

Research Article

Control of Discrete Event Systems by Means of Discrete Optimization and Disjunctive Colored PNs: Application to Manufacturing Facilities

Juan-Ignacio Latorre-Biel,¹ Emilio Jiménez-Macías,² Mercedes Pérez de la Parte,³ Julio Blanco-Fernández,³ and Eduardo Martínez-Cámara³

¹ Department of Mechanical, Energy and Materials Engineering, Public University of Navarre, Campus of Tudela, 31500 Tudela, Spain

² Department of Electrical Engineering, University of La Rioja, 26006 Logroño, Spain

³ Department of Mechanical Engineering, University of La Rioja, 26006 Logroño, Spain

Correspondence should be addressed to Emilio Jiménez-Macías; emilio.jimenez@unirioja.es

Received 21 February 2014; Accepted 30 April 2014; Published 11 June 2014

Academic Editor: Guanglu Zhou

Copyright © 2014 Juan-Ignacio Latorre-Biel et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Artificial intelligence methodologies, as the core of discrete control and decision support systems, have been extensively applied in the industrial production sector. The resulting tools produce excellent results in certain cases; however, the NP-hard nature of many discrete control or decision making problems in the manufacturing area may require unaffordable computational resources, constrained by the limited available time required to obtain a solution. With the purpose of improving the efficiency of a control methodology for discrete systems, based on a simulation-based optimization and the Petri net (PN) model of the real discrete event dynamic system (DEDS), this paper presents a strategy, where a transformation applied to the model allows removing the redundant information to obtain a smaller model containing the same useful information. As a result, faster discrete optimizations can be implemented. This methodology is based on the use of a formalism belonging to the paradigm of the PN for describing DEDS, the disjunctive colored PN. Furthermore, the metaheuristic of genetic algorithms is applied to the search of the best solutions in the solution space. As an illustration of the methodology proposal, its performance is compared with the classic approach on a case study, obtaining faster the optimal solution.

1. Introduction

Many products and services are produced and offered to the customers in a global market. This globalization may imply the participation, in the lifecycle of a manufacturing facility, of many and very diverse agents, resources, and information from very different and distant regions of the world. As a consequence, the management and operation of a manufacturing system, seen as discrete event dynamic systems, require the consideration of a large number of elements, which participate in a greater or less extent in the production yield.

Another consequence of this fact consists of the existence of agents and subsystems, which configure the manufacturing facility and present independent and parallel stages in their

evolution, as well as multiple mutual relationships, such as sharing common resources or competing for them. Hence, the behavior of the complete production system may be very complex, despite the fact that the structure and evolution of the subsystems, analyzed independently, might be simple [1, 2].

Some of the components that may configure a subsystem in a manufacturing process are flexible manufacturing systems (FMS), machining centers, conveyor belts or roller tables, automatic guided vehicles (AGV), robots, buffers, or combinations of the previous ones [3, 4].

On the other hand, a manufacturing facility presents a number of controllable parameters, which can also be called freedom degrees [5]. These freedom degrees allow the decision maker to manage the manufacturing facility or

facilities with the purpose of achieving certain objectives, such as a high yield, rate of utilization of the resources, like manpower and machinery, degree of satisfaction of customers, and production timespan or delivery time of the manufactured products.

The decision maker may choose a combination of values for the controllable parameters, which lead to a single configuration of the manufacturing system. The objective of the decision maker consists in finding the best configuration for the freedom degrees of the system to achieve the objectives. The way in which the configurations are built up, by means of combinations of values for the different freedom degrees, implies a combinatorial construction of the solution space [6, 7]. This fact may lead to a combinatorial explosion, which can be associated with NP-hard problems, such as the exploration of this kind of solution spaces [8].

The development of strategies for the solution of such difficult problems has been the goal of many researchers during the last decades [9–11]. However, despite the research effort focused on this problem so far, there is not any methodology able to cope with every type of problem and solve it in a reasonable amount of time.

