Hybrid Petri Nets in Modeling the Packing Processes: Case Study

Mircea Adrian Drighiciu, Daniel Cristian Cismaru
University of Craiova / Faculty of Electrical Engineering, Craiova, 107 Decebal Blvd., Romania
adrighiciu@em.ucv.ro

Abstract — **In many cases, the packing processes are considered as hybrid systems, consisting of a set of workstations, where various components must be processed, and at least one flexible transport system used for loading and/or unloading operations. This paper is focused to a Hybrid Petri Nets approach for modeling and simulation the behavior of such systems, considered a hybrid dynamic structure, with both continuous and discrete components interacting. For heuristic representation of the model, several sequences and rules were however followed. For the case study, starting from the structure of the system, in order to achieve a primary topology of the model, a bottom-up synthesis technique was used, which allowed us to obtain a basic version, consisting in several sub-models of related physical subsystems of the packing station. Aggregation of these partial models was made in accordance with the interactions of the physical elements of the whole system, laid down in the operation protocol. Hence, into model topology various elements characteristic to generalized Petri Nets were used, with inhibitor arcs and test arcs, mainly aiming to reduce the complexity of the whole model. The validation of the model was done through on-line simulation, in various scenarios, under the Visual Object Net++ tool, which offers multiples facilities for the analysis of behavioral properties in various real operating conditions of the real physical system. Starting on the modular structure of the whole model, consisting in several sub-models of the same type, the authors obtained a multilevel architecture, using Object Petri Nets paradigm for representation and exploiting their analysis potential.**

Keywords* — *discrete event drive systems; hybrid systems; Petri Nets; simulation

I. INTRODUCTION

In the most known approaches, packing stations are high-speed systems, consisting in a set of workstations for processing various (same or different) parts of products and a flexible transport system. In order to obtain a final product in such working process (a sequence of operations), every part follows a route through the set of system resources, according to a pre-established schedule.

Traditionally, such a system is either modeled as a continuous – state system or a discrete – state system driven by time or asynchronous external events. Various artificial structures can be classified as hybrid systems, including logic – based switching control systems, intelligent transport systems, flexible manufacturing lines, batch processes and many other electromechanical systems (power electronics, robotics, flexible manufacturing systems, control of the electrical drives, hydraulics and pneumatics systems etc.) [1], [2], [5], [6], [7], [8], [18], [19], [23], [24], [25], [26].

Hybrid Petri Nets is a powerful tool, which inherits all the advantages of the Petri Net model such as the ability to capture behaviors including concurrency, synchronizations and conflicts. In this paper it was considered a modified Hybrid Petri Net model whose continuous part was represented by continuous Petri Nets entities and its discrete parts were represented by a timed Petri Net elements [5], [8], [16], [17], [22]. Our attention was limited to build a hybrid model for an particular automatic packing system, then to verify its behavior properties by simulation in order to achieve a hierarchical structure of the model, using the Object Petri Nets paradigm.

II. CASE STUDY

A. System Structure

The analyzed packaging system allows the automatic arrangement of margarine packets into cardboard boxes, by multiple layers. Automatic packing workstation is a complex structure containing components of an electric drive system (ED) and electro-pneumatic systems (EPS), (Fig.1), communicating with each-other via control system (CS). It receives all state information of the process provided by the sensorial system, and commands the execution elements of EPS according to the proposed schedule.

The ED contains an electric motor, which operates in steady-state a single way conveyor – C (Fig.2.a). It was assumed that the speed of the conveyor – for one simulation scenario – has a constant value. The pneumatic drive system has a basic structure, consisting of single or double action cylinders, single or double command valves and flow control devices (Fig.2.b).

