

SIMULATION

<http://sim.sagepub.com>

Simulation and Optimization of Logistic and Production Systems Using Discrete and Continuous Petri Nets

E. Jiménez Macías and M. Pérez de la Parte

SIMULATION 2004; 80; 143

DOI: 10.1177/0037549704045044

The online version of this article can be found at:
<http://sim.sagepub.com/cgi/content/abstract/80/3/143>

Published by:



<http://www.sagepublications.com>

On behalf of:



Society for Modeling and Simulation International (SCS)

Additional services and information for *SIMULATION* can be found at:

Email Alerts: <http://sim.sagepub.com/cgi/alerts>

Subscriptions: <http://sim.sagepub.com/subscriptions>

Reprints: <http://www.sagepub.com/journalsReprints.nav>

Permissions: <http://www.sagepub.co.uk/journalsPermissions.nav>

Citations <http://sim.sagepub.com/cgi/content/refs/80/3/143>

Simulation and Optimization of Logistic and Production Systems Using Discrete and Continuous Petri Nets

E. Jiménez Macías

Department of Electrical Engineering
System Engineering and Automation Group
University of La Rioja
C/Luis de Ulloa 20 E-26004 Logroño, Spain
emilio.jimenezm@die.unirioja.es

M. Pérez de la Parte

Department of System Engineering and Automation
University of Sevilla
Av. Descubrimientos E-41092 Sevilla, Spain

The use of Petri nets as a tool for optimizing logistic systems of industrial supply by simulation is analyzed in this article. The application of these tools presents two basic advantages: (1) their graphic/mathematical interpretation makes them particularly useful to be applied in simulation/analysis (optimization) applications, and (2) they can be considered from a discrete/continuous point of view, as a paradigm for modeling and analyzing discrete systems or as a continuous approximation to the real discrete model. These properties are very interesting in a global simulation analysis of logistic systems because their management is usually limited by their high complexity and large dimension. Petri nets are applied to the modeling and analysis of discrete logistic systems in the classical way, using the relatively recent continuization and fluidification techniques of discrete systems for their continuous approximation. This new methodology provides results with less computational effort.

Keywords: Logistic systems, manufacturing systems, Petri nets, continuous Petri nets, optimization

1. Introduction

Graph theory has traditionally played a very important role in the development and application of logistic systems. Petri nets (PNs) are a powerful extension of the state graphs that provide a more compact and effective representation of concurrent systems (with parallel evolutions and synchronization). They can be applied to logistic systems in several ways: their graphic (bipartite-directed graph) and mathematical (state equation) interpretations make them particularly interesting in coping with complex systems. They can be efficiently simulated because of the simplicity and power of PNs in graphical representations, and they can be optimized using the properties that the PNs' mathematical tool provides (Fig. 1).

PNs' basic characteristics in logistic systems can be summarized as follows:

- They provide a methodology and formalism to model discrete systems.
- Their graphical representation is clear, but they are able to cope with complex systems, especially those that include concurrence. That makes PNs an ideal tool for simulation.
- A large mathematical theory permits the analysis of the system behavior, including techniques for its optimization (reduction, block detection, etc.).
- They are extended throughout both the scientific and industrial communities, so there are several applications to manage PNs.
- The relatively recent contributions in the continuization of discrete systems for large populations provide approximated results by the simulation of the continuous model, so decisions can be made with less computational effort.
- They represent a paradigm for the modeling and analysis of dynamic systems, with several levels of abstraction and interpretation [1-3], so they can be applied to diverse systems with different levels of detail.

Furthermore, logistic systems usually present complex models with large populations. In addition, discrete

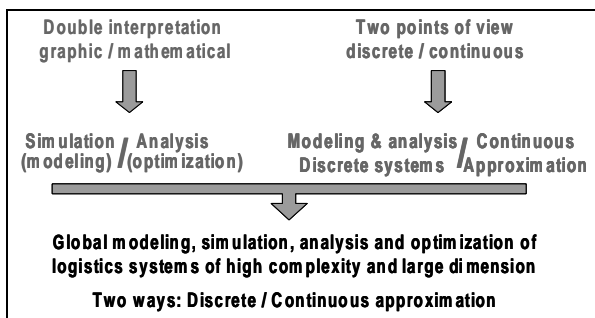


Figure 1. Analysis of Petri nets as a tool for optimization by simulation of logistic systems

systems present the state explosion problem (set of reachable states that is extremely large), which is inherent to the enumerative analysis due to large populations. Usually, though, large populations imply relatively small errors if the model is relaxed to a continuous approximation (in nonnegative real numbers). Then, the computational complexity of the model is reduced, and different mathematical tools can be used: linear programming techniques, differential equations, and so on [4]. So another important property of PNs that deals with these types of systems is their double point of view, as a discrete system or as a continuous approximation to the discrete real system (Fig. 1).

In this article, all the mentioned properties are illustrated through the optimization by simulation of a net of production plants, with each plant similar to one another. They share the logistic system either in the reception of raw materials or in the distribution of the final products and, occasionally, even in the intermediate products.

The different levels of abstraction and interpretation have facilitated integrating the models of the logistic and production systems. It is possible, therefore, to combine the simulation for global optimization and decision making in complex operations in only a PN model. This fact constitutes one of the particularities that has been developed in the simulation.

Some limitations that arise in the global analysis of logistic and production systems, such as complexity and large dimension, can be overcome with the useful technique of model continuization. However, the approximations to the continuous models, when they are possible, lose reliability when decreasing the number of components (the marking). So, a switched continuous/discrete simulation technique has been introduced to obtain more precise results. This is another particularity of the simulation in this work.

All this investigation and its applications have been developed as part of a research project that has been recently applied to a real logistic and production system. Thus, this piece of research is methodological in nature because it shows how to model and simulate logistic and produc-

tion systems, but it is illustrated with an application that shows that the provided topics are applicable to real-world problems.