A classic approach, intensive in handwork, consists of building a formal description of the manufacturing system in the form of a model, hiring experienced and expensive experts for choosing a reduced set of promising configurations, performing a simulation of the evolution of the real system for every one of the chosen configurations, and interpreting the results for selecting the best simulation and, hence, the best managing decision consisting of the associated configuration.

This approach presents many drawbacks, such as the cost, the time required to complete the process, the lack of objectiveness, the reduced amount of feasible configurations analyzed, and the low repeatability when other decision makers are hired.

To overcome these inconveniences, several strategies may be considered. Among the most popular approaches, the automation of part of the decision making process [5] and the reduction in the size of the model of the real system can be considered [12]. This automation process leads naturally to the statement of an optimization problem, where the objectives to be achieved by the manufacturing facility are quantified by means of the objective or multiobjective functions

One of the stages of such an optimization problem, the search for the best configuration of the manufacturing system in the solution space, is the focus of an important research effort, since an efficient search may reduce considerably the computational resources required by the whole optimization process [10].

Among the most successful search techniques applied in the manufacturing arena are the ones based on artificial intelligence and more particularly the metaheuristics. This methodology, after the preparation stage of adjustment of the metaheuristic parameters, may lead to adequate and realistic solutions in no matter which optimization problem it is applied. Some metaheuristics, which have been applied

successfully, are tabu search, genetic algorithms, ant colony, particle swarm, or simulated annealing [13].

In general, these techniques provide acceptable solutions in a reduced period of time by means of a probabilistic approach. The metaheuristic provides the search algorithm with a mechanism for converging to an optimum, while the probabilistic behavior leads the search out from local optima, since better solutions may be found in a global optimum.

Once a strategy for the decision making has been chosen, the statement of an optimization problem in conjunction with a metaheuristic-based search in the solution space, it is convenient to describe the optimization problem itself. Despite the fact that later in this same document a more detailed description will be shown, it is possible to indicate that an optimization problem must quantify the configurations of the system. That is, a way to measure how close to the objectives is a certain configuration should be defined, depending on variables or freedom degrees of the system (manufacturing facility or supply chain). In other words, the quality of every solution in the pool of feasible configurations, or solution space, should be able to be measured.

One common way to measure the quality of a solution of the mentioned optimization problem is an objective or multiobjective function. The calculation of the value of the objective function for a given solution usually implies the need to know how the real system evolves under the effect of the chosen solution or configuration for its freedom degrees. In the case of a manufacturing facility or a supply chain it is commonly unfeasible to experiment with the real system. An approximated way to examine the behavior of the real system for a given configuration, and to foresee its evolution with time, consists in performing a simulation of a model of the real system [14].

For this reason, the following question to deal with, in the statement of an optimization problem such as the one stated in this paper, consists of choosing a formal language for describing in a precise way the model of the real system.

Many manufacturing facilities and supply chains can be considered discrete event systems (DES), or, what is the same, event-driven systems with a finite discrete state space [3]. There are several formalisms especially suitable for the modeling of DES, such as finite state machines, generalized semi-Markov processes, discrete event system specification (DEVS), or the Petri nets [15–18].

The Petri nets belong to a family of formalisms that have been extensively studied and successfully applied in the manufacturing and logistic arena. Their ability to cope with parallel and concurrent behavior, the broad theoretical knowledge of them present in the scientific literature, and their double representation, graphical and algebraic, constitute some examples of features of the Petri nets that make them an appropriate choice for the construction of a simulation model for a DES [3, 19].

A particular formalism belonging to the paradigm of the Petri nets is called high-level Petri nets and a type of them is the colored Petri nets. This formal language in particular has been successfully used as modeling formalism, for example, in industrial, communication, and logistic systems [20]. Among other advantages of the colored Petri nets, the existing

tools for modeling and simulation, the well documented and broad area of application of this formalism, and the theoretical results available for the researchers and practitioners can be mentioned [21, 22].