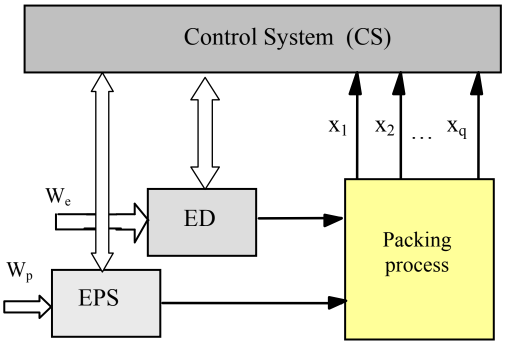

Fig.1. The structure of the packing system

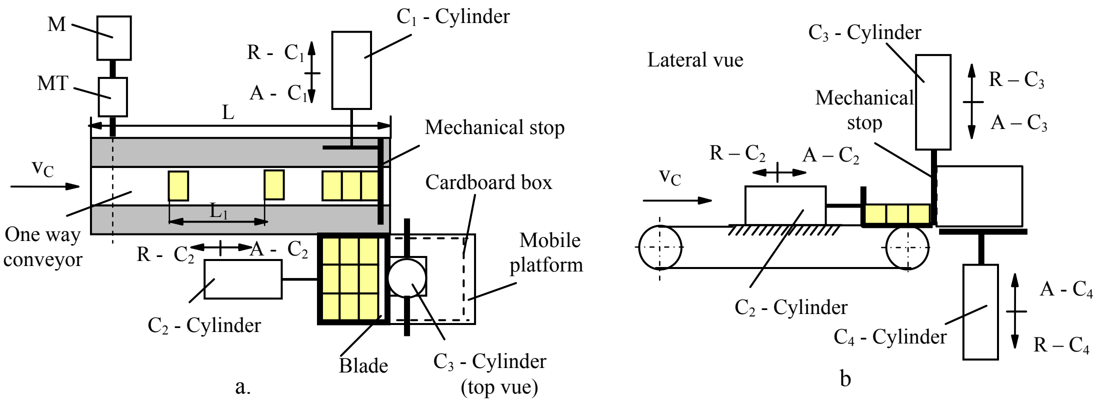

Fig. 2. Physical structure of the system: a) Top vue; b) Side vue.

An operating cycle consists in the fallowing sequences: margarine packages are transported to the packing station by the conveyor system, arranged at a distance from each other. At the right extremity of the conveyor, a mechanical sensor stops the packages and allows - in the meantime – their accumulation, without stopping the conveyor. At the moment in which three packets are stored at the right extremity of the conveyor, a counter device commands the advance of C_1. Its piston pushes those tree packages to the mobile platform, and then returns immediately at the initial position. In the same time, another device counter is increasing (for the counting of the package rows on the mobile platform). When the first layer of the mobile platform was formed (nine packages precisely), the cylinder C_2 pushes them into the cardboard box and remains in this final state. In the same time, C_3 moves down and assures that packages are stored into the box when the C_2 returns. Then the C_3 returns at the initial position and the mobile platform is descended one step by cylinder C_4. Then, a new cycle may be proceeded in the same sequence, until tree layers are formed into the cardboard box.

B. Petri Nets – a overview

Petri Nets (PN) has been extensively used to model and analyze manufacturing systems. One of the major advantages of using PN models is that the same model can be used for the analysis of behavioral properties and performance evolution, as well as for the systematic construction of discrete event simulators and controllers [1], [2], [3], [4], [5], [6], [7].

Known initially as an analysis tool of discrete event drive systems PN gained – through subsequently developments of their own formalism - unique strength in hybrid systems representation and in study of qualitative and quantitative their properties. Hence, Petri Nets is a powerful tool in the modeling of hybrid systems with autonomous commutation of the model generated by a hysteresis phenomenon through a particular Petri Nets structures, called Modified Hybrid Petri Nets (MHPN). They are 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 [9], [10], [11], [12], [13], [14], [15].

MHPN is a combination of ordinary and continuous PN. This model can treat integer variables together with real variables and symbolic variables usually encountered in other models of hybrid systems. It can inherit all the modeling facilities of PN such as the ability to capture concurrency, synchronizations and conflicts, allowing to model systems with continuous flows and linear evolutions in an intuitive way [5], [6], [7], [11], [12], [21]. But when a PN contains a large number of tokens, the number of reachable states explodes and this is a practical limitation of the use of this model.

An autonomous Hybrid Petri Net, [5] may be defined as a sextuple HPN = {P, T, Pre, Post, $\mathbf{m}_0$, *h*} such that: P is a finite, not empty, set of places: P = {P_1, P_2, …P_n}; 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 : $P \times T \rightarrow \mathcal{R}^+$ or $\mathcal{N}^+$, is the input incidence mapping; Post : $P \times T \rightarrow \mathcal{R}^+$ or $\mathcal{N}^+$, is he output incidence mapping; $\mathbf{m}_0 : P \rightarrow \mathcal{R}^+$ or $\mathcal{N}^+$, is the initial marking. Thus, 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 (pace or transition) to a continuous node (transition or place), [4], [5], [6] (Fig.3).

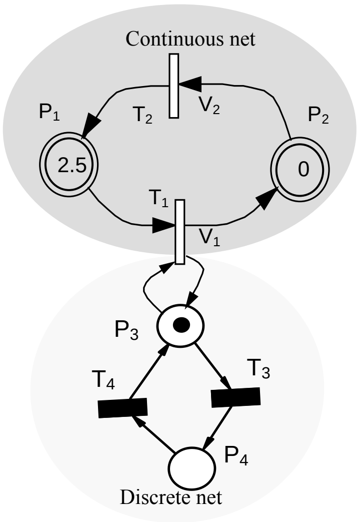

Fig. 3. Hybrid Petri Net structure

The continuous places 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. Both continuous transitions are enabled, and it can be fired [5], [6], [18].