Section 2 of this article introduces the use of PNs in the modeling, simulation, analysis, and optimization of discrete systems, and it presents an overview of formal analysis techniques. Section 3 shows the continuization of the real model to avoid the state explosion problem, which derives from large systems (as often happens with logistic systems). It also references the use of PNs for the simulation of these approximated systems. In section 4, the application of the optimization of logistic productive systems is shown by means of a real example. It consists of a system of manufacturing production and distribution. That example is used not only to show the modeling and simulation methodology of logistic systems using PN but also to explain, in a understandable way for the general reader, the innovations that have been included in the simulation: the analysis of the logistic and productive systems has been integrated (section 4.2), and it is simulated with continuous/discrete (and hybrid) switching models. Finally, section 5 summarizes the conclusions and the developed methodology.

2. Simulation, Analysis, and Optimization of Discrete Systems with PNs

A PN is a direct bipartite graph with two types of nodes: places and transitions. The marking defines the state of the system, and it evolves due to the firing of the transitions when their associated events occur. But a PN can also be represented as a tuple $N = \langle P, T, \mathbf{Pre}, \mathbf{Post} \rangle$, where the static structure of the system is defined by **Pre** and **Post**, or in pure (self-loop-free) nets, by the incidence matrix $\mathbf{C} = \mathbf{Pre} - \mathbf{Post}$. The state (or fundamental) equation can be represented as $\mathbf{m} = \mathbf{m}_0 + \mathbf{C} \cdot \sigma$, where $\sigma \in N^{|T|}$ and $\mathbf{m} \in N^{|P|}$. The reader can consult Silva [3, 5], Murata [6], and Silva and Teruel [7] to review the usual terms of PNs.

The PN includes a large mathematical theory that allows one to analyze the behavior of the system, as well as techniques for their optimization, including simplification by means of implicit places, the coalition of places, or the method of the places' source. The analysis of the PN habitually consists of validation, verification, and analysis of properties. Among the basic properties are liveness, cyclic property, limitation, conflict, mutual exclusion, weighted synchronic lead, conservative (p -invariant) and consistent (t -invariant) invariants, marking and firing invariants, absence of blockage, deadlocks, and traps [3].

The methods of PN validity analysis can be divided into dynamic (simulation) and static. Static methods include enumeration, reduction (or transformation), and structural analysis (for linear algebra or for deadlocks and traps) [8]. Much effort has been devoted to developing analysis techniques and properties of PN, which permits one to know the behavior of the system without using simulation. But it is also important to simulate with the model any

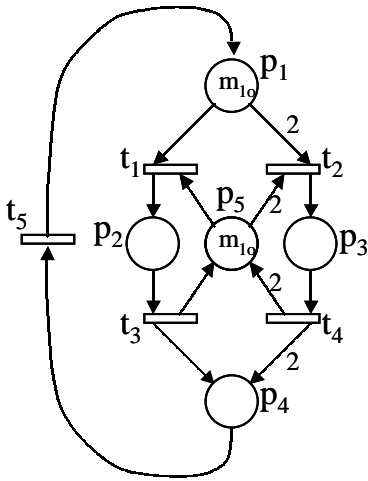


Figure 2. Petri net (PN) modeling a simplified logistic system

operation that will be developed in the real system to avoid unexpected behaviors.

Figure 2 shows the simplification of a logistic supply system for two processes that work in parallel (p_2 and p_3), where the p_1 and p_4 tokens represent the transport elements (wagons), and p_5 tokens are containers, which are necessary for the supply of both processes (let us suppose as many as the wagons). Delays are modeled by the input and output transitions of both processes, that is, t_1 , t_2 , t_3 , and t_4 . These transitions also model the synchronization between wagons and containers. Let m_{1o} be the initial marking in p_1 . For $m_{1o} \geq 2$, the net is pure, live, cyclic, structurally bounded, and k -bounded (with $k = m_{1o}$). It also has a structural conflict between t_1 and t_2 (which is an effective conflict for the autonomous net), p_3 is in mutual exclusion with all the other places iff $m_{1o} = 2$, and transitions t_1 and t_3 have a weighted synchronic lead of m_{1o} that is twice as much as the one between t_2 and t_4 . In addition, the net is conservative and consistent, with two marking invariants ($m_2 + 2 \cdot m_3 + m_5$ and $m_1 + m_2 + 2 \cdot m_3 + m_4$) and two firing invariants ($[t_1, t_3, t_5]$ and $[t_2, t_4, t_5, t_5]$); it does not present blockages; it has two deadlocks ($[p_1, p_2, p_3, p_4]$ and $[p_5]$), which are also traps (it possesses the deadlock-trap property) [3]; and p_5 is a redundant place (and therefore can be eliminated). Its incidence matrix is as follows:

$$C = \begin{bmatrix} -1 & -2 & 0 & 0 & 1 \\ 1 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 1 & 2 & -1 \\ -1 & -2 & 1 & 2 & 0 \end{bmatrix}$$

Let the model be deterministic, with time delays in the transitions of $\tau_1 = 3$, $\tau_2 = \tau_3 = \tau_5 = 1$, and $\tau_4 = 10$, where τ_i represents the time delay of the transition t_i . If

Table 1. Throughput of t_5 depending on the initial marking of p_1 (m_{1o})

m_{1o}	Throughput (t_5)
1	0.20
2	0.15
3	0.35
$4 + 2 \cdot n$ with $n = 0 \dots \infty$	0.20
$5 + 2 \cdot n$ with $n = 0 \dots \infty$	0.40

the number of elements of transport (m_{1o}) increases successively from 1, the throughput improves and worsens alternately, and starting from 5, the only important thing is to have an odd number, as can be seen in Table 1. So, if there are seven wagons, for example, an economic effort to increase one more wagon would provide a worse behavior. Or equivalently, if one is damaged (and is being repaired), it is better to separate another one.