Conventional colored Petri nets may represent easily the DES with certain types of freedom degrees. In fact, the representation of a freedom degree of a real system can be performed by a single controllable parameter, which can be called undefined parameter or a set of them in the Petri net model. The freedom degrees of a DES can be classified according to different criteria. When the selected modeling formalism is the paradigm of the Petri nets, a useful classification criterion consists of considering the role that its counterpart undefined parameter plays in the model of the real system.

The undefined parameters of the Petri net model of a DES with freedom degrees can be undefined marking parameters, undefined transition-firing parameters, undefined structural parameters, and so on. In particular, undefined marking parameters are related to the initial values of the state variables of the model of the system. Moreover, undefined transition-firing parameters present a strong link with conflicts or, in other words, with exclusive evolutions of the model of the system, when the model itself does not specify which one of the feasible evolutions will, in fact, take place.

A classic approach in the decision making based on the statement of an optimization problem, which includes a Petri net model of the real system for simulation, consists of the development of a Petri net model which contains such types of undefined parameters as marking and transition-firing ones. In a certain number of decision making processes, associated with the management of a manufacturing facility, for example, some cases of scheduling, planning, or routing, it is enough [9, 23, 24].

However, despite their increased difficulty, it is possible to face more complex decision making challenges, where the structure of the real system itself may change. Some examples of this case, where a manufacturing facility can change its structure, can be seen in the following:

- (a) a redesign of the layout of production lines or equipment implying changes in the materials flow;
- (b) the addition (activation) or removal (disconnection) of production lines;
- (c) a significant breakdown of part of the industrial machinery;
- (d) an intensive program of preventive maintenance that requires stopping sequentially parts of the facility, while the rest are still manufacturing products;
- (e) a change in the manufacturing strategy, such as push, pull, or a combination of them.

A classic approach consists of dividing the decision making process into two separate stages. The first of these stages is manual and the second is automated by means of the statement and solving an optimization problem [25].

In the first stage, a reduced set of n promising structural configurations is chosen. Each alternative structural configuration leads to a different Petri net model of the real

system. According to this classic approach, the second stage consists in the solution of n optimization problems, one problem for every one of the structural configurations of the real system. The solution of every optimization problem leads to an optimum associated with the associated structural configuration. As it can be seen, the first stage chooses values for the undefined structural parameters, while the second stage selects values, in an automatic process, for the rest of the undefined parameters.

The best of the optima allows choosing the alternative structural configuration that allows the manufacturing facility to achieve more efficiently the objectives.

Some examples of research performed according to this classic approach are the following. On the one hand Recalde et al. [26] and Zimmermann et al. [27] apply this methodology to the choice of the best manufacturing strategy for a manufacturing process. This selection is made from a small number of alternative structural configurations. On the other hand, Tsinarakis et al. [28] analyze a limited number of alternative structural configurations for a manufacturing facility. Every one of the alternative configurations presents a different level of flexibility. The choice of the best one is performed by comparing their performance.

This approach, however, presents certain drawbacks, some of which can be found in the following lines:

- (a) requirement of experienced and expensive experts;
- (b) slowness in the application of the method;
- (c) reduced amount of alternative structural configurations analyzed;
- (d) possibility of overlooking and skipping promising configurations due to a manual selection;
- (e) low objectivity and repeatability of the process, since it depends largely on the judgment of the decision maker.

In order to overcome the mentioned limitations of the classic approach, the authors have developed and applied a natural methodology of automation of the complete process of selecting a configuration for all the undefined parameters of the system, including the structural ones. This approach implies the statement of a single optimization problem, which can be solved by means of a single process, instead of the n processes provided by the classic technique.

The main problem in the new approach consists of integrating the information contained in the models that correspond to the different alternative structural configurations of the real system into a single optimization problem. This challenge provides, however, one of the main advantages of this decision making methodology, since if a systematic and appropriate technique is used for obtaining a complete model for the single optimization, it is possible to remove large amounts of redundant information from the alternative structural configurations. This property is justified because it is very common that these alternative structural configurations present similarities due to common subsystems.

The authors have proposed several formalisms, belonging to the paradigm of the Petri nets, for representing a model

including alternative structural configurations. All of these formalisms can be considered as Petri nets with exclusive entities, being the exclusive entities, different ways of representing the alternative structural configurations.

In particular, alternative Petri nets, compound Petri nets, and alternatives aggregation Petri nets can be counted among these formalisms.

In this paper, however, a promising formalism is discussed. It is based on the colored Petri nets, which have been modified with the purpose of adding exclusive entities and, hence, leading to the so-called disjunctive colored Petri nets [29]. Among the advantages of this new formalism the possibility of using all the knowledge, experience, applications, and software tools for modeling, simulation, and optimization, which are available for the colored Petri nets, can be considered.

To complete the mentioned methodology for efficient decision making, the authors propose the use of a genetic algorithm as search technique in the solution space of the associated optimization problem [30]. The application of this methodology is illustrated in this same paper by the consideration of a benchmark presented in [25]. In the original version of the problem, the best manufacturing strategy for a production facility is chosen by manual choice from a reduced set of alternative structural configurations and simulation of their behavior.

Section 2, Materials and Methods, includes in Section 2.1 the modeling formalism developed for representing a discrete event system with alternative structural configurations, a disjunctive colored Petri net. Section 2.2 details an algorithm, developed for the construction of a disjunctive colored Petri net, and it is followed by Section 2.3, which provides a simple example of application of the algorithm; Section 2.4 describes a benchmark and outlines the modeling process with the algorithm introduced in the previous section. Section 3, Results and Discussion, includes in Section 3.1 the statement of an optimization problem and the methodology for managing its solution, while Section 3.2 discusses the results of a classic methodology and the proposed methodology. Finally, the results are summarized, the conclusions are provided, and the future research is outlined in Section 4, Conclusions.

2. Materials and Methods

2.1. The Modeling Formalism. The purpose of the next two sections is to state the basis of the construction of a model of a system with alternative structural configurations and its illustration with an application example. In fact, there are diverse ways for performing this representation. The following explanation will provide some general concepts, which are valid for any representation. Nevertheless, in general, the description provided in this paper will focus on two representations. On the one side the text will deal with a classic approach, where an independent model for every alternative structural configuration is created and, on the other side, the construction of a single model for all the alternative structural configurations using the formalism of the disjunctive colored Petri nets will be analyzed.

In the technical literature, there are many introductory texts to the paradigm of the Petri nets, for example, [3, 31, 32]. However, some very succinct comments on the basis of the Petri net formalism will be given.

A Petri net can be represented in two ways. The first one is a bipartite directed graph. This graphical representation is very intuitive and easy to construct and interpret. In it, there are two types of nodes: places and transitions. The first ones, related to the state variables of the model, are represented by circumferences. Moreover, the second type of node, related to the feasible evolution between states, is depicted by rectangles.

In addition, the integration of the values of the state variables in the model of the DES is performed by a set of tokens, represented by dots inside the places. The evolution rules of the Petri net allow the tokens to move from certain places to others under the firing of one or more transitions. This process defines the dynamic of the Petri net by describing the changes in the states of the model. The definition of Petri net presented below has been derived from [15].

Definition 1 (marked Petri net). A *marked Petri net graph* is a weighted bipartite graph described by the five-tuple:

$$R = \langle P, T, \text{pre}, \text{post}, \mathbf{m}_0 \rangle, \quad (1)$$

where $P = \{p_1, p_2, \dots, p_n\}$ is a finite set of places and $T = \{t_1, t_2, \dots, t_m\}$ is a finite set of transitions, where $P \cap T \neq \emptyset$. $\text{pre}: P \times T \rightarrow \mathbb{N}$ is the preincidence function, $\text{post}: T \times P \rightarrow \mathbb{N}$ is the postincidence function, and $\mathbf{m}_0: P \rightarrow \mathbb{N}$ is the initial marking.

The preincidence function defines the directed arcs going from places to transitions, whereas the postincidence function defines the directed arcs from transitions to places. In a weighted or generalized Petri net it is possible to write the weight of the arcs close to their representations.

Instead of interpreting a Petri net as a graph, it can be represented in an algebraic fashion by means of matrices. In fact, from the preincidence function it can be defined as a preincidence matrix \mathbf{W}^- , while from the postincidence matrix it can be defined as a postincidence matrix \mathbf{W}^+ . Both incidence matrices present the same number of rows, equal to the number of places in the Petri net, and the same number of columns, equal to the number of transitions in the Petri net.

Furthermore, the elements that constitute the incidence matrices are the weights of the arcs linking the place and transition that correspond to the row and column of their position.

As it has been summarized before, the firing of a transition is the process that leads to changes in the state of the Petri net model. A necessary condition for a transition to fire is to be enabled.

This process of firing moves tokens of the marking of the places with arcs connecting them to the fired transition in the direction indicated by the direction of the arcs themselves. The weight of a given arc consists of the number of tokens that this arc removes or adds to the place where it is connected when the associated transition fires.

One transition t is enabled if every place with an arc from it to the transition t contains a number of tokens equal to or greater than the weight of the arc.

Once the concept of a Petri net has been introduced, it is possible to discuss the particularities of a Petri net model for a discrete event system with alternative structural configurations.

In fact, a natural way to represent a set of alternative structural configurations of a discrete event system by means of the paradigm of the Petri nets consists of developing an independent model for every one of the configurations. For this reason, this approach is the one that has been followed in most of the application cases found in the literature, solved by a classic approach. Even though in many cases it is not explicitly indicated, all the models representing alternative structural configurations of a DES by means of the formalism of the Petri nets belong to the category of alternative Petri nets.

Given a certain DES with k alternative structural configurations, it is possible to develop k Petri net models, one for every configuration. These models are alternative Petri nets and with them a set of alternative Petri nets, containing the k Petri net models of the configurations, can be constructed.

One important property that the elements of a set of alternative Petri nets should comply is the mutually exclusive evolution [1]. This property is a requirement of the real characteristic of the alternative structural configurations, since only one of them can be chosen as solution for the decision making process. In fact, the undefined structural parameters, which are the representation in the model of the system of certain freedom degrees, can only take a single value for each parameter and the set of all the values defines a single alternative structural configuration. As a consequence, it is not possible to choose and simulate the simultaneous evolution of several alternative Petri nets, only one of them at a time.

More formally, it is possible to state the following.

Definition 2 (mutually exclusive evolution). Let R_i and R_j be two Petri net models that verify $R_i \neq R_j$. They have mutually exclusive evolutions if they comply with these two conditions:

- (i) if $\mathbf{m}(R_i) \neq \mathbf{m}_0(R_i) \Rightarrow \mathbf{m}(R_j) = \mathbf{m}_0(R_j)$;
- (ii) if $\mathbf{m}(R_j) \neq \mathbf{m}_0(R_j) \Rightarrow \mathbf{m}(R_i) = \mathbf{m}_0(R_i)$.

In other words, if an alternative Petri net is allowed to evolve from its initial state, another alternative Petri net of the same discrete event system, which corresponds to a different alternative structural configuration for the freedom degrees of the DES, cannot modify its state simultaneously.

This statement does not refer to the possibility of parallelizing the simulation of different alternative Petri nets with the purpose of performance evaluation [33].

Once the property of mutually exclusive evolution has been introduced, it is possible to define the concept of set of alternative Petri nets in the following way.

Definition 3 (set of alternative Petri nets). Let $S_R = \{R_1, \dots, R_n\}$ be a set of Petri nets.

S_R is also a set of alternative Petri nets if $n > 1$ and $\forall i, j$, such that $i \neq j$ and $1 \leq i, j \leq n$, R_i and R_j comply with the following two conditions:

- (i) R_i and R_j present mutually exclusive evolution;
- (ii) $\mathbf{W}(R_i) \neq \mathbf{W}(R_j)$.

R_j is called the i th alternative Petri net of S_R .

A set of alternative Petri nets can be seen as a group of different feasible models for a given discrete event system. It is common, in industrial and logistic applications, that the different alternative Petri nets present important similarities due to common subsystems.

On the other hand, conceptually, it is also possible to consider a set of alternative Petri nets as a single model of a system with structural freedom degrees. It is the work of the decision makers to reduce the set of alternative Petri nets to a single one, the best one for the objectives of the industrial or logistic systems. During the decision making process, the set of alternative Petri nets is the single model used by the solving methodology to obtain the best solution.

This approach leads to two important and immediate implications.

- (a) It seems logic to try to profit from the redundant information present in the different parts of the model (alternative Petri nets with similarities) in order to remove it and, hence, obtain as a result a model reduced in size. It has to be considered that a small model may require less computational resources for solving a decision problem.
- (b) In the quest for ways of reducing the size of models by removing the redundant information present in a set of alternative Petri nets it is natural to find different representations of the exclusiveness between different parts of the model (originally alternative Petri nets). This topic leads to the possibility of transforming substantially the original concept of set of alternative Petri nets as a group of independent models.

In order to be able to define new formalisms aiming to describe discrete event systems with alternative structural configurations, a previous step with the definition of a set of objects that generalize the idea of set of alternative Petri nets showing mutually exclusive evolution will be introduced. The appropriate concept is that of set of exclusive entities, which are formally introduced in the following [34].

Definition 4 (monotypic set of exclusive entities). A set of exclusive entities, of the same type, associated with a discrete event system D is a set $S_x = \{X_1, \dots, X_n\}$ that complies with the following conditions.

- (i) The elements belonging to S_x are exclusive; that is to say, only one of them can be chosen as a consequence of a decision.
- (ii) $\forall i, j \in \mathbb{N}^*$, such that $i \neq j$ and $1 \leq i, j \leq n$, where $|S_x| = n$; it is verified that $X_i \neq X_j$.

- (iii) $\exists f : S_x \rightarrow S_R$, where $S_R = \{R_1, \dots, R_n\}$ is a set of alternative Petri nets, feasible models of D . f is a bijection $\Rightarrow \forall X_i \in S_x \exists! f(X_i) = R_i \in S_R$ such that R_i is a feasible model for D and $\forall R_i \in S_R \exists! f^{-1}(R_i) = X_i \in S_x$.

A trivial example of set of exclusive entities is a set of alternative Petri nets.

In this paper, a formalism will be derived from the colored Petri nets by the addition of a set of exclusive entities. This formalism is called the disjunctive colored Petri nets and is created with the purpose of making it able to describe a discrete event system with alternative structural configurations.

As it has already been mentioned, colored Petri nets constitute a formalism commonly applied for obtaining models of discrete event systems with the purpose of performing structural analysis, simulations, performance evaluations, or optimizations of discrete event systems [3], among them industrial production systems or supply chains [9]. There is a class of Petri nets, where the colored Petri nets belong; it is the class of the high-level Petri nets [21]. This class of formalisms arises from a combination of programming languages and Petri nets.

One of the more exploited and useful characteristics of the colored Petri nets is the possibility of folding similar subnets repeated in the same Petri net model, hence, reducing the size of the net, in some cases, in a significant proportion. In the folding process itself, some information is lost. In order to keep the amount of information in the folded net rather than in the original one, some attributes (colors) are assigned to the individual tokens, hence, the name of the formalism.

As a consequence, colored Petri nets lead to compact representations of large Petri net models just by adding colors to the tokens.

In a similar way, the formalism applied in this paper performs a folding of the common subnets between the different alternative Petri nets just by adding a special type of colors, which should allow the decision maker to choose an alternative structural configuration.

As it has been mentioned, the tokens of a colored Petri net may present attributes (colors) and these attributes may take values from certain types (color sets).

In addition, it is defined as a choice color for the disjunctive colored Petri nets allowing the folding of subnets belonging to different alternative Petri nets. As a consequence, the choice colