A timed Petri net based approach for the design of industrial processes

Marco Bortolini*, Mauro Gamberi, Alessandro**
Persona, Francesco Pilati**, Alberto Regattieri***

**Department of Industrial Engineering, University of Bologna, Viale del Risorgimento 2, 40136,*
Bologna – Italy (marco.bortolini3@unibo.it, alberto.regattieri@unibo.it)

***Department of Management and Engineering, University of Padova, Stradella San Nicola 3,*
36100, Vicenza – Italy (mauro.gamberi@unipd.it, alessandro.persona@unipd.it,
pilati@gest.unipd.it)

Abstract:

The effective design of industrial processes represents a key asset to compete and to create a sustainable value for each company. Such a complex issue needs to integrate technological, economical, environmental and safety constraints and opportunities to be mixed together through a win-win strategy. The development of useful, easy-use and flexible approaches to analytically support both the researchers and the practitioners in the industrial process design, development and simulation is expected and strongly encouraged.

Purpose

This paper focuses on the design and simulation of industrial processes adopting a timed Petri net based approach. Considering a product flowing the manufacturing/assembly area and visiting a set of resources according to its working cycle, a timed Petri net design strategy is presented together with the dynamic analysis of the net to predict the process key performances. The resource peculiarities, e.g. availability and conflicts, are, further, considered. A realistic case study, taken from the automotive sector, is, finally, presented, to exemplify the proposed approach.

Methodology and approach

Starting from an overview of Petri nets and their fundamentals, the paper investigates an approach for their effective adoption in the industrial process design. An industrial and operative perspective is adopted, while a realistic case study is proposed to highlight the applicability of such a tool in common operative contexts. The manufacture process of high value components for sport-cars requiring dedicated resources and auxiliary devices is presented and fully discussed. Starting from the production process analysis, to identify both the production features and the required resource characteristics, the timed Petri net is designed adopting the most suitable approach, i.e. physical vs. functional. Given the net, its simulation and analysis is faced to obtain useful indices about the production process performances, e.g. throughput, required resources, resource availability, waiting times and buffer dimensions, etc. Such analysis outcomes represent useful guidelines to drive possible design or re-design actions.

Originality

Petri nets and, particularly, timed Petri nets represent a diffuse modeling language for the description of event-driven distributed systems. Their adoption in the automation control theory is known and widely discussed by the literature, together with a wide set of customization opportunities. On the contrary, minor attention is paid to study the applicability of this modeling language in the industrial sector to support the design of manufacturing and assembly processes with a deep focus on the operative context features and on the effects in terms of production efficiency, times and costs. This paper tries to overcome such a weakness focusing on the applicability of timed Petri nets to the industrial sector. Both the approach input and output are oriented to drive the process design, while the realistic case study exemplifies the opportunities of applying a timed Petri net based approach to the design of the production system for automotive high-value parts.

Keywords: timed Petri net, industrial process design, resource planning, simulation, automotive sector.

1. Introduction

The increasing complexity of products and the challenges coming from the global market scenario force all companies to make continuous efforts in the design and optimization of industrial processes to join high efficiency levels to low production costs (Girard, 2002, Demongodin, 2009).

Manufacturing and assembly are recognized as high value processes that significantly contribute to add value to the products (Rekiek, 2002, Riis et al., 2007). At the same time such industrial processes require a set of resources, e.g. machines, human resources, auxiliary devices and plants, etc., working together to ensure the target production levels to match the market demand. These resources represent a crucial asset for all companies and frequently require relevant long-term investments.

The design of effective approaches optimizing the aforementioned industrial processes is a hot topic for both researchers and practitioners of the industrial engineering sector (Darses, 2002). Several strategies are proposed by the literature in the past years considering different criteria, e.g. single vs. multi-objective planning, static vs. dynamic design, system analysis vs. simulation, together with a large set of tools and languages to model the operative systems (Bligh and Sodhi, 2007).

Petri nets, introduced by C.A. Petri in 1962 (Petri, 1962) to synchronize communicating automata, are, actually, a diffuse event-driven modelling language adopted to define a large set of models with increasing complexity and capabilities (Diaz, 2009). Particularly, major applications refer to the ICT sector and telecommunications, while minor attention is paid to define effective approaches to take advantage of the Petri net language and tools in the industrial sector.

This paper focuses on timed Petri nets and proposes a systematic approach for their adoption in the design of industrial processes, e.g. manufacturing and assembly. A realistic case study focused on the production process of high value automotive components is, further, presented to discuss the applicability and advantages of the proposes approach.

The reminder of this paper is organized as follows: the next Section 2 shortly revises the past and recent literature about Petri nets and their adoption in the industrial sector. In Section 3 fundamentals about Petri nets are introduced, while Section 4 outlines the steps of the proposed timed Petri net approach to support the design of industrial processes. In Section 5 the introduced approach is applied to the aforementioned realistic case study taken from the automotive sector. The model key outcomes, driving the process design, are fully described in Section 6 before drawing the paper conclusions together with suggestions for future developments in the last Section 7.

2. Literature review

Concerning Petri net fundamentals, evolutions and classifications a wide range of contributions are proposed (Girault and Valk, 2003, David and Alla 2005, Diaz, 2009), while several surveys face the application of Petri net language to the analysis of controllers for discrete event systems (Holloway et al., 1997, Cabeza and Mikolajczak, 1999, Zouari, 2003, Iordache and Antsaklis, 2009, Liu and Li, 2009).

Focusing on the industrial sector, literature contributions about the adoption of Petri nets for the design and management of production processes may be classified according to the industrial issue they address.

Manufacturing. D'Souza and Khator (1994), Moore and Gupta (1996) and Zurawski (2001) investigate the state of the art of Petri net models applied to automated manufacturing systems (AMSs). Among their findings the adoption of Petri net language to model and control flexible manufacturing systems (FMSs) and lines is of strong interest due to the needs of integration and coordination of several PLC-based workstations, working both in parallel or series, with the production schedule. A complete application of such an issue is proposed by Bohez (2004) discussing a generic timed Petri net model for the design of a dual kanban FMS.

Assembly. Assembly requires to integrate task/control planning, i.e. to define the activity sequence for each assembly station or resource, and assembly planning, i.e. to define the best sequence of the working stations and assembly processes (Thomas et al., 1996). Rosell (2004) discusses how to tackle these issues adopting high-level Petri nets presenting a survey of methods and tools for both assembling and disassembling processes. Finally, Li and Wang (2009) propose a modelling method of an assembly logistic process adopting timed coloured Petri nets.

Scheduling. Gu and Bahri (2002) and, recently, Gradišar and Mušič (2012) consider the scheduling problems for batch processes and propose timed Petri nets as an effective language to schedule and simulate production activities. They further present application examples from the chemical sector, i.e. production process of solvents and dyes, providing details about the Petri sub-net modelling for each entity, e.g. operations, resources, etc., their integration to compose the global net and, finally, the simulation and analysis to draw useful outcomes to build the production schedule.

At last, a further set of contributions focus on the description of tools and software environments to support the decision making process. Actually, no commercial tool cornered the market dictating an accepted standard. Berthomieu et al. (2004) and Hashemi and Blondin (2010) present two different tools for Petri nets and timed Petri nets, while a complete list of the available tools is reviewed by the Hamburg University (*www.informatik.uni-hamburg.de*).

3. Petri nets: fundamentals

Formally, a Petri net structure is a tuple:

$$N_s = \langle P, T, A, \mathscr{w} \rangle \tag{1}$$

where:

$P = \{p_1, \dots p_i, \dots, p_{|P|}\}$ is the set of places (the circles of Figure 1);

$T = \{t_1, \dots, t_j, \dots, t_{|T|}\}$ is the set of transitions, with $P \cap T = \emptyset$ (the black bars of Figure 1);

$A \subseteq (P \times T) \cup (T \times P)$ is the set of arches (p_i, t_j) or (t_j, p_i) defining a flow relationship between places and transitions and vice versa;

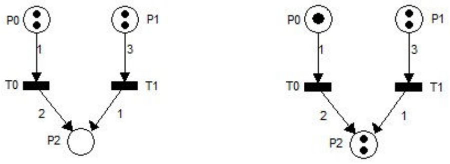
$\mathscr{w}: A \to \mathbb{N}^+$ is the so-called weight function associating an integer positive value, $w(p_i, t_j)$ or $w(t_j, p_i)$, to each arch.

Given a Petri net structure, N_s, the current state of a Petri net is univocally identified by a marking $\mathcal{X}$, defined through the following marking function:

$$x: P \to \mathbb{N}^+ \tag{2}$$

that associates an integer number of tokens to each place, $x(p_i)$, (the black dots of Figure 1). Consequently, a Petri net is defined as follows:

$$N = \langle N_s, \mathcal{X} \rangle = \langle P, T, A, \mathscr{w}, \mathcal{X} \rangle \tag{3}$$



(a) (b)

Figure 1. Petri net. Firing of transition t_0, t_1 is disabled.

The dynamic evolution of Petri nets is based on the concepts of enabled and fired transitions. A generic transition, t_j, is enabled if all places entering it have a number of tokens $x(p_i)$ not lower than the weight of the arches between the places and the transition, $w(p_i, t_j)$. Formally, t_j is enabled if and only if:

$$x(p_i) \geq w(p_i, t_j) \qquad \forall p_i: (p_i, t_j) \in A \tag{4}$$

Given a marking $\mathcal{X}$, all the enabled transitions are *fireable*. Firing a transition means generating a dynamic evolution of the net. Particularly, when a specific transition t_j is fired all the places entering it are decreased of a number of tokens equals to the weight of the correspondent arch, while all places exiting the transition are increased of a number of tokens equals to the weight of the correspondent arch. Formally, if t_j is fired:

$$x'(p_i) = x(p_i) - w(p_i, t_j) \quad \forall p_i: (p_i, t_j) \in A \tag{5}$$

$$x'(p_i) = x(p_i) + w(t_j, p_i) \quad \forall p_i: (t_j, p_i) \in A \tag{6}$$

After each firing a new marking $\mathcal{X}'$ of the Petri net is reached. In the beginning, the current net marking is called initial marking $\mathcal{X}_0$.

Further advanced structures of Petri nets, relevant for industrial applications, are inhibitor arcs, $\bar{A} \subseteq \overline{P \times T}$, and capacitated places. An inhibitor arc, $\overline{(p_\iota, t_j)}$, disenables the transition t_j if $x(p_i) \geq \overline{w(p_\iota, t_j)}$. Capacitated places allow an upper-limited number of tokens, $x^*(p_i)$, and disenable all transitions trying to add tokens exceeding such a capacity. Formally, t_j is disenabled if

$$x(p_i) + w(t_j, p_i) > x^*(p_i) \tag{7}$$

3.1 Timed Petri nets

Despite Petri nets are introduced as an event-driven language frequently, in industrial applications, the dynamic evolution of a system is defined by both event occurrences and the time flow, e.g. phase duration, temporal delays, etc.

Timed Petri nets introduce the time dependence to the aforedescribed language. In this context transition-timed Petri nets are considered, only. Formally, a clock structure $\mathcal{V}$ is introduced with:

$$v: T \to \mathbb{R}_0^+ \tag{8}$$

that associates a temporal delay, $v(t_j)$, to each transition t_j. Consequently, when the transition t_j is enabled it becomes *fireable* after $v(t_j)$ time units. The clock structure introduces a temporal delay to all transitions having $v(t_j) > 0$, i.e. timed transitions. If $v(t_j) = 0$, t_j is called immediate transition and is *fireable* immediately. Timed transitions are represented with rectangles instead of black bars.

Thus, a timed Petri net is formally defined as follows:

$$TN = \langle N, \mathcal{V} \rangle = \langle N_s, \mathcal{X}, \mathcal{V} \rangle = \langle P, T, A, \mathscr{w}, \mathcal{X}, \mathcal{V} \rangle \tag{9}$$

3.2 Timed Petri net language

To represent a generic industrial process through Petri nets or timed Petri nets a correspondence between the process entities and events and the net places and transitions needs to be fixed. Furthermore, interactions, conflicts and parallelisms are modelled through arcs and inhibitor arcs. Formally, a univocal correspondence function between each entity and event and a net features, i.e. places, transitions, arcs, etc., is required. Such a function generates a language $\mathcal{L}$ able to translate the industrial environment to the Petri net environment, i.e. semantic. The outcome of this process is the so-called labelled timed Petri net, defined as:

$$LTN = \langle TN, \mathcal{L} \rangle = \langle P, T, A, \mathscr{w}, \mathcal{X}, \mathcal{V}, \mathcal{L} \rangle \tag{10}$$

A labelled timed Petri net is a useful tool to support the design and management of industrial processes. The next Section 4 describes a general approach to build a labelled timed Petri net (labelled is omitted in the following) for industrial process analysis.