This apparently strange behavior has a simple explanation when one bears in mind the two characteristics that the system exhibits simultaneously:

1. It has two parallel ways, p_2 and p_3 : one presents fewer temporal restrictions at the beginning (p_3) due to $\lambda_2 < \lambda_1$, and the other presents higher total restrictions ($\lambda_2 + \lambda_4 > \lambda_1 + \lambda_3$).
2. The source place for both ways (p_1) cannot provide enough tokens so that both of them can work nonstop (since the wagons are reused cyclically).

Bearing in mind the previous explanation, the results shown in Table 1 can be analyzed as follows:

- When there is a single token, it must go the globally faster way (i.e., through p_2) due to the weight of the input arc for the globally slower process, and it spends the sum of the delays $\tau_1 + \tau_3 + \tau_5$ in accomplishing a complete cycle. As is known, the throughput corresponds to the inverse of the cycle time.
- When there are two tokens, they will only go the slower way (i.e., through p_3) since $\tau_2 < \tau_1$, and they need $\tau_2 + \tau_4 + 2 \cdot \tau_5$ to complete a cycle.
- When there are an even number of tokens and more than two of them, the only active process is the slower one, but at least one token will be waiting in p_3 during the delay that corresponds to t_4 (in steady state). In fact, there will be at most two tokens out of p_3 . Therefore, the global time of the cycle is one of the slowest transitions (the bottleneck)—that is, τ_4 for two tokens, which is equivalent to $\tau_4/2$.
- When there are an odd number of tokens and more than three of them, their behavior in steady state will be equivalent to having a parallel evolution of a single token (through p_2 , as shown previously) and of an even number of tokens (through p_3). Consequently, the throughput corresponds to the sum of both processes' throughputs, that is, $1/(\tau_1 + \tau_3 + \tau_5) + 2/\tau_4$. It can be proved that the structural conflict of the net (between t_1 and t_2) does not become an effective conflict in steady state.

Table 2. Throughput of t_5 depending on the time constant of t_2 (τ_2)

τ_2	Throughput (t_5)
1–3	12/30
4–5	13.5/30
6	14/30
7–8	15/30
9–10	16/30

- When the number of tokens is exactly 3, its behavior in steady state will be equivalent to having a parallel evolution of only one token (through p_2) and of two tokens (through p_3). Therefore, the throughput corresponds to the sum of the throughputs of both processes, that is, $1/(\tau_1 + \tau_3 + \tau_5) + 2/(\tau_2 + \tau_4 + 2 \cdot \tau_5)$. In this case, the structural conflict becomes an effective conflict. Depending on how this conflict is solved, another throughput can be considered, but it is not important in this analysis.

From the previous analysis, it can be deduced that the maximum available throughput with the present temporal restrictions (considering the delays as minimum values) is the one that corresponds with having both processes (p_2 and p_3) working at their maximum capacities. The bottleneck of these processes will provide the throughput (i.e., $1/\tau_1 + 2/\tau_4 = 16/30$). In fact, the maximum throughput will be the minimum between that value and $1/\tau_5$.

Another strange behavior can occur with the time parameters. Let us suppose, for instance, $m_{1o} = 7$ in the previous example. In this case, if t_2 becomes slower, the system can improve the global throughput, as can be seen in Table 2, which shows the throughput in t_5 with the time delay in t_2 . The best behavior is obtained with a slower time parameter ($t_2 = 9$ or 10), and in those cases, the throughput (16/30) is the best possible one for any initial marking, taking into account the temporary parameters of the other transitions (as has been seen previously).

This other surprising behavior again occurs when one of the ways presents a worse behavior than the other one and monopolizes the marking flow. For this reason, even when the local behavior in certain areas (the input to the worst way) is improved, the global performance can be worse.

Two discrete simulations of the model corresponding to each example shown in Tables 1 and 2 are illustrated in Figures 3 and 4. In Figure 3, with $m_{1o} = 8$ and $t_2 = 1$, a throughput of 6/30 is obtained; in Figure 4, with fewer wagons and more delay ($m_{1o} = 7$ and $t_2 = 10$), the throughput is 16/30. The places p_1 , p_2 , and p_3 are only represented in the figures because the remaining ones are deducible from the previously presented conservative components (in this case, $p_4 = m_{1o} - m_1 - m_2 - 2 \cdot m_3$ and $p_5 = m_{1o} - m_2 - 2 \cdot m_3$).

But the maximum throughput can also be obtained by controlling the firing of the transitions appropriately (always respecting the minimum time requirements). In fact, the increase in delay of t_2 (τ_2) can be seen as a way of con-

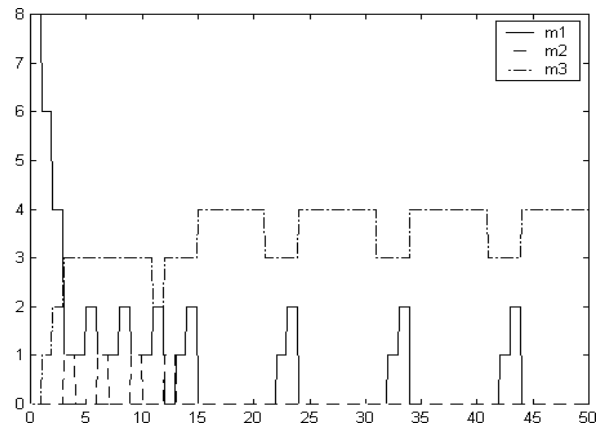


Figure 3. Discrete model simulation with $m_{1o} = 8$ and $\tau_2 = 1$ (throughput = 6/30)

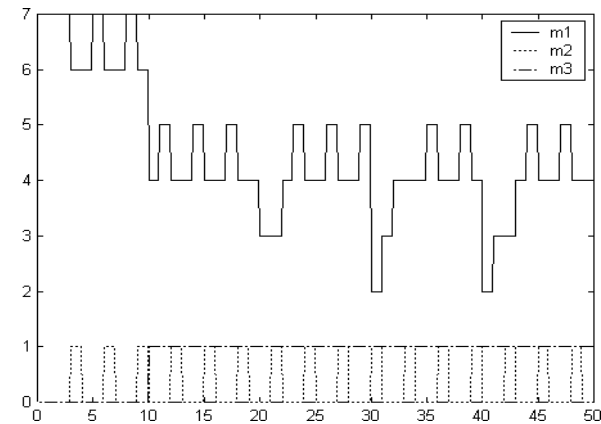


Figure 4. Discrete model simulation with $m_{1o} = 7$ and $\tau_2 = 10$ (throughput = 16/30)

trolling the process by delaying the firing of the transitions appropriately. Another simple control that also obtains the maximum throughput consists of including an inhibitor arc from p_3 to t_2 . Inhibitor arcs go from a place to a transition, and they are used to avoid the firing of the transition while the place is not empty; these arcs neither consume nor produce resources (tokens).

Then, with the mentioned inhibitor arc, for any $m_{1o} \geq 6$, the maximum throughput is obtained (16/30). That value derives from both processes working in parallel, with their respective bottlenecks the only restriction, and then the throughput is composed by two partial throughputs: 1/3 and 1/10. But this simple way to control the transitions works correctly only when $m_{1o} > 5$; in other cases, it is

Table 3. The throughput obtained with intelligent control in the discrete model

m_1	Throughput (t_5) Discrete with Intelligent Control
1	1/5
2	1/3
3	1/5 + 2/13
4	1/3 + 2/13
5	1/3 + 2/12
≥ 6	1/3 + 2/10

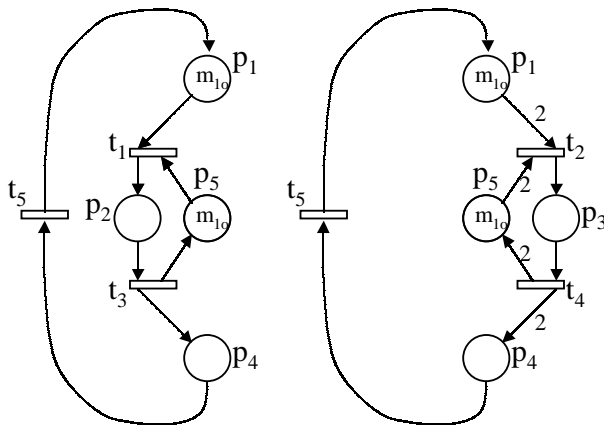


Figure 5. Cycles that must follow one and two tokens, respectively, for maximum throughput

necessary to use another different control in the transitions to obtain the maximum throughput. For $m_{1o} = 1$ and $m_{1o} = 3$, it is not necessary to control the transitions to have the optimum throughput. When $m_{1o} = 2$, the only necessary control is to avoid firing the transition t_2 ; that is, the only allowed process is p_2 (as if the system were only the cycle: $p_1 - t_1 - p_2 - t_3 - p_4 - t_5$) (left-hand side of Fig. 5). If $m_{1o} = 4$, two tokens must follow the process p_2 (partial throughput = 1/3) and the other two tokens the process p_3 (partial throughput = 2/13) (right-hand side of Fig. 5). Thus, using the appropriate control on the transitions (intelligent control), the maximum throughputs can be obtained, which are shown in Table 3.

In the proposed example, some of the found behaviors can be straightforwardly predicted. Besides, it is an example that is difficult to reproduce with those characteristics in a production or logistic system. However, it must be taken into account that finding structures with those characteristics as a result of the connections between different parts of a system is simpler. These structures can be more complex, and what has been treated as a place can be a subnet as complex as can be imagined. Hence, although a system such as the one in the example is not designed explicitly, similar structures can appear in a complete system.

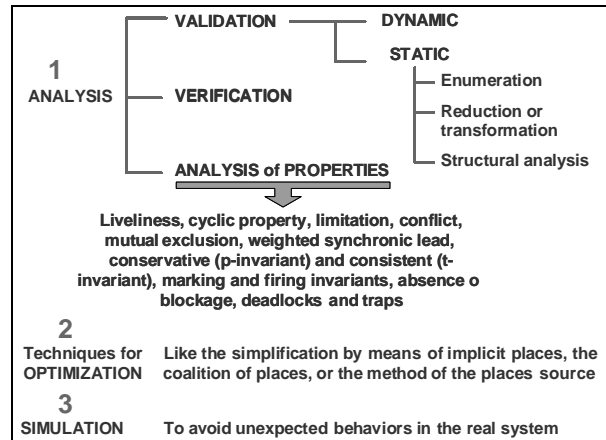


Figure 6. Analysis of Petri nets as a tool for optimization by simulation of logistic systems

Thus, it can be seen that increasing the number of elements of the system or improving the time parameters in certain points may not improve the global behavior. The simulation, combined with formal analysis techniques, can be very important to ensure that any future operation in the system will be correct. All this can be summarized with the sketch in Figure 6.

3. Analysis by Means of Continuous PN Simulation

The discrete models must be continuized due to the well-known *state explosion problem*, which means that the group of reachable states of a discrete PN system can become extremely large when the number of elements increases. The continuization simplifies the model and permits the use of different mathematical tools (linear programming techniques, differential equations, etc.).

