hiding*. In this way we get more flexibility.

To obtain a hierarchical structure of hybrid systems using HON, several steps are necessary to be followed, including some specific concepts. Then, always the attributes are represented at the model by continuous or discrete places with their markings. Information hiding is realized by encapsulation the detail topology of the net and by publishing selected places, using an interface. Abstraction is the step from a concrete net structure to a class; it is realized by filling the objects into a class hierarchy. Then, inheritance is the step from a class to a child class. If a new class is inherited from a class, it inherits the whole net structure, including the interface. Data exchange is given by the token flow between the objects. Reusing, the most important quality of object orientation, is given by inheriting or instantiating classes. Derived objects can be refined by adding places, transitions, arcs and objects, but no inherited element can be deleted.

Thus, every object is represented as a hierarchical structure, which contains three layers (Fig.4) [10], [11].

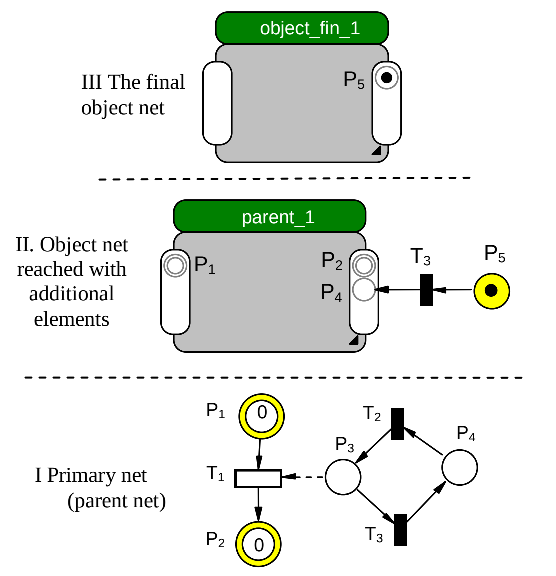

Fig.4 Hierarchical structure of the model

In the lowest layer, the parent net is represented. In the middle layer, the net inherited by the class is enclosed in an object frame. In this layer, various net elements and objects can be added, in order to modify the behavior of the object. In the top layer, the object frame is presented, which encapsulates the inner net structure of the object [8], [11], [12].

IV. CASE STUDY

In order to illustrate the resources of HPN formalism for hybrid systems representation, we are proposed in our paper a model for the control process of temperature during the heating of a room, using a bi – positional thermostat with anticipative resistance.

A. Physical System Structure

Fig. 5 illustrates the structure and the behavior of the system [14]. During the temperature control process, for different fluctuations against the reference value of the room temperature, the environmental thermostat switches between “on” and “off” positions and the anticipative resistance is connected into the circuit, bringing the heating of the bi – metal. In the mean time, the relay -7 turns on the supply valve - 4 and the gas – generating station is started. When the temperature rises and become greater than reference value, the anticipative resistance is disconnected.

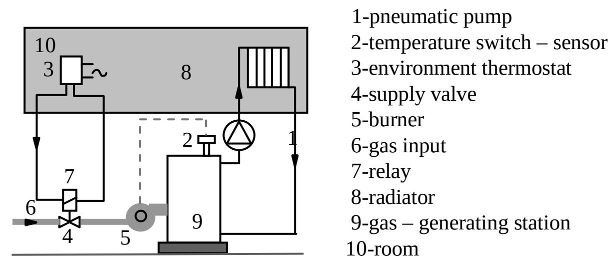

Fig.5 Physical system structure

In a qualitative representation of the control process (Fig.6), discrete variable „q” reaches two values (i.e. 1 and 0) according with one or other of steps θ_{t1} and θ_{t2}, considered as lower, respectively upon limits of the room temperature.

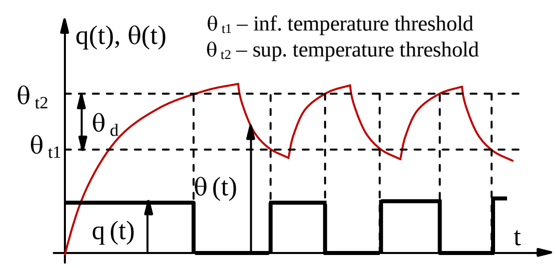

Fig.6 Illustrative for temperature control process

Temperature fluctuations induced by thermal inertia of the device can be reduced using one anticipative resistance R. Hence, the exceeding of θ_d value (Fig.6) is restricted and the thermostat switches “off” before the reference value of temperature being reached. In this way, the temperature oscillations can be diminished and, in the meantime, releasing frequency of the device can be increased.

B. Mathematical Model

The MPN model of the process was synthesized with a hybrid technique starting from a simplified mathematical model, established according to the Fourier’s low of the heating process, supposing a proportional dependence between the heat flows and gradients of the temperature. Thus, room temperature variation – θ can be expressed from a linear dependence between temperature values of the external environment - θ_e and the radiator - θ_r [14], [18]:

$$\frac{d\theta}{dt} = -c_1(\theta - \theta_e) + c_2(\theta_r - \theta) \tag{9}$$

Similarly, temperature of the thermostat - θ_{th} depends on heat changed between that device and the room and on the thermal energy – Q_1 produced by the anticipative resistance – R, during it connection to the power supply:

$$\frac{d\theta_{th}}{dt} = -c_3(\theta_{th} - \theta) + q(\theta_{th}) \cdot Q_1 \qquad (10)$$

On the other hand, the temperature variation of the radiator – θ_r depends on the thermal changing between the radiator and the room and, also, on the thermal energy – Q_2 produced by radiator itself:

$$\frac{d\theta_r}{dt} = -c_4(\theta_r - \theta) + q(\theta_{th}) \cdot Q_2 \qquad (11)$$

In (9), (10) and (11), c_i (i = 1, …4) denotes the global coefficients of heat transfer, and „q" is a discrete variable who can reaches only two different values (0 or 1), according to the hysteresis thermostat cycle (Fig.6) and controls the starting and the stopping process of the heating system [4], [14], [18].