Figure 4 shows a modified HPN. In this case the continuous part represents a production system. The transition T_1 corresponds to the working process of a machine, continuous production or approximation by a continuous flow of a discrete production. When the output buffer reaches a certain level (10,8 on Fig. 4), production stops (the transition T_2 id disable). This transition takes priority over the continuous transitions.

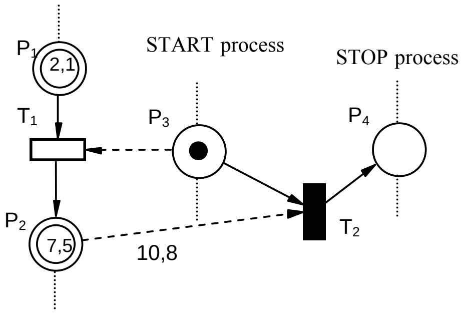

Fig .4. Modified Hybrid Petri Net structure

C. *Petri Net model of the packing system*

According to the proposed scenario, a primary model was made, which consist in a modified discrete Petri Net (Fig.5). Beside of basic elements, this model contains several extensions (test arcs: $P_8 – T_4$, $P_{11} – T_8$, $P_6 – T_{11}$ and inhibitor arc P_{19} – T_1) which increases the power of representation.

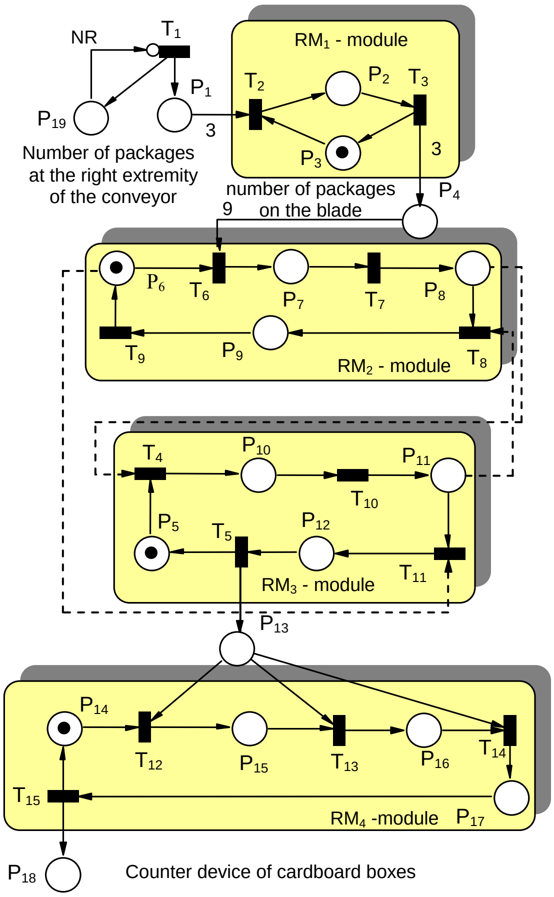

Fig. 5. The primary model of the packing system

The supply conveyor is a hybrid system and it has been represented by a MHPN model (Fig. 6).