In continuous PNs, any nonnegative real marking is allowed (in a discrete PN, the marking is restricted to integers). The firing of a transition is similar: t is enabled at \mathbf{m} iff $\forall p \in t, \mathbf{m}[p] > 0$. The *enabling degree* is defined as $enab(t, \mathbf{m}) = \min_{p \in t} \{\mathbf{m}[p] / \mathbf{Pre}[p, t]\}$, and the firing of t in a certain quantity $\alpha \leq enab(t, \mathbf{m})$ leads to a new marking $\mathbf{m}' = \mathbf{m} + \alpha \cdot \mathbf{C}[p, t]$. Independently of the temporary interpretation of the PN (deterministic, Markovian stochastic, etc.), the interpretation of the transition firing will be deterministic in the approximated continuous model [9]. Not all discrete nets can be appropriately continuized [10], and the properties can differ surprisingly between the discrete model and its continuous approximation [11]. Then, care must be taken when continuizing a model.

The state equation $\mathbf{m} = \mathbf{m}_0 + \mathbf{C} \cdot \sigma$ summarizes the marking evolution, and as it varies in a continuous way, it

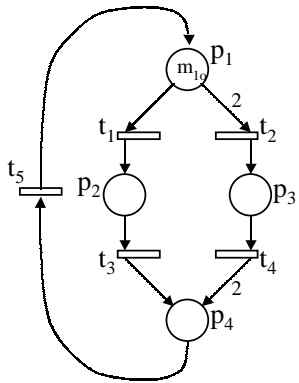


Figure 7. Model used with finite server semantics (FSS) approximation

can be derived with respect to time: $\dot{m} = C \cdot \dot{\sigma}$. The value $\dot{\sigma}$ represents the flow through the transitions, and it is usually denoted by \mathbf{f} [4]. On the whole, it depends locally on the marking and, therefore, on time. If $\mathbf{f}(\tau)$ is defined with the usual semantics in discrete nets, the temporary evolution of the continuous PN can be obtained by an interpretive extension [10, 12]:

- *Infinite server semantics (ISS)*. The enabling degree of t_i is $\mathbf{e}(\tau)[t_i] = \min_{p \cdot t_i} \{ \mathbf{m}[p] / \mathbf{Pre}[p, t_i] \}$ and represents the number of active servers in the transition at instant τ . The rate associated with t_i is $\lambda[t_i]$, and transitions fire with $\mathbf{f}(\tau)[t_i] = \lambda[t_i] \cdot \mathbf{e}(\tau)[t_i]$.
- *Finite server semantics (FSS)*. The firing speed $\mathbf{f}(\tau)[t_i]$ is upper bounded by $K[t_i]$ times the speed of one server $\mathbf{F}[t_i]$, and then the transitions are fired with $\mathbf{f}(\tau)[t_i] \leq K[t_i] \cdot \mathbf{F}[t_i]$ [13].

The simulation of the continuized PN, as an approximation to the real discrete model, is used to determine its behavior parameters (e.g., the throughput). However, one must be careful when interpreting the results of the continuized model. For example, in the previous PN of Figure 2, the implicit place can be eliminated (because, by definition, it never restricts its output transitions [14]), and then the structure is converted into two parallel lines (Fig. 7).

If an FSS continuization is used to evaluate the throughput of t_5 , the model will provide the maximum value (16/30) for any m_{1o} , and it has been seen previously that it is not possible for $m_{1o} < 6$ in the discrete case, not even when applying intelligent control in firing the transitions. An ISS model can also be used. In this case, it is considered only the “bottleneck” of each parallel process (i.e., the slowest transition), and a self-loop with a token is included in each transition to model the upper bound in firing the transitions (Fig. 8). The results are listed in Table 4. This table shows

Table 4. Comparison between finite server semantics (FSS), infinite server semantics (ISS), and the optimal discrete model

m_{1o}	Throughput (t_5) with FSS	Throughput (t_5) with ISS	Throughput (t_5) with Discrete (IC)
1	0.53	0.30	0.20
2	0.53	0.48	0.33
3	0.53	0.53	0.35
4	0.53	0.53	0.48
5	0.53	0.53	0.50
≥ 6	0.53	0.53	0.53

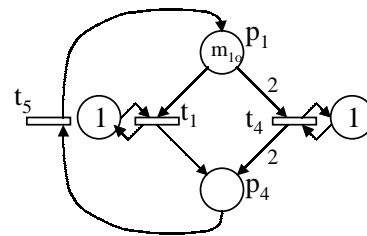


Figure 8. Model used with infinite server semantics (ISS) approximation

that valid conclusions can be obtained with the continuous approximation, with large populations, and with less computational effort; in addition, the table also shows that without large markings, the continuous approximation can give wrong conclusions.

It must be noted that the example is used with a deterministic interpretation, with time restrictions corresponding to delays in the firing of the transitions. It can also be used as a stochastic model (nondeterministic), with the time parameters indicating the probability (usually with exponential distribution) that the firing occurs. In these cases, the continuization again produces a deterministic model, and its simulation provides an approximation to the behavior characteristics of the system. Recalde, Teruel, and Silva [11] provide a complete example of manufacturing production that compares all the models in which the ISS and the discrete model completely agree.