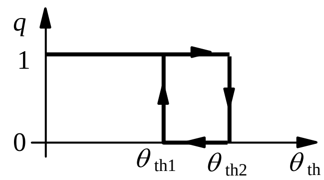

Fig.7 Hysteresis thermostat cycle

Considering as state vector of the system $\boldsymbol{\theta}^t = (\theta \quad \theta_{th} \quad \theta_r)^t$ and as input vector $\mathbf{u}^t = (\theta_e \quad Q_1 \quad Q_2)^t$, we can represent the simplified mathematical model of the process through a linear system of equations, in accordance to (1):

$$\begin{bmatrix} \dot{\theta} \\ \dot{\theta}_{th} \\ \dot{\theta}_r \end{bmatrix} = \begin{bmatrix} -(c_1 + c_2) & 0 & c_2 \\ c_3 & -c_3 & 0 \\ c_4 & 0 & -c_4 \end{bmatrix} \cdot \begin{bmatrix} \theta \\ \theta_{th} \\ \theta_r \end{bmatrix} +$$

$$+ \begin{bmatrix} c_1 & 0 & 0 \\ 0 & q & 0 \\ 0 & 0 & q \end{bmatrix} \cdot \begin{bmatrix} \theta_e \\ Q_1 \\ Q_2 \end{bmatrix} \qquad (12)$$

In (12), the variable q can reach two distinct values (Fig.7): 1 and 0 respectively. The switching of these two values causes two distinct operated services for the whole process: ON (for q = 1) and OFF (q = 0). The commutation of the hybrid system between its states is released when the state space vector $\boldsymbol{\theta}$ (more precisely it θ_{th} component) reaches for the first time the threshold value θ_{th1} (q = 0 and $d\theta_{th}/dt < 0$), then the other threshold value θ_{th2} (q = 1 and $d\theta_{th}/dt > 0$).

C. *Hybrid Petri Net Model of the Process*

The Petri Net model achieved for analyzing the control process of temperature was obtained starting at the previously observation in connection with the autonomous commutation of the system, due to hysteresis cycle threshold values θ_{th1} and θ_{th2} respectively. Hence, starting from the initial state, until the temperature of thermostat become θ_{th2}, the behavior of the system is described by (12) with q = 1. Then, during [θ_{th1}, θ_{th2}] interval of temperature, when the anticipative resistance R and the gas – generating station are turned off, the dynamic of the process is described by the same mathematical equations system, with q = 0. It is easy to observe that the model switches periodically, due to the achievement of θ_{th1} or θ_{th2} values.

Particularities of the mathematical model of the process, considered as a hybrid system representation, and the behavioral particularities due to a permanent switching between its states leads to a Hybrid Petri Net topology for the looking model. The commutation between the continuous space sates values of the system is induced by the occurrence of some external events (commands or disturbances). In order to synthesize easier a basic frame of the model in accordance with the behavior of the process, the main idea was to use all the facilities dues to MPN formalism.

Thus, the topology of the whole model (Fig.8) contains continuous sub-models (continuous Petri Nets) activated or no by a permanent interaction with a discrete control sub-model (a discrete Petri Net). For the continuous part of the model were used elements of continuous Petri Nets with variable firing speed of continuous transitions, dependent on continuous places marking. The discrete control sub – model, can be synthesized - for example - using T-timed discrete Petri Nets elements.

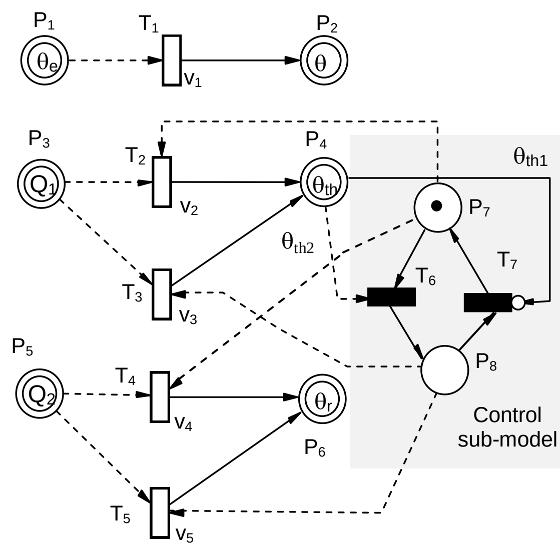

Fig. 8 MPN of the process

The model was synthesized considering the state equation of the state space model (12) and using each one of the basic elements for every row of the equation system. The coupled in – and outputs of the system were represented by arcs. The firing speed functions, assigned to the continuous transitions, can be gained line by line from the equation system, from both values (0 and 1) of q variable:

$$v_1 = -(c_1 + c_2) \cdot \theta + c_2 \cdot \theta_r + c_1 \cdot \theta_e \qquad (13)$$

$$v_2 = c_3 \cdot \theta - c_3 \cdot \theta_{th} + Q_1 \qquad (14)$$

$$v_3 = c_3 \cdot \theta - c_3 \cdot \theta_{th} \qquad (15)$$

$$v_4 = c_4 \cdot \theta - c_4 \cdot \theta_r + Q_2 \qquad (16)$$

$$v_5 = c_4 \cdot \theta - c_4 \cdot \theta_r \tag{17}$$

The initial values of the system are described by the initial values of the variables θ, θ_{th}, θ_r and θ_e, Q_1, Q_2.