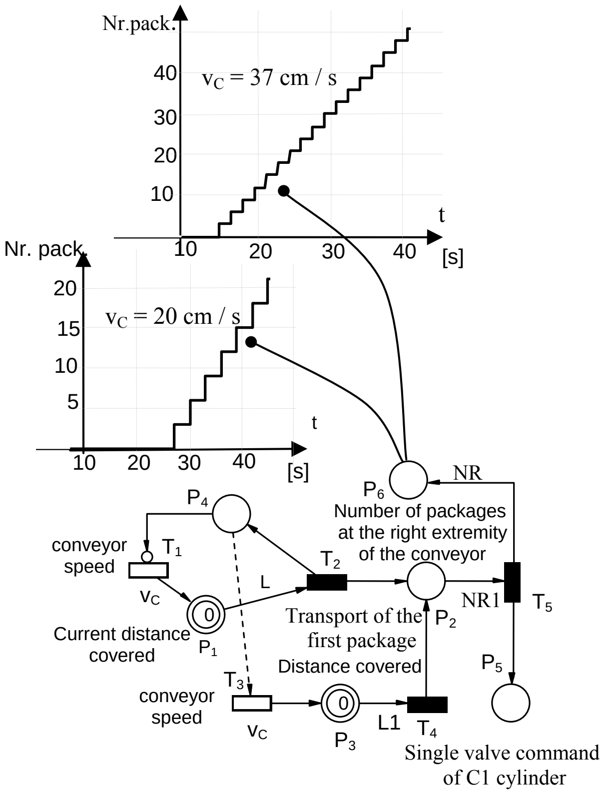

Fig. 6. Hybrid model of the conveyor

As was shown, the main topology of the packing station (Fig.2) emphasizes a transport system (single product conveyor), a modular manipulator (consisting in 4 separate electro-pneumatic modules) and an output buffer, disposed around to the packing area. For this configuration, it's obvious that the productivity of the packing system depends on the conveyor speed and on normal operation of all modules. The real value of the speed is necessary to be correlated with the operated time of the first electro-pneumatic module, also with the cadence of the rest of entities upstream disposed.

For the model's validation, also for analysis of its behavioral properties, and because of the complex topology caused by the particularities of the Petri Net model dynamics, it was necessary to use some specialized and dedicated software tools. Starting from the same topology of the net, the selected software tool allows us to modify the values of the transition's firing speed, providing a greater flexibility of the entire model and various graphical on-line representations of the number of parts transported by the conveyor (Fig.7). [8], [11], [12].

In order to obtain an adequate and refined model, a top – down technique can be used. Thus a transition and/or a place of the basic model are replaced with another, detailed sub-net, and so on, until the model satisfies the technical requirements.

The Petri Nets structures allows the user to observe the system's real behavior in an inadequate working situation, caused by the temporary malfunctioning of some subassemblies, by bad implementations of the control procedure or because of wrong management of its available resources. When a malfunction appears, the control system must allow the user to search and find it; in the meantime, the system remains into an intermediate stand-by state until the fault is removed and, then it will be restarted with or without reboot of its equipment.

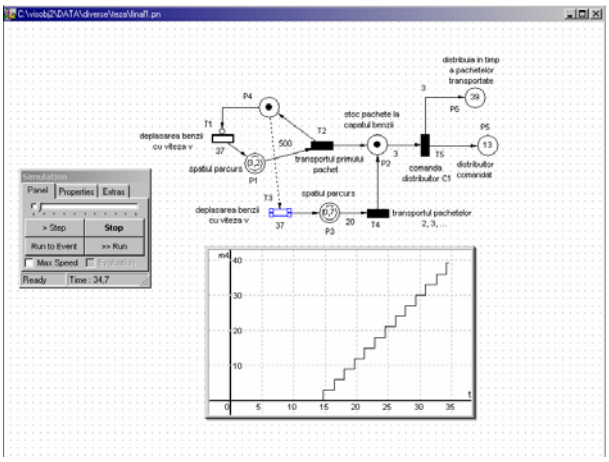

Fig. 7. Simulation's results for the conveyor speed $v_c = 0{,}37$ m/s

At the model synthesis should be considered that the fault system status is caused by the occurrence of a stochastic external event, while the whole dynamic of the model follows general deterministic rules. Even the events which leads the system between its states have an asynchronous distribution, the dynamic of the whole structure (hybrid system - in essence) is achieved in a deterministic manner. Thus, each external event which determines the system's dynamics has a well-defined position in relation with others, integrated in a structure similar with task schedulers, found in the real-time command systems. The model topology must allow parallel evolutions of the marking, similar with a real system behavior. The best results given by the malfunction's analysis and their impact on the control architecture's reorganization are obtained by using Stochastic Petri Nets models or by using a certain stochastic temporizations, similar somewhat with Markov's chains.

In the deterministic models, a system's malfunction is represented by a configuration which makes the evolution into complementary states possible, excluding themselves, being therefore unable to be simultaneously touched (structural conflict). The system can evolve either in one state (specific to a normal behavior) or in another state (malfunction), (Fig. 8)

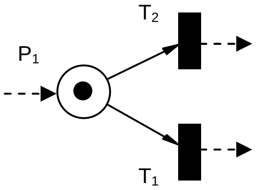

Fig. 8. Explanatory for fault modeling (free-choice net).

From the current state (place P_2 marked with a token) both transitions are fireable, but only one will be executed. To select the transition that will be executed, one doesn't need to follow a priorities rules nor precedent restrictions: if T_1 corresponds to a normal model evolution, the T_2 transition is associated to the event that leads to a malfunction.

Also, it's important to specify that after the fault state is eliminated, the affected equipment can be either returned to the initial's state (Fig. 9 a) or its dynamic can evolve from the current state (the state in which it was after the malfunction appears), (Fig 9 b).

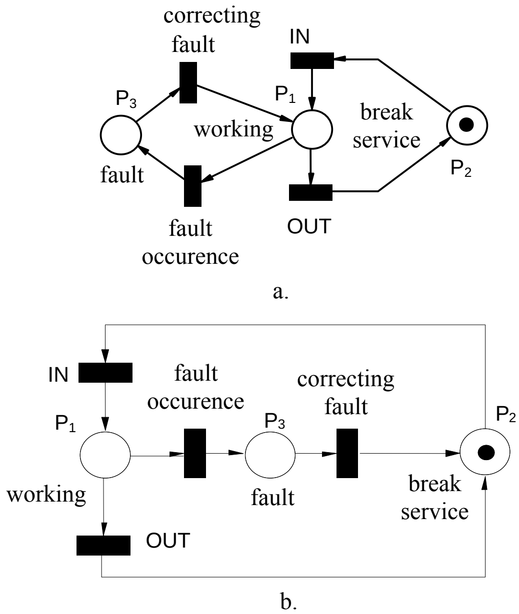

Fig. 9. Modeling of failures: a) The system returns in initial state after fix failure; b) The system will work from the current state.

Moreover, each component can host a fault, leading to a hold/stop of the system. So, each Petri Net model must be able to reproduce the real system's flexibility using their inner structure. An important malfunctioning zone is also the junction point of the modules: when a command is issued or when they synchronize.

A situation which forces the temporary stop of margarine packs supply conveyor is the one in which the control signal given by NR_1 does not excite the distributor's coil associated with cylinder C_1 and it remains in initial's position Fig. 10.

The structural conflict is created by P_3's place and its output transitions T_9 and T_3. After placing in P_3 three tokens –corresponding to the transported packages - T_3 and T_9 are validated. It can be executed either T_9 (distributor's command) or T_3 (the fault appearance). In this latter case, P_4 will be marked and allows the stop command of the conveyor supply (the T_5 transition will be executed). Once the fault was fixed, the system returns to the state prior to the malfunction. If the system's dynamic continue without any fault, the T_9 transition is executed, the P_3 becomes unmarked place, authorizing T_6's transition (start of the conveyor). If the fault state continues, the token from P_3 returns to P_4 and the systems reaches a stand-by state.

All in all, by searching and finding possible fault scenarios, the model becomes a complex Petri Net structure.

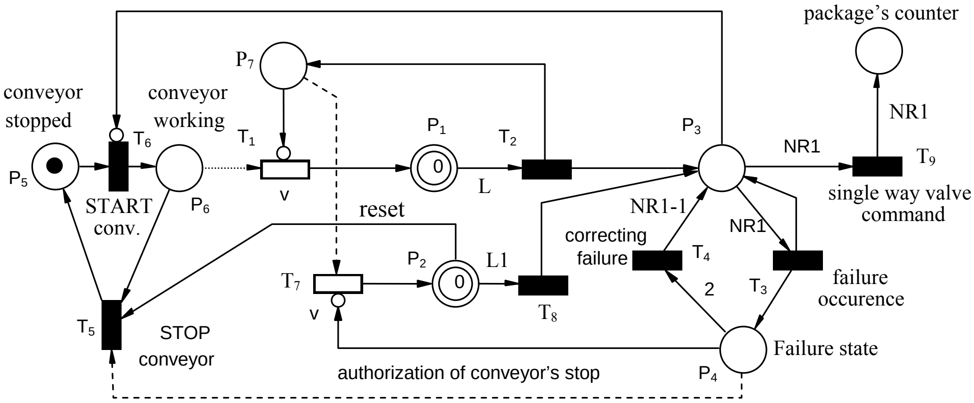

Fig.10. Conveyor model reached in order to specify working failures

In order to achieve the whole model of packing station, several sub-models of electro-pneumatic drives were interconnected. Usually, a pneumatic circuit contains a directional control valve which when is turned on switches and commands the forward stroke of an actuator with single or double action.

Directional control valves are commutation devices operating in pneumatic power circuits. Generally, they are used in addition with single or double acting cylinders, for its command. The model of a directional control valve is a synchronized PN (Fig.11): their places represent suitable conditions for states evolution and its transitions are fired at the occurrence of external events (sensors outputs and external command signals), [14], [15].