Characteristically, the ISS provides more satisfactory results, even without large populations, and that is why we normally use it. In these continuous models (ISS), the simulation is very useful because the analytic solutions are difficult to obtain, basically, for two reasons: the speed depends on the marking (and therefore on the state) locally, and the synchronizations introduce the “minimum” function, which is really useless in analytic developments. Consequently, systems of switched differential equations are obtained, which are easy to simulate but difficult to solve. As an example, a simple ordinary sequential cycle

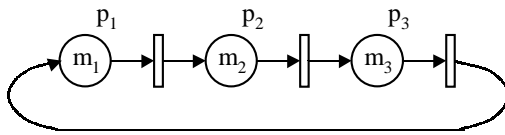


Figure 9. Simple ordinary sequential cycle

(of unitary weights) with three places, which does not even include synchronization (Fig. 9), provides the following complicated solution (deducible from the previous equations that are defined by the ISS model):

$$\begin{aligned}
 m_1(t) = & (1/3 + \cos(3^{1/2} \cdot t/2) \cdot e^{-3t/2} \cdot 2/3) \\
 & \cdot m_{01} + (1/3 - \sin(3^{1/2} \cdot t/2) \cdot e^{-3t/2} \cdot 3^{-1/2} \\
 & - \cos(3^{1/2} \cdot t/2) \cdot e^{-3t/2} / 3) \cdot m_{02} \\
 & + (1/3 + \sin(3^{1/2} \cdot t/2) \cdot e^{-3t/2} \cdot 3^{-1/2} \\
 & - e^{-3t/2} \cdot \cos(3^{1/2} \cdot t/2) / 3) \cdot m_{03}.
 \end{aligned}$$

The expressions of the other two state variables ($m_2(t)$ and $m_3(t)$) can be deduced by symmetry.

It is very important to emphasize that the simulations corresponding to this section are developed to determine the behavior of the system in an approximated way. From either a deterministic or stochastic discrete model, an approximated continuous model is deduced that is deterministic and that has a simulation that approximately provides the results for the behavior of the discrete model (to delve deeper into the continuization of stochastic PN, readers can review Jiménez, Recalde, and Silva [4]; Silva and Recalde [10]; and Recalde and Silva [12]). These simulations are different from the ones developed in the previous section, which correspond to a maneuver simulation, in which the deterministic model is simulated to check how the real system would work, especially in complex operations.

Even in this continuous simulation, the use of PNs as a basis for modeling and simulation is advantageous compared to the use of equations only [11].

4. Application to the Optimization of a Logistic/Manufacturing System

In this section, we show the application of PNs and continuous PNs to the optimization of a system composed of several industrial plants. These plants are similar to one another and share the logistic system of supply and distribution.

4.1 The Manufacturing Production and Distribution System

The production system is composed of several different synchronized processes that work in parallel. It is usu-

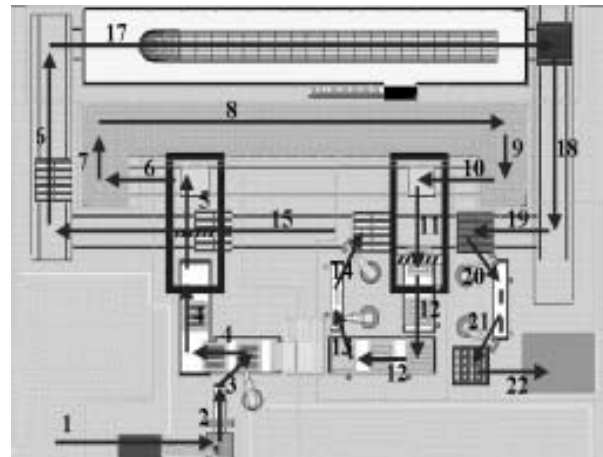


Figure 10. Concurrent production system

ally necessary to use PNs to model it due to the degree of high concurrence. On the other hand, the marking is very large because it considers many elements: shared resources (robots, ferry platforms, conveyor belts), transport and storing elements (wagons, trays, and boxes), and overall production elements (bricks). Therefore, it is appropriate to complement discrete analysis with the continuization techniques.

The system behavior (Fig. 10) can be summarized as follows [15]: shaped parts (1) move along a conveyor belt (2), and a robot (3) places them on the trays. When a tray is complete, another conveyor belt (4) transports it to a gantry crane that places it (5) in a trays wagon. When this trays wagon is full, it moves (6) on a rail to the dryer's entrance and then it follows a route (7-9) through a dryer for a specified time. As soon as the trays wagon has finished, it goes (10) to a discharge area, where another gantry crane discharges (11) the trays from the wagon. These trays advance (12) for another transfer. A robot (13) piles the parts from the trays. Then, each empty tray follows its cyclical route. The parts that have been piled are heaped again by another robot (14) in a new wagon for parts. When the wagon is full, it advances (15) to a ferry platform that transports it (16) perpendicularly to its advanced route. Once transported, it follows a route (17) through the oven for a time, and when it has finished, another ferry platform transports it again (18). Then the parts wagons advance (19) to an unloading area, where a robot (20) piles the individual parts and another robot (21) packs them into boxes (22).

The system can be modeled as shown in Figure 6, with colored PNs [16, 17]. Models with similar behaviors can be simplified using colored PNs, but they do not increase the descriptive power of ordinary PNs. Besides, it is a deterministically synchronized sequential process (DSSP) [18] because it can be divided into blocks that behave as state

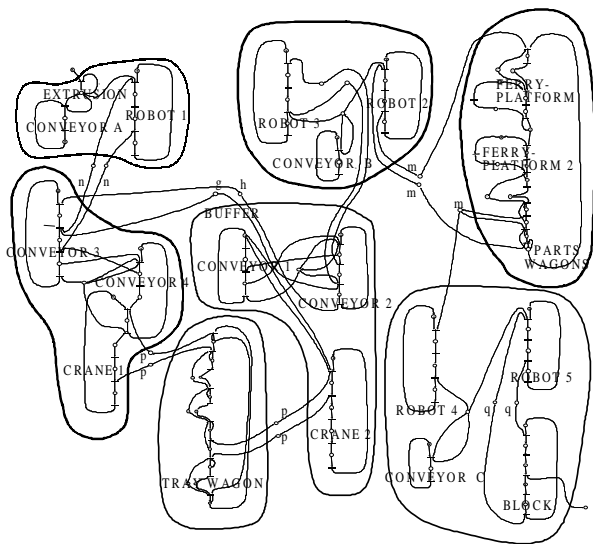


Figure 11. Petri net (PN) model for the system in Figure 10. Deterministically synchronized sequential process (DSSP) elements are grouped.

machines interconnected by buffers and that satisfy some restrictions. In Figure 11, the blocks are marked with closed curves, and the buffers can be seen among them.

This model can be considered deterministic, when carrying out the simulation of complex maneuvers, or stochastic, when determining behavior properties—and then continuization is necessary. Figure 12 shows the complexity of discrete simulation for just the first few seconds of working in the plant [19].

Moreover, a DSSP can be used to generate the state space and to find the steady-state probabilities of the stochastic extension of the net in a efficient way (without ever explicitly computing and storing the reachability graph of the system) [18].

4.2 Integration of the Logistic and Manufacturing System: Switching Continuous/Hybrid Simulation

In Figure 11, the system is divided into seven blocks, interconnected by buffers, with an input place and an output place. The last block (on the right-hand side) could also have been separated into two (or even three) new blocks. That complex system corresponds to only one production plant, which must be integrated with the rest of the plants and with the logistic system.

The logistic system of the production plants consists of three kinds of processes: (1) the raw material supply (to the input places), (2) the distribution of resultant products (from the output places), and (3) the redistribution of the intermediate products into the buffers, from one plant to the other, to compensate for failures and to reduce the buffer

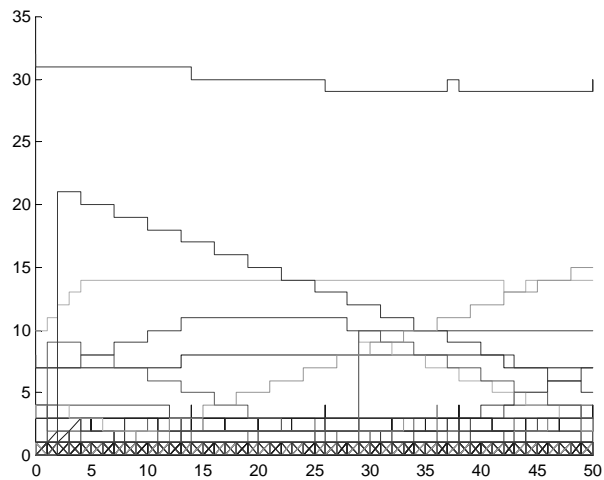


Figure 12. Discrete simulation in the previous concurrent production system

sizes. An example of simulation of the logistic/production system with three plants is illustrated in Figure 13. There are two logistic centers (lc_1 , lc_2) of raw material supply and only one center of resultant product storing (rp_1)—the real system has doubles of all of them. The model also includes the raw material supply to the input places (p_i) (1 in the simplified model in Fig. 13), the distribution of resultant products from the output places (p_o) (2), and the redistribution of the intermediate products in the buffers in the production processes (3). This redistribution is schematized with simple arrows between each pair of factories, but the logistic systems include the redistribution of the six types of buffers in the plants.

With this scheme, the joint simulation of the complete system can be carried out in the two ways described previously: test of maneuvers and determination of the behavior characteristics. It is important to analyze the logistics and the production globally because improving a particular element cannot be beneficial for the complete system (as shown in section 2).

However, the simulation of a (global) system that is so extensive has the difficulties of heterogeneity and large dimension. Heterogeneity obliges one to use techniques of continuization (seen in section 3), but it is necessary to use these techniques carefully to satisfy the conditions required to consider “large populations” in the system. Some parts of the system could not satisfy them momentarily. Furthermore some other parts could never satisfy these conditions, and then the system must be considered hybrid (e.g., shared resources, as robots). But sometimes the case is one of “short temporary populations” (e.g., the simulation of a buffer that is emptied as a result of a failure of a previous production system). In these cases, if the system is simulated as discrete, it will undergo state explosion when the

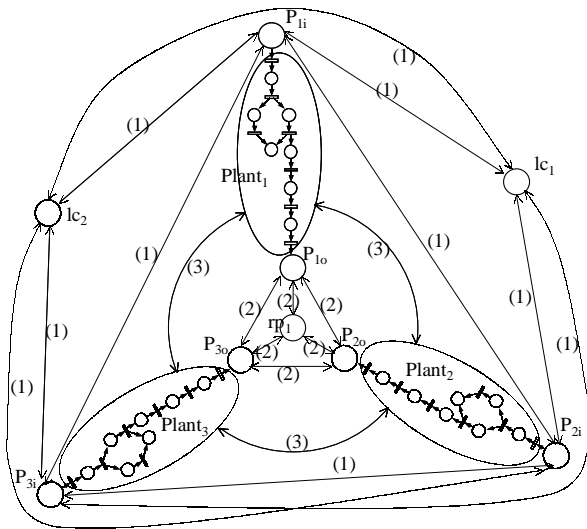


Figure 13. Simplified global logistic/production system. (1) Raw material supply (to the input places, p_i); (2) distribution of resultant products (from the output places, p_o); and (3) redistribution of the intermediate products in the buffers.

population grows again. That is why the adopted solution is to implement a switch in the system that converts from discrete to continuous (and vice versa) as a function of its value [18].

Thus, with the techniques referred to in this article (the application of the methodology), improvement has been achieved in the global behavior of the logistics and production processes using simulation as the basic tool.