Thus, the linear equations of the initial mathematical model were represented into continuous sub – net, composed of $P_1 \div P_6$ places and $T_1 \div T_5$ transitions. That sub-net models the behavior of the system between commutation processes. The changing of Petri net structure according a commutation process is obtained due to discrete sub – net (control sub – model : P_7, P_8 places and T_6, T_7 transitions), which activates or no – through test arcs – some of continuous transitions. In this way, the Petri net switches permanently between two similar structures, this behavior been induced thanks to the weight of arcs $P_4 – T_6$ (test arc with θ_{th2} weight) and $P_4 – T_7$ (inhibitor arc with θ_{th1} weight) respectively. Specification of all Petri net elements was made according to the hybrid model and, also, according to the evolution rules, due to the specific net formalism: P_1 is assigned with the value of external environment temperature, P_2 – controlled room temperature, P_3 – thermal energy produced by R, P_4 – the current thermostat temperature, P_5 – thermal energy due to the gas – generating station, P_6 – radiator temperature. The marking non – null of P_7 activates the firing of T_2 and T_4 transitions (q = 1) and, similarly, the marking non – null of P_8 activates the continuous firing of T_3 and T_5, for q = 0.

The behavior of the process follows – according to switching operations – two mathematical models, M_1 and M_2, obtained from (12) for the two values of q:

If $\theta_{th} < \theta_{th\,2}$ and $d\theta_{th}/dt \geq 0$
Then M_1 (for q = 1)
else
M_2 (for q = 0)
Until $\theta_{th} = \theta_{th\,1}$ and $d\theta_{th}/dt < 0$

Even for a less complex topology of Petri Nets, it is difficult to find always the most simply solution to synthesize a compact model, with a minimum number of elements and, also, with goods behavioral and structural properties. Moreover, the model achieved it is not unique. On the other hand, to verify the correctness of the solutions using analytical methods may be a very laborious process, especially for a large number of system states of or when the places marking of the model becomes very great. Therefore, both in the primary stage, when the model is developed and in the other stage, when the analysis of goods properties is started, various dedicated software tools with a friendly graphical user interfaces are used.

Thus, the Petri Net model of the process was developed using Visual Object Net++, a Petri Nets modeler tool, supporting mixed continuous and discrete event Petri-Nets. This software package offers to the user various facilities in the synthesis of the models, also in their behavioral properties analysis by on-line simulation (animation), even for the continuous and hybrid models with variable speed of continuous transitions [9], [10], [12], [13], [14], [19].

So, the Petri net model was developed and its behavioral properties were verified by on-line simulation using Visual Object Net++ tool [9], [10]. For a quantitative analysis, the values for global coefficients of heat transfer c_i (I = 1 ÷ 4) and, also, for Q_1 and Q_2 were the same used in [18]. The model allows a very easily modification of all initial variable values. Thus, the reference value of room temperature can be initially adopted and then eventually changed only by setting the threshold values of the thermostat.

Fig.8 shows a set of simulation scenarios obtained using the MPN model, for: $c_1 = 10^{-4}\ s^{-1}$, $c_2 = 5 \bullet 10^{-5}\ s^{-1}$, $c_3 = 5 \bullet 10^{-3}\ s^{-1}$, $c_4 = 10^{-3}\ s^{-1}$, Q_1 = 0,0132 °C/s, Q_2 = 0,06 °C/s, $\theta_{th\,1}$ = 20 °C; $\theta_{th\,2}$ = 22 °C, θ_e = 15 °C. The threshold values $\theta_{th\,1}$ and $\theta_{th\,2}$ of the thermostat were set so as to ensure a reference room temperature θ = 20 °C.

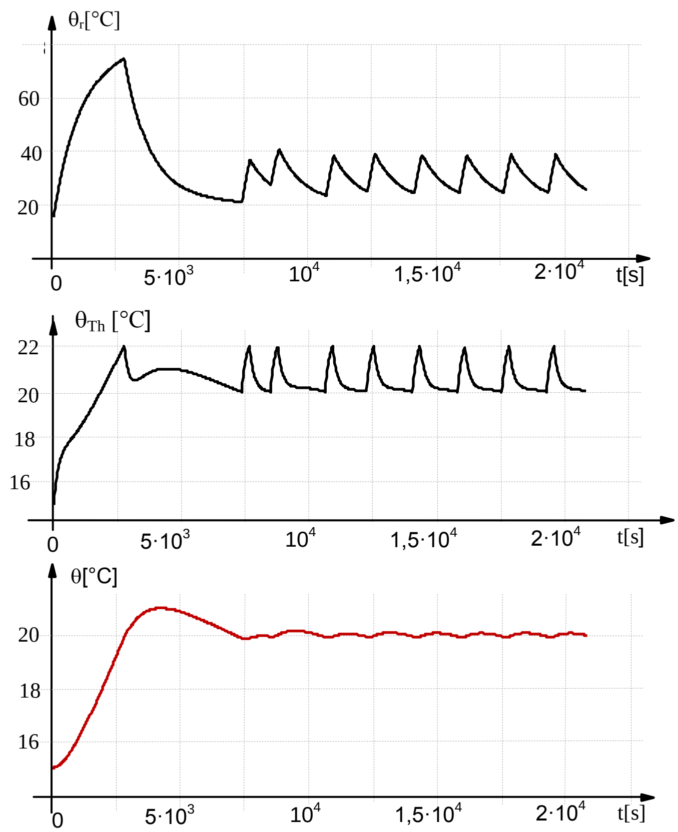

Fig.9 Simulation results of MPN model

All graphical representation were obtained by on-line simulation, using the Visual Object Net++ tool, considering that the initial values of state space vector are equals with the value of external temperature. Moreover, the effect of external perturbation on whole dynamic process was neglected, the process dynamic being roughly the same whit that of an isolated system. The system response confirms all the observations stated since the initial stage of model development about the great time constant value of the process and about the advantages due to use a thermostat with anticipative resistance for the temperature control.

The correctness of the model can be verified for many simulation scenarios, using the same MPN topology, in various environmental initial conditions at different controlled values of room temperature (Fig.10, Fig.11).

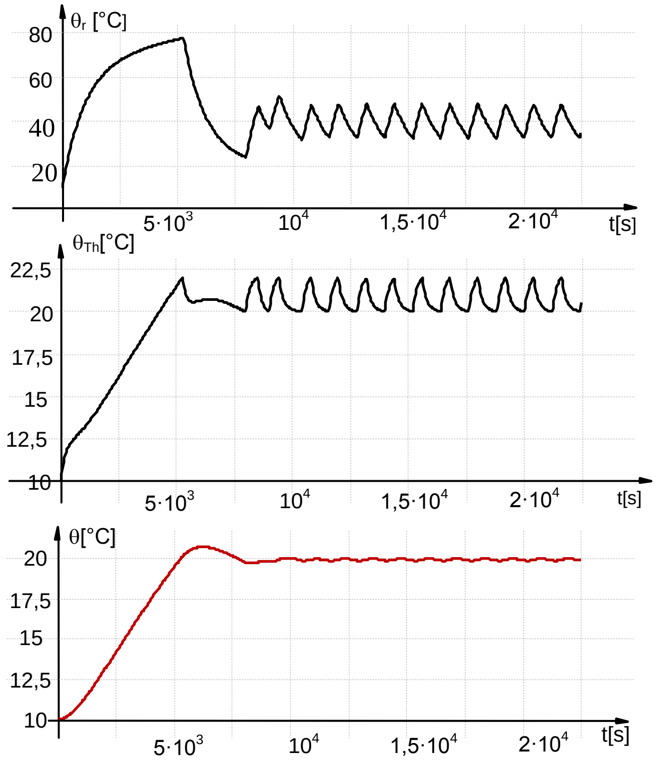

Fig.10 Petri net simulation results for $c_1 = 10^{-4}$ s^{-1}, $c_2 = 5 \cdot 10^{-5}$ s^{-1}, $c_3 = 5 \cdot 10^{-3}$ s^{-1}, $c_4 = 10^{-3}$ s^{-1}, $Q_1 = 0.012$ °C/s, $Q_2 = 0{,}06$ °C/s, $\theta_{th1} = 20$°C, $\theta_{th2} = 22$ °C, $\theta_e = 10$ °C.

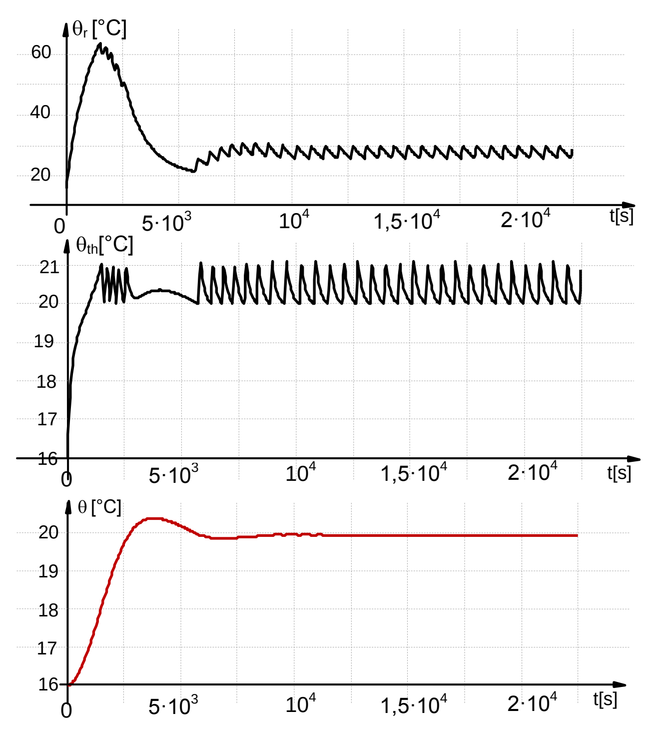

Fig.11 Petri net simulation results for $c_1 = 10^{-4}$ s^{-1}, $c_2 = 5 \cdot 10^{-5}$ s^{-1}, $c_3 = 5 \cdot 10^{-3}$ s^{-1}, $c_4 = 10^{-3}$ s^{-1}, $Q_1 = 0.0168$ °C/s, $Q_2 = 0{,}06$ °C/s, $\theta_{th1} = 20$°C, $\theta_{th2} = 22$ °C, $\theta_e = 16$ °C.