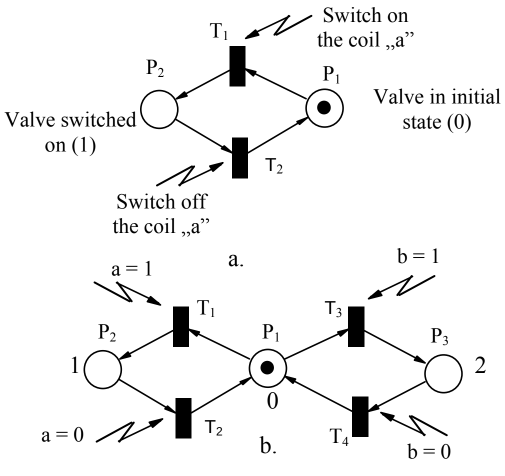

Fig. 11. PN models of directional control valves :
a) with 2 states ; b) with 3 states.

Each of pneumatic actuators can be represented as a discrete synchronized PN (Fig.12.) in which: P_1 denotes the initial state of the actuator (S_1 = 1), P_2 denotes the forward stroke until S_2 is activated, P_3 shown the extreme position of the actuator and finally, P_4 denotes its comeback stroke. The start of forward action is given by the firing of transition T_1, after its directional control valve has been turned on; then, after activating the sensor S_2 (end of stroke) transition T_2 is fired and P_3 in marked with one token and so on, until the entire forward – return cycle it's done.

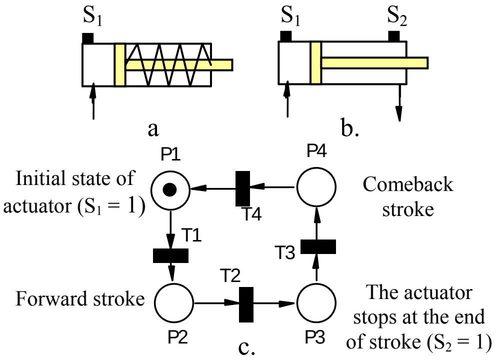

Fig. 12. Explanatory for the actuator PN model:
a) single acting actuator; b) double acting actuator;
c) The PN model.

Based from the above sub-models, the pneumatic actuator with its directional control valve have been shown as a synchronized PN (Fig.13.).

The synchronisation between the PN transitions were indicates by test arcs (dotted-line represented). Hence, one output transition of a test arc will be fired only the marking of its source place becomes greater or equal with the weight of the arc connecting them (Fig.13.). The test arcs do not realize tokens transportation between the places of the PN through the transitions connected by this [9], [10], [11], [12], [13], [14].

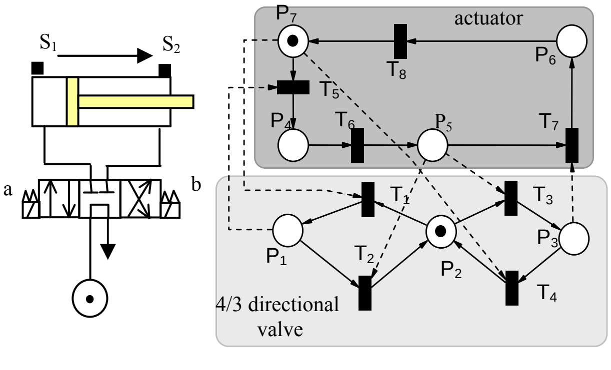

Fig. 13. PN model of subsystem actuator, 3 states
directional control valve

Using this model's synthesis technique, all modular sub-models of the subsystems that compose the packing station can be rejoined in a unique and refined structure (Fig. 14).

In order to obtain the Hybrid Petri Net model's topology, and for its validation in various simulation scenarios, Visual Object Net ++ software tool was used.

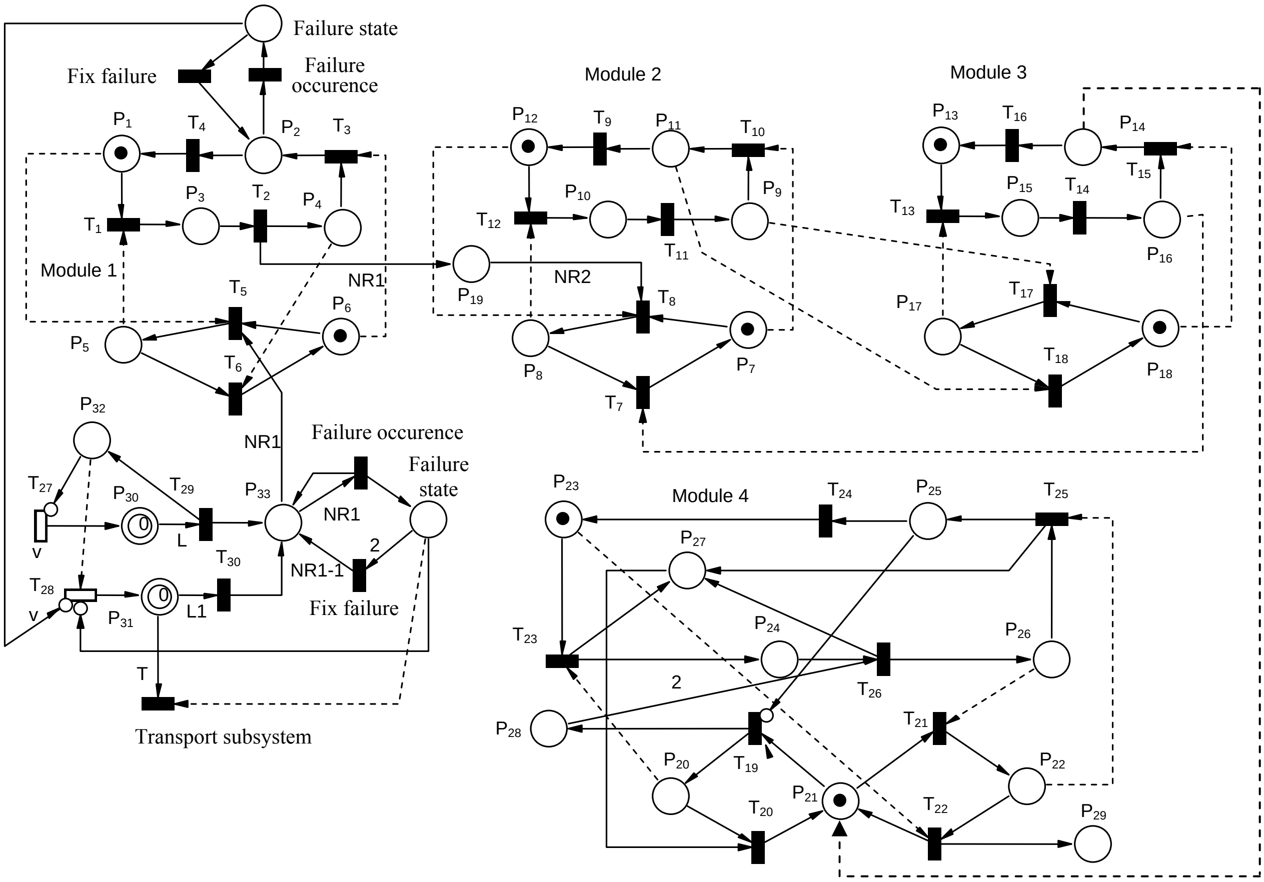

Fig.14. The Hybrid Petri Net model of packing system, enriched with failure sub-nets.

D. *The Object Net paradigm*

The Object Net paradigm allows representation of a system with a large and complex topology in a modular way, replacing the sub-models synthesized by different techniques with some entities represented as interacting object-models. Due to the objects properties, changing of the general system model could be easier achieved, because the object-oriented concept combines the advantages of the modular representations and hierarchies and adds useful new concepts (such inheritance, reuse, encapsulation, information hiding, data exchange etc.). In this way the flexibility and the versatility of the model are increased and the entire model becomes a hierarchical structure [11], [12], [21]

In this manner, every object can be represented as a hierarchical structure, which contains - generally - three layers (Fig.15). In the lowest layer, the parent net is represented. In the middle layer, the net inherited by the class was 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 [11], [12], [20], [21], [22].

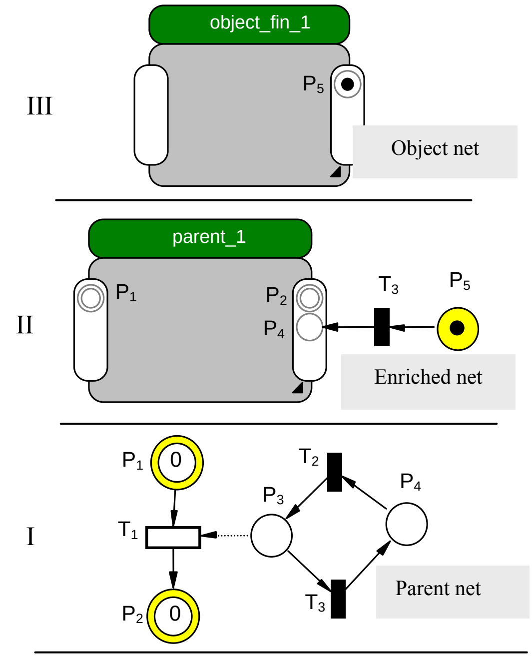

Fig. 15. Hierarchical structure

For example, the initial structure of hybrid model of the conveyor system was encapsulated into "Conveyor" object, (Fig.16).