4. Proposed timed Petri net based approach

A generic industrial process involves several resources working together to transform, e.g. manufacture, assembly, inspect, etc., a product or a product mix. In order to model this process through timed Petri nets, building the aforementioned model language, the following systematic approach could be of help.

Process analysis and data collection. The goal is to provide a quantitative description of the process. Key data deal to: product working cycles and times, required resources, number and availability, buffers and capacity limits, work shift length and cycle times, productivity, etc. Such data are the input of the modelling process.

Modelling approach definition. Two different approaches are possible.

- Physical approach. The focus is on the system components classified as resources, i.e. working elements, or buffers, i.e. waiting elements. Products flow through the resources to complete their working cycles.
- Functional approach. The focus in on the work flows and process phases that define the basic structure of the net. Resources are shared among phases according to the working cycles.

Entity sub-net design. According to the chosen approach basic net structures are designed independently. As example, Figure 2 proposes a typical sub-net for a single process phase requiring a resource.

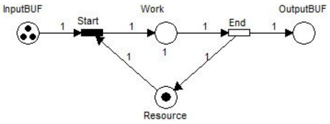


Figure 2. Sub-net for a single phase with resource.

InputBUF place contains raw products, represented by tokens. Immediate transition *Start* starts the working phase. It is enabled when at least one raw product is present and the resource is available, i.e. a token is present in the *Resource* place. The place *Work,* with capacity of one token, contains WIPs. Working time, ξ, is modelled through the *End* timed transition, with $v(t_{End}) = \xi$, so that this transition is enabled ξ time units after the work starts. When *End* is fired the *OutputBUF* increases of one token, i.e. finished product, and the resource is released.

Sub-net integration and synchronization. This phase deals to the timed Petri net full definition through the integration of previous entity sub-nets. Typical aspects to face refer to:

- working cycle modelling, faced creating ordered sequences of atomic phases with intermediate buffers
- shared resource allocation, faced introducing *conflict structures* among all phases requiring the same resource, i.e. resource allocation to a phase forces the others to wait its release
- work shift modelling, faced introducing a trigger defining working and rest periods and adding inhibitor arcs preventing products, i.e. tokens, flow out of the working time
- parallel activity synchronization, faced through independent nets feeded by a unique scheduling structure setting the cycle time.

Timed Petri net simulation. To run timed Petri net simulation the preliminary definition of the initial marking $\mathcal{X}_0$ is required. Common approaches assumes all resources available, no WIPs and fix the origin of simulation time at the beginning of the first work shift of the days. Other assumptions are possible and, generally, impact on the initial transient phase only, not on the regular behaviour of the net. Furthermore, simulation length is defined by the occurrence of at least one of these conditions:

- a defined simulation time limit is reached
- a defined set of fired transitions is reached
- no transition is, further, enabled, i.e. deadlock

The simulation report, including the ordered sequence of reached marking states, $\mathcal{X}$, and the correspondent simulation time represent the key simulation outcome driving process design actions.

Report analysis and actions. The simulation report allows to study the timed Petri net evolution, suggesting dynamic information profiles about working times, delays, required resources, buffer dimensions, critic phases, etc. Furthermore, the existence of a cyclic behaviour of the system can be highlighted, i.e. a sequence of marking states that occurs cyclically. These evidences are key elements driving the design or re-design actions on the considered industrial process.

5. Realistic case study

A job-shop production area to manufacture two high-value parts for luxury sport-cars is investigated through the proposed timed Petri net based approach. Both parts, called *PartA* and *PartB* in the following, share machines and human resources in their working cycles and are manufactured in parallel. Table 1 shows data about phases, timing, resources and required auxiliary devices.

Table 1. Case study, data collection.

PartA				PartB			
Phase	Resource	Time [min]	Aux	Phase	Resource	Time [min]	Aux
A1		157	Aux_I	B1		157	Aux_II
A2		157	Aux_I	B2		157	Aux_II
A3		157	Aux_I	*B3*	R_A	34	Aux_II
A4	R_A	30	Aux_I	*B4*	R_A + R_B	60	Aux_II
A5	R_A + R_B	60	Aux_I	*B5*	R_A	155	Aux_II
A6	R_A	145	Aux_I	*B6*	R_A	28	Aux_II
A7	R_A	32	Aux_I	B7	R_C	77	Aux_II
A8	R_C	24	Aux_I	B8	R_I	0	Aux_II
A9	R_D	0	Aux_I	B9	R_G	45	
A10	R_E + R_F	20					
A11	R_F	70					
A12	R_E + R_F	20					
A13		5					
A14	R_G	12					
A15	R_H	10					
A16	R_H	55					
A17		45					

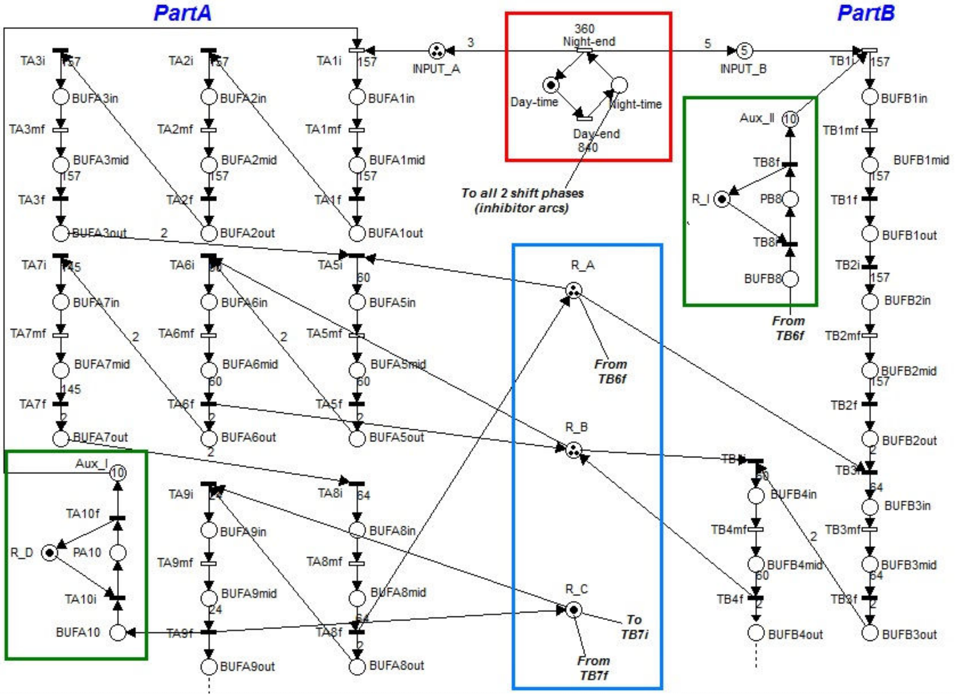


Figure 3. Timed Petri net for PartA and PartB production (extract).

All resources work for two shifts per day, i.e. 840min/day, except *R_A* with a supplementary night-time shift, i.e. 360min/day. Required system productivity is of 3 units of *PartA* and 5 units of *PartB* per day. Part cycle time is of 157min/part. Finally, each resource *R_A* works two parts of the same type together, i.e. *PartA* or *PartB* alternatively (correspondent phases are in italics in Table 1).

Figure 3 represents an extract of the developed timed Petri net. The functional modelling approach is adopted. *PartA* e *PartB* work flows are represented separately. Each phase is modelled through three places, i.e. input, WIP and output buffers, and three transitions. The red square includes the trigger managing two and three shift working time, while three of the shared resources are in the blue rectangle. Finally, auxiliary devices are modelled through dedicated tokens and sub-nets (green rectangles). The dynamic evolution of the net begins, cyclically, at the end of night-time, when new parts are ready to be manufactured (*Input_A* and *Input_B* places). These parts enter the first phase every 157 minutes, i.e. cycle time, through the timed transitions *TA1i* and *TB1i*. Required resources and auxiliary devices are captured at the beginning of the phase and released at its end. Tokens in places called *R_X* indicate the available resources. Finally the arcs connect the steps of each phase and synchronize all phases together building the process flow. The arc weight represents both the phase length, inside phase sub-net, and the number of part flowing the system together, between two consecutive phases. Weight for arcs with $w(p_i, t_j) = 1$ or $w(t_j, p_i) = 1$ is omitted. Finally, during night-time shift all phases except A6 and B7, i.e. phases requiring resource *R_A* only and no inspection that can be executed during the third shift, are stopped through inhibitor arcs departing from *Night-time* place.

6. Simulation and key-results

A simulation of 30 working days is done collecting data about the sequence of net markings. Simulative process duration is not a critic variable, i.e. few seconds adopting a common laptop.

Among the obtained outcomes, two of them are the most relevant:

- Production profile and system cyclic behaviour. The following Table 2 presents, for both parts, the production profile highlighting the presence of a cyclic behaviour of the system (dashed areas).

Table 2. Production profile and cyclic behaviour [min].

PartA			PartB		
Part n.	Exit time	Time gap	Part n.	Exit time	Time gap
1	1735	1735	1	1378	1378
2	1805	70	2	1455	77
3	2781	976	3	1759	304
4	2851	70	4	1836	77
5	3095	244	5	2488	652
6	3165	70	6	2565	77
7	4135	970	7	2802	237
			8	2879	77
			9	3116	237
			10	3193	77
			11	3778	585

  Both part profiles present an initial transient condition, i.e. production of part n.1, followed by a cyclic behaviour of 2400min, i.e. two working days. Consequently, every two working days the system behaviour repeats itself. Such outcome significantly eases the process design.

- Resources and auxiliary devices. Studying the trends of the token number in *R_X* and *Aux_X* places the resource utilization is calculated. Results show that

R_A is the most critic resource, used for 54.37% of its available time, justifying the additional night-time shift. Focusing on the auxiliary devices the following Figure 4 shows their utilization profiles.

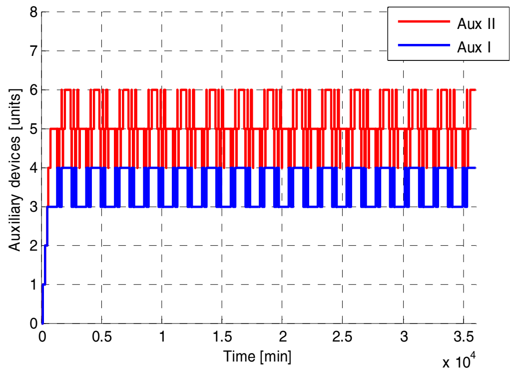


Figure 4. Auxiliary device utilization profiles.

Four *Aux_I* devices are necessary and three of them are always used, while six *Aux_II* devices are required and four of them are always used. In such a way no delays caused by the absence of these devices occur.

Finally no buffer queues are experienced at any time of the simulation. Minimum buffer dimension is of one part except for those related to phases requiring resource *R_A*. For them all, the required buffer minimum capacity is of two parts due to the contemporary work of two *PartA* or *PartB*.

7. Conclusions and further research

A general approach to support the design and analysis of industrial processes through timed Petri nets is discussed proposing useful steps for both the net development and analysis. Fundamentals about Petri net language are revised together with a short review of the literature. The adoption of this modelling language to face typical industrial problems, e.g. manufacturing, assembly, scheduling, etc., is, actually, limited despite the potential of Petri net language and its massive adoption in automation and event driven process control. A realistic case study, taken from the luxury sport-car automotive sector is introduced to highlight possible design guidelines coming from the timed Petri net simulation and outcome analysis.

Further research deals to refinements in the approach steps to ease its applicability, its adoption to other industrial contexts, the inclusion of stochastic time distributions for timed transitions and the adoption of Petri net static analysis next to simulation.

References

Berthomieu, B., Ribet, P.-O., Vernadat, F., (2004). The tool TINA - construction of abstract state spaces for Petri nets and timed Petri nets. *International journal of production research*, v. 42(14), p. 2741-2756.

Bligh, A., Sodhi, M., (2007). Designing the product development process. *Proceedings of the ASME international design engineering technical conferences and computers and information in engineering conference*, v. 3A, p. 613-624.