5. Conclusions

In this article, a methodological analysis of applying PNs for the modeling and simulation of logistic and production systems, which provide their optimization, has been presented. Complexity and large dimension are important characteristics of the mentioned systems, which is why PNs constitute a very useful tool to deal well with them. On one hand, PNs' versatility makes them appropriate to incorporate models on different scales, such as the logistic and production models. This fact facilitates global management of the system. On the other hand, the continuization techniques of discrete models permit one to simplify complex models and to use mathematical tools (to solve the state explosion problem). The continuous model is an approximation of the discrete model (used to estimate the properties of the real discrete system). The application of the continuization techniques and the interpretation of its results must be made carefully. Taking that into account while avoiding the problem of higher relative error when the marking is small, a switched simulation can be intro-

duced according to the system internal state. The switch is accomplished between continuous parts (when large marking occurs) and discrete parts (when marking decreases).

Another conclusion of this work is realizing the importance of the mixed use of simulation and analysis of properties for system optimization. If only simulation is undertaken, erroneous or limited conclusions can be obtained because it cannot take some characteristics, such as instabilities, into account. PNs provide a very important framework for the analysis of properties, which permits one to know how the system will behave without the computational effort that the simulation process implies. However, the mixed use of simulation and analysis of properties provides very good results and allows one to look for strange and unexpected behaviors that could affect the system.

It is important to differentiate the two types of simulations presented in this work. One is the simulation of the real discrete system, which permits one to know the system properties and its expected operation and to be able to anticipate the failures. The other type is the simulation of the continuized model, which is aimed at estimating approximately but easily the characteristics of operation, but it does not indicate at all how the system will work.

6. References

- [1] Jensen, K., and G. Rozenberg, eds. 1991. *High-level Petri nets*. New York: Springer.
- [2] Thiagarajan, P. S. 1987. *Elementary net systems: Part I*. Lecture Notes in Computer Science, vol. 254. New York: Springer.
- [3] Silva, M. 1985. *Las Redes de Petri: en la automática y la informática*. Madrid: AC Editores.
- [4] Jiménez, E., L. Recalde, and M. Silva. 2001. Forrester diagrams and continuous Petri nets: A comparative view. In *Proceedings of the 8th IEEE International Conference on Emerging Technologies and Factory Automation*, pp. 85-94.
- [5] Silva, M. 1993. *Practice of Petri nets in manufacturing*. London: Chapman & Hall.
- [6] Murata, T. 1989. Petri nets: Properties, analysis and applications. *Proceedings of the IEEE* 77 (4): 541-80.
- [7] Silva, M., and E. Teruel. 1996. A systems theory perspective of discrete event dynamic systems: The Petri net paradigm. In *Symposium on discrete events and manufacturing systems (CESA 96 IMACS Multiconference)*, pp. 1-12.
- [8] Silva, M., and E. Teruel. 1997. Petri nets for the design and operation of manufacturing systems. *European Journal of Control* 3 (3): 182-99.
- [9] David, R., and H. Alla. 1987. Continuous Petri nets. In *8th European Workshop on Application and Theory of Petri Nets*, pp. 275-94.
- [10] Silva, M., and L. Recalde. 2000. Réseaux de Petri et relaxations de l'intrégrité: Une vision des réseaux continus. In *Conf. Internationale Franco-phonie d'Automatique (CIFA 2000)*, pp. 37-48.
- [11] Recalde, L., E. Teruel, and M. Silva. 1999. Autonomous continuous P/T systems. In *Application and theory of Petri nets*, vol. 1639 of Lecture Notes in Computer Science, edited by J. Kleijn S. Donatelli, 107-26. New York: Springer.
- [12] Recalde, L., and M. Silva. 2000. PN fluidification revisited: Semantics and steady state. In *4th International Conference Automation of Mixed Processes: Hybrid Dynamical Systems (ADMP 2000)*, pp. 279-86.
- [13] Alla, H., and R. David. 1998. Continuous and hybrid Petri nets. *Journal of Circuits, Systems, and Computers* 8 (1): 159-88.

- [14] Colom, J. M., and M. Silva. 1990-1991. *Improving the linearly based characterization of P/T nets*. Lecture Notes in Computer Science, vol. 483. New York: Springer.
- [15] Jiménez Macías, E., and E. Jiménez Tofé. 2001. Factory modelling and control with Petri nets in processes with cyclic subsystems. In *Proceedings of the International Conference on Intelligent Systems and Control*, pp. 140-45.
- [16] Dutheillet, C., G. Franceschinis, and S. Haddad. 1998. Analysis techniques for colored well-formed systems. In *Performance models for discrete event systems with synchronizations: Formalisms and analysis techniques*, edited by G. Balbo and M. Silva, 233-84. Zaragoza: KRONOS.
- [17] Jensen, K. 1994. *Coloured Petri nets: Basic concepts, analysis methods, and practical use*. EATCS Monographs on Theoretical Computer Science. New York: Springer.
- [18] Campos, J., S. Donatelli, and M. Silva. 1997. Structured solution of stochastic SDDP systems. In *Proceedings of the 6th International Workshop on Petri Nets and Performance Models*, p. 91.
- [19] Jiménez, E. 2001. *Técnicas de Automatización Avanzada en Procesos industriales*. Ph.D. diss., Ed. Univ.de La Rioja.

Emilio Jiménez Macías studied industrial engineering (with a computer science, electronics, and automation specialty) at the

University of Zaragoza. After working several years in the industrial private sector (head position in the Researching and Developing Department), he became an assistant professor at the University of La Rioja, where he presently works in the Electrical Engineering Department (System Engineering and Automation Group). In 2001, he presented his Ph.D. dissertation about industrial automation. His research areas include factory automation, modeling and simulation, and industrial processes. His main works are developed with the Automation Group of the University of Zaragoza.

Mercedes Pérez de la Parte received her B.S. degree in telecommunications engineering from the University of Seville in 1997. She has held research and teaching positions at the University of Los Andes, Venezuela, and currently has a research position at the University of Seville. She is presently preparing her Ph.D. dissertation. Since 2000, she has been an assistant professor in the Departamento de Ingeniería de Sistemas y Automática of the University of Seville, Spain.