Only the P_1 and P_5 places (interface places) are accessible for a possible connection with others modules, similar or different, from the superior level (level II). In the same way, each subsystem of the packing station can be represented by a hybrid Petri net object, resulting at the final a modular and flexible structure. All reached objects can communicate one with each other by data exchange, given by the token flow between its.

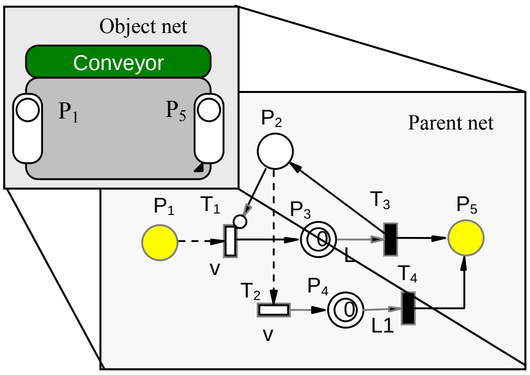

Fig. 16. The Conveyor object net.

III. CONCLUSIONS

The goal of this paper was to propose an Hybrid Petri Net model for representation and behavioral analysis of an automatic packing system, considered such system as a hybrid structure. This approach is based to the fact that a hybrid system contains in it topology discrete and continuous subsystems which interacts. This remark has allowed us to consider sometimes that the packing process has a similar dynamic with a batch process, in which the material is operated by finite quantities (the batches); at any time, an integer number of batches are in operation at many locations in the plant.

Used rather heuristic techniques in order to synthesize the model, first, an autonomous PN was achieved whose the behavioral properties have been analyzed; then in addition of delays to the transitions, the model was converted into a T-timed PN model. Taking into account the continuous dynamic of the transport system was achieved a MHPN model using the facilities of the software tool. For this model the behavioral properties was verified by on-line simulation in various scenarios. Finally, the model is the result of the aggregation of partial sub-models. It is obvious that the general model is not a unique structure, the MHPN formalism – in conjunction with the facilities offered by the Visual Objet Net ++ software platform - allowing multiple solutions.

The most mathematical, textual or graphical approaches to describe hybrid systems are currently usable for small examples, but models of complex systems are unwieldy. Therefore a hierarchical concept to structure a model is needed. Thus, using the Object Net paradigm, the authors were represented each of the subsystem of the packing station by a Hybrid Petri net Object, finally resulting a modular and flexible structure. Due of the properties of these modules, the objects obtained can communicate with each other by data exchange, given by the token flow between the objects.

Starting from this point, the next step is the implementation of an control-command structure and a controller set of rules based on the Petri Net model for the real system. This will be the subject and the purpose of the authors in a future paper.

Received on July 25, 2015
Editorial Approval on November 16, 2015

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.Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

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

[8] I. Demongodin, F. Prunet, "Batch Petri Nets", *Proceedings of 7th Annual European Computer Conference*, Paris, 1993, p.29 – 37.

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

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

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

[12] 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.

[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] M.A. Drighiciu, Anca Petrişor, " A Modified Petri Net Framework for Hybrid Systems Modeling", *Analele Universităţii din Craiova; Seria Inginerie Electrică,* Anul 35, No.35, ISBN:1842-4805, Editura Universitaria Craiova, pp.158-165.

[16] 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.

[17] 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.

[18] J. M. Flaus, “Hybrid flow nets for batch processes modelling and simulation”’ *Proceedings of 2nd IMACS MATHMOD Conference*, Viena, 1997, p.211 – 216.

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

[20] L. A. M. Riascos, L. A. Moscato, P.E. Miyagi, “Detection and Treatment of faults in Manufacturing Systems based on Petri Nets”, *Journal of the Brazilian Soc.of Mech.Sci.&Eng.*, Vol.XXVI, No.3, July – September, 2004, p.280 – 289.

[21] A. T. Sava, H. Alla, “Combining Hybrid Petri Nets and Hybrid Automata”, *IEEE Transaction on Robotics and Automation*, Vol.17, No.5, October 2001, p.670 – 678.

[22] 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.

[23] 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.

[24] J. Zaytoon (Coordinator), *Systèmes dynamiques hybrides* , Hermes-Science, 2001.

[25] A. Zimmermann, G. Hommel, “Modeling and evaluation of manufacturing systems using dedicated Petri nets”, *International Journal of Advanced Manufacturing Technologies* No.15(2),p.132–138.

[26] W.M. Zuberek, “Timed Petri Nets in Modeling and Analysis of Manufacturing Systems*“, Emerging Technologies, Robotics and Control Systems*, Vol.1 International Society for Advanced Researches, Palermo, Italy, 2007, ISBN 978-88-901928-1-4, p. 66-73.