The model above can be used as a framework, able to purpose a quantitative evaluation of behavioral properties and, obvious, it is a modular structure which allows the adding of other Petri Nets sub - models (continuous, discrete or hybrid) in order to detect and analyze more precisely all the consequences of external perturbations on the process dynamic.

As noted, the MPN representation of the temperature control process analysis is a modular and flexible structure, providing the opportunity to be refined or modified according with various simulation scenario modifications. These behavioral changes can be easily caught by adding new elements (places, transitions or arcs) at initial model. Sometimes, the modification of arcs weight only is sufficient for adapt the model to new conditions.

Then, returning to our model, the results of simulation were obtained by neglecting of all external disturbing factors; otherwise, the time response of the system will not be the same, even if, qualitatively, the differences are not significant.

Let us consider, for example, that during the temperature control process, the room is no longer a perfect isolated system, and that the temperature can be modified also by the flow of inputs and outputs which introduces cold air in the room. To illustrate this new hypothesis, the main nucleus of the model can be preserved, and a hybrid sub-model can be added (Fig.12). It is mentioned again the fact that the model is not unique; each user can synthesize your own Petri Net model following a precisely or heuristic method to achieve a good process representation.

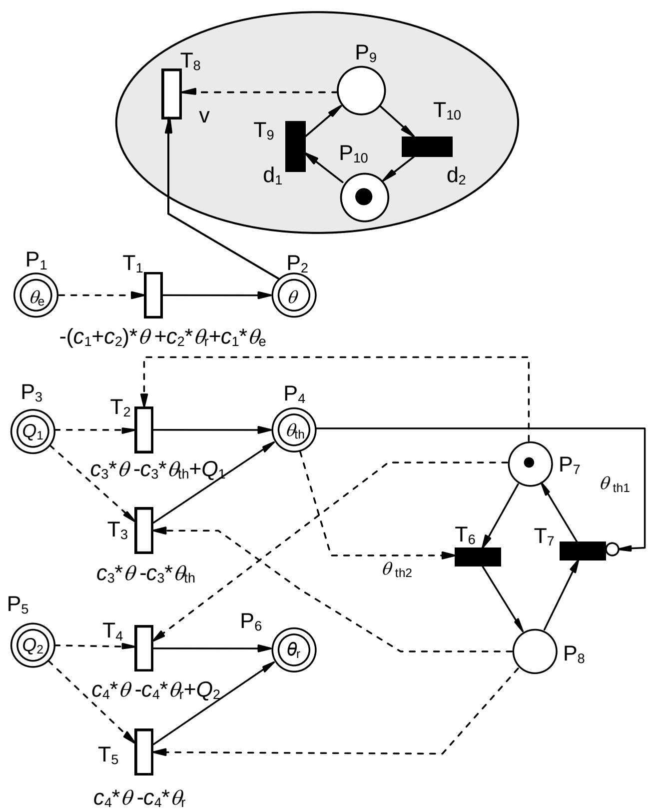

Fig.12 Illustrative at main MPN model modification

The new added sub-model is also a Hybrid Petri Net, consisting of T_8 continuous transition and discrete elements P_9, P_{10} and T_9, T_{10}. In the initial state, the token existing in P_{10} place validates the T_9 transition, which will be fired after the d_1 associated period of time. After firing of T_9 transition, the token will be transferred in P_9 place, authorizing the firing of continuous transition T_8 trough the test arc P_9 – T_8. The continuous firing of T_8 with v speed (with a constant or a variable value) leads to a decrease of continuous marking of P_2 place (the current value of room temperature) with the same v value. After the d_2 time associated to T_{10} transition, T_8 is inactive and the transition cannot be fired. Then, at the end of d_1 temporization the cyclical behavior of the model will be resumed. Periodic execution of T_9 and T_{10} transition may models the occurrence of some external events (deterministic or stochastic) which drives the process between its states (for example: the room door is closed during the d_1 temporization and it will be opened during the d_2 period).

Starting from the modified model (Fig.12) and according with above observations, Fig.13 and Fig.14 shows the time response of the process in for d_1 = 3000 s, d_2 = 60 s, v = 0,005 and for d_1 = 3000 s, d_2 = 60 s , v = 0,01 respectively.