Bohez, E.L.J., (2004). A new generic timed Petri net model for design and performance analysis of a dual kanban FMS. *International journal of production research*, v. 42(4), p. 719-740.

Cabeza, A.M., Mikolajczak, B., (1999). Rapid prototyping methodologies and object-oriented Petri net models - a conceptual review. *Proceedings of the IEEE international conference on systems, man and cybernetics*, v. 3, p. 787-792.

Darsers, F., (2002). A framework for continuous design of production systems and its application in collective redesign of production line equipment. *Human factors and ergonomics in manufacturing*, v. 12(1), p. 55-74.

David, R., Alla, H., (2005). Discrete, continuous and hybrid Petri nets. Springer-Verlag, New York (USA).

Demongodin, I., Sauer, N., Truffet, L., (2009). Performance evaluation in manufacturing systems. In M. Diaz (ed.), *Petri nets, fundamental models, verification and applications*, p. 527-576.

Diaz, M., (2009). *Petri nets, fundamental models, verification and applications*. Wiley, Hobokey NJ (USA).

D'Souza, K.A., Khator, S.K., (1994). A survey of Petri net applications in modeling controls for automated manufacturing systems. *Computer in industry*, v. 24, p. 5-16.

Girard, P., Callot, M., Hugues, E., Kromm, H., (2002). Performance evaluation of the innovative product design process. *Proceedings of the IEEE international conference on systems, man and cybernetics*, v. 4, p. 66-71.

Girault, C., Valk, R., (1998). *Petri nets for systems engineering*. Springer-Verlag, New York (USA).

Gradišar, D., Mušič, G., (2012). Automated Petri-net modelling for batch production scheduling. In P. Pawlewski (ed.), *Petri nets - manufacturing and computer science*, p. 3-26. InTech, Rijeca.

Gu, T., Bahri, P.A., (2002). A survey of Petri net applications in batch processes. *Computers in industry*, v. 47, p. 99-111.

Hashemi, R.R., Blondin, J., (2010). SASSY: a Petri net based student-driven advising support system. *7th international conference on information technology: new generations*, p. 150-155.

Holloway, L.E., Krogh, B.H., Giua, A., (1997). A survey of Petri net methods for controlled discrete event systems. *Discrete event dynamic systems: theory and applications*, v. 7, p. 151-190.

Iordache, M.V., Antsaklis, P.J., (2009). Petri nets and programming: a survey. *Proceedings of the American control conference*, p. 4994-4999.

Li, Y., Wang, X., (2009). Petri net modeling for production logistics process. *Proceedings of the international conference on computational intelligence and software engineering*, p. 1-3.

Liu, J., Li., X., (2009). Application of colored Petri net in command and control system. *2009 international conference on intelligent human-machine systems and cybernetics*, v. 1, p. 323-326.

Moore, K.E., Gupta, S.M., (1996). Petri net models of flexible and automated manufacturing systems: a survey. *International journal of production research*, v. 34(11), p. 3001-3035.

Petri, C.A., (1962). *Kommunikation mit automaten*. Ph.D. dissertation, Institut für Instrumentelle Mathematik, Bonn (Germany).

Rekiek, B., Dolgui, A., Delchambre, A., Bractu, A., (2002). State of art of optimization methods for assembly line design. *Annual reviews in control*, v. 26, p. 163-174.

Riis, J.O., Johansen, J., Waehrens, B.V., (2007). Strategic roles of manufacturing. *Journal of manufacturing technology management*, v. 18(8), p. 933-948.

Rosell, J., (2004). Assembly and task planning using Petri nets: a survey. *Proceedings of the institution of mechanical engineers, part B: journal of engineering manufacture*, v. 128, p.987-994.

Thomas, J.P., Nissanke, N., Baker, K.D., (1996). A hierarchical Petri net framework for the representation and analysis of assembly. *IEEE transactions on robotics and automation*, v. 12(2), p. 268-279.

Zouari, B., (2003). High-level Petri net approach for supervisory control. *Proceedings of the IEEE international conference on systems, man and cybernetics*, v. 2, p. 1161-1166.

Zurawski, R., (2001). Petri net models and functional abstractions; applications to the design of automated manufacturing systems. *IEEE symposium on emerging technologies and factory automation*, v. 2, p. 71-83.