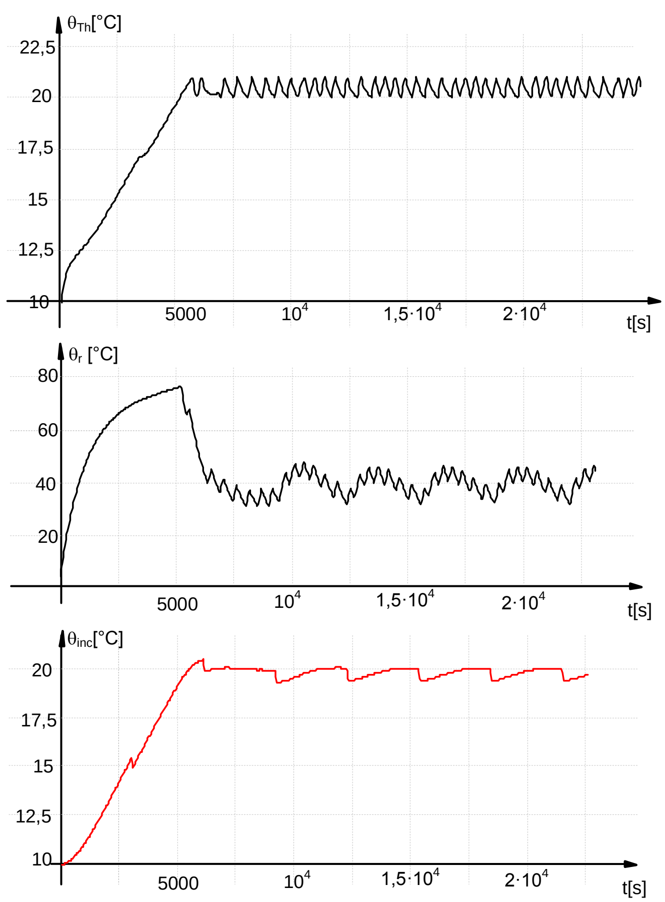

Fig.13 Petri net simulation results for $c_1 = 10^{-4}\ s^{-1}$, $c_2 = 5\bullet10^{-5}\ s^{-1}$, $c_3 = 5\bullet10^{-3}\ s^{-1}$, $c_4 = 10^{-3}\ s^{-1}$, Q_1 = 0.012 °C/s, Q_2 = 0,06 °C/s, θ_{th1} = 20°C, θ_{th2} = 22 °C, θ_e = 10 °C, d_1 = 3000 s, d_2 = 60 s, v = 0,005.

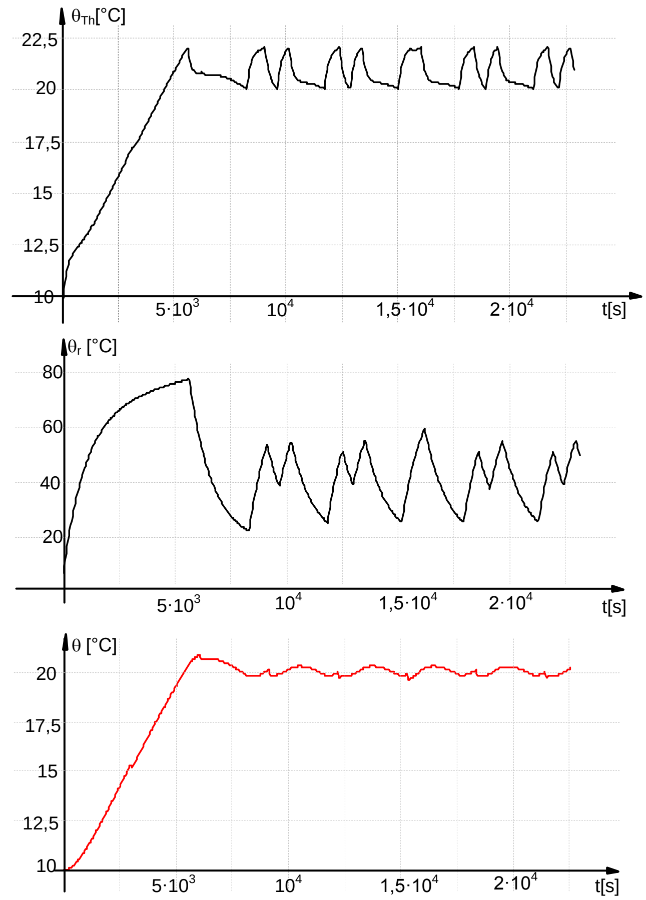

Fig.14 Petri net simulation results for $c_1 = 10^{-4}\ s^{-1}$, $c_2 = 5\bullet10^{-5}\ s^{-1}$, $c_3 = 5\bullet10^{-3}\ s^{-1}$, $c_4 = 10^{-3}\ s^{-1}$, Q_1 = 0.012 °C/s, Q_2 = 0,06 °C/s, θ_{th1} = 20°C, θ_{th2} = 22 °C, θ_e = 10 °C, d_1 = 3000 s, d_2 = 60 s, v = 0,01.

In order to obtain a more synthetic structure of the model, which can be easily integrated into another complex process representation, HON architecture of the main MPN can be reached, by following the mentioned principles of this paradigm.

Due to its capacities to handle hierarchical structures, in order to realize the Hybrid Petri Net Object model, Visual Object Net++ tool was used. The first step to achieving a Petri Net Object consist in selection of the places which will be located in the two published interfaces (input and output respectively) that allows the communication of the model with other models of the same kind (other Object Petri Nets) or different [9], [10], [11].

Hence, from the initial Hybrid Petri Net representation (Fig.8) P_1, P_3 and P_5 continuous places were included in the input interface and the others (P_2, P_4 and P_6) were placed in the output interface (Fig.15). All these places contain (by its marking) visible attributes of the model. The communication between the objects in a complex hybrid system is possible through interfaces. So, input places may be influenced from outside the object, and the output places can be utilized by other objects. On the other hand, both behavior of the parent net and its hidden attributes (caused by other model structural elements and by the connection between them) are encapsulated inside the object. In this way, the whole object is represented as a frame with a head line (which contains the name of the model – i.e. *Temp_contr_pr*) and interface channels.

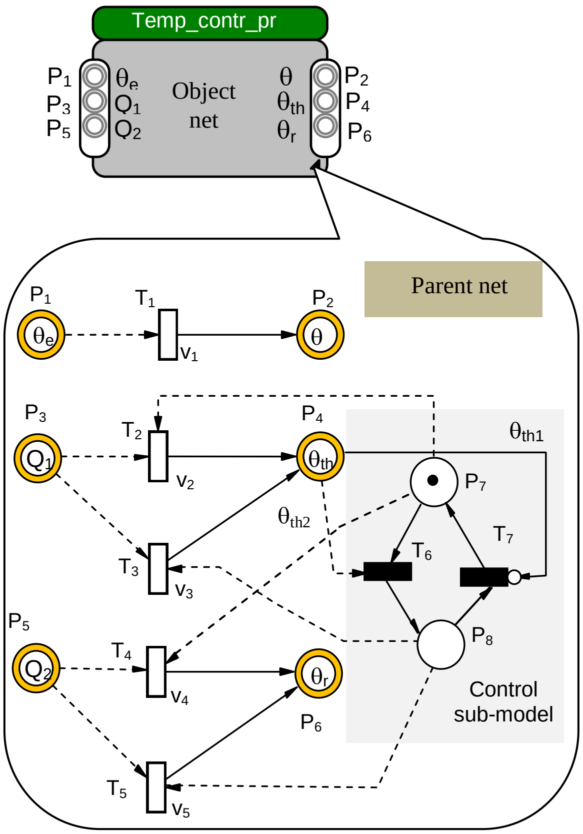

Fig.15 Hybrid Object Net of the initial model

The HON model achieved by this formalism can then be used to define a whole class of objects; the definition of a class starts with the modeling of a concrete object, thereafter it has to be generalized as a class [8], [9], [11].

The object models can communicate with each other or they structure can be filled by data exchange which makes it possible to model the behavior of complex systems; with respect to hybrid Petri nets rules, the data exchange is given by the token flow between the models (Fig.16).

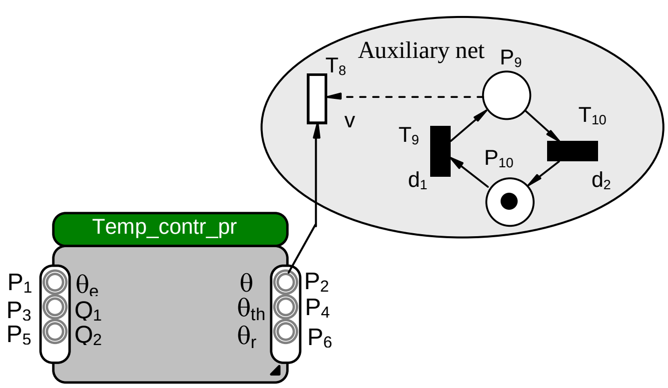

Fig.16 Data exchange between HON and auxiliary net

The auxiliary hybrid net which allows the analysis of the process in different simulation scenarios (Fig.12) was linked from the HON by direct connection between the P_2 output place and the continuous transition T_8, as in the initial topology; the whole structure of the model becomes more compact.

V. Conclusion

The main idea of this paper was to consider an temperature control process such a particular hybrid system, with autonomous state commutation and, starting of this point, to find a specific frame for describing his behavior. A mathematical model of such a process has thus to be a hybrid model involving discrete variables (integers or with a domain in a finite set) and continuous variables (real numbers). Both dynamics (discrete and continuous) have to be modeled: a discrete event based dynamics for discrete variables (sequence of operations) and a continuous time dynamics for continuous variables (differential algebraic equations). The general approaches have strong similarities with hybrid automata, but the discrete dynamics is represented by Hybrid Petri Nets in place of automata in order to address in an explicit way resource allocation policies. True concurrency is indeed required and the interleaving semantics of automata based approaches is not sufficient.

Adding new elements, with a grown power of analysis, may enrich the general model proposed. Hence, in the hybrid Petri Net model achieved, a first step for increasing the details was made due to variable delays associated to the discrete transitions. The concept "variable delay" is more often utilized in a determinist way. The delays are in any time defined in occurrence with an external event, a priori estimated, and generated by the decisional structure of the entire system. A major step to refine the initial model can be made using several powerful extinctions of basic Petri Nets models, i.e. stochastic, colored or fuzzy Petri Nets.

In this context, another problem, which leads to interesting results can be formulated is the controller synthesis of hybrid dynamical systems. Briefly, such approach is generally based on three steps: the behavioral description of the system (called open loop system) by a HPN model, the definition of specifications required on this behavior and, finally, the synthesis of the controller, which restricts the model behavior to the required one, using a controller synthesis algorithm. These algorithms use traditionally automata (finite state, timed and hybrid automata) because of their ease of formal manipulation; however, a model like HPN or MPN is preferred in the first step, of behavioral description.

Concluding, hybrid systems represent a highly challenging area of research that encompasses a variety of challenging problems that may be approached at varied levels of detail and sophistication. Also, it is very important to have good software tools for the simulation, analysis and design of hybrid systems, which by their nature are complex dynamical structures.

REFERENCES

[1] P. J. Antsalkis, W. Khone, A. Nerode, S. Sastry(Coordinators), *Hybrid Systems II*, Lecture Notes in Computer Science 999, Springer-Verlag, 1995.

[2] P. J. Antsalkis, J. A. Stiver, M. D. Lemmon, "Hybrid System Modeling and Autonomous Control Systems", Workshop *on Theory of hybrid Systems*, Lecture Notes in Computer Science 736, Springer-Verlag, pp.336-392, 1993.

[3] M. S. Branicky , "A Unified Framework for Hybrid Control", *Proc. 33rd Conference on Decision and Control*, Lake Buena Vista, Fl., USA, pp. 4228 – 4234, 1994.

[4] B. Cébron, "Commande de systèmes dynamiques hybrides", Thèse, Université d'Angers, France, 2000.

[5] R. David, H. Alla, "On Hybrid Petri Nets", *Discrete Event Dynamic Systems: Theory and Applications*, 11, 2001, pp. 9 – 40.

[6] R. David, H. Alla, Discrete, *Continuous and Hybrid Petri Nets*, Springer-Verlag 2005.

[7] I. Demongodin, N. T. Koussoulas, "Modelling of hybrid control systems using Petri nets", (Zaytoon, 1998), pp.330 – 337, 1998.

[8] R. Drath, "Objektorientierte Modellierung hybrider Prozesse – Vorstellung eines neuen Werkzeuges", *IWK* , 42, pp. 533 – 540, 1997.

[9] R. Drath, Tool Visual Object Net ++, Available: http://www.daimi.au.dk/PetriNets/tools

[10] R. Drath, "Hybrid Object Nets: An Object Oriented Concept for Modelling Complex Hybrid Systems", *Dynamical Hybrid Systems*, ADPM98, Reims, 1998.

[11] M. A. Drighiciu, "Petri Net formalism for a possible hierarchical structure of complex hybrid systems", *Proceedings of 6th International Carpathian Control Conference*, Miskolc – Lillafüred, Hungary, pp. 371- 376, 2005.

[12] M. A. Drighiciu, Anca Petrişor, "Hybrid Systems Modeling Using Petri Nets; A Case Study", *Buletinul Institutului Politehnic Iaşi*, Tomul LII (LVI), Fasc.5A, ISSN:1223-8139, pp.89-94, 2006.

[13] M. A. Drighiciu, Gh. Manolea, D. C. Cismaru, Anca Petrişor, "Hybrid Petri Nets as a New Formalism for Modeling Electrical Drives", *Proceedings of International Symposium on Power Electronics, Electrical Drives, Automation and Motion* – SPEEDAM 2008, ISBN: 978-1-4244-1664-6, IEEE Catalog Number: CFP0848A-CDR, Ischia, Italy, 11th – 13th June, pp.626-631, 2008.

[14] M. A. Drighiciu, Gh. Manolea, Anca Petrisor, M. C. Popescu, "On Hybrid Systems Modeling with Petri Nets", *Recent Advances in System Science and Simulation in Engineering, Proceedings of the 7th WSEAS International Conference on System Science and Simulation in Engineering (ICOSSSE'08)*, Venice , 21- 23 November, pp.73-78, 2008.

[15] E. Dubois, "Continuous Petri net with maximal speeds depending on time", *Proc.Of 15th International Conference on Application and Theory of Petri Nets*, Zaragoza, Spain, 1994.

[16] S. Engell, G. Frehse, E. Schneider (Coordinators), *Modelling, Analysis, and Design of Hybrid Systems*, Lecture Notes in Control and Information Sciences 279, Springer-Verlag, 2002, ISBN: 3-540-43812-2, pp.3-36.

[17] R. L. Grosmman, *Hybrid Systems*, Lecture notes in Computer Science, Springer, New York, Vol.736, 1993.

[18] J.-C. Joly, "Recherche de cycles pour une classe de systèmes dynamiques hybrides à commutation autonomes. Application à un thermostat à résistance d'anticipation", Rapport LAAS, CNRS, LISA, Université d'Angers, France, 2004.

[19] Anca Petrişor, N. G. Bîzdoacă, M. A. Drighiciu et al. "Modeling and Simulation of Walking and Climbing Robots based on Stable States Transition Approach as Control Strategy" , *Proceedings of the 12th WSEAS International Conference on Systems*, Heraklion, Greece, July 22-24, pp.216-221, 2008.

[20] M. C. Popescu, Anca Petrişor, M. A. Drighiciu, R. Petrisor, "Fuzzy Control of the Position for the Piston of an Industrial Robot", *Proceedings of the 12th WSEAS International Conference on Systems*, Heraklion, Greece, July 22-24, pp.222-226, 2008.

[21] C. Valentin-Roubinet, "Hybrid Dynamic System verification with Mixed Petri Nets", *The 4th International Conference on Automation of Mixed Processes: Hybrid Dynamic System*, Dortmung, Shaker Verlag, Aachen, pp.231-236.

[22] Emilia Villani, P. E. Miyagi, R. Valette, *Modelling and Analysis of Hybrid Supervisory Systems; a Petri Net Approach*, Springer-Verlag, 2007, ISBN 978-1-84628-650-6, pp. 1-13.

[23] J. Zaytoon, *Systèmes dynamiques hybrides*, Hermes-Science,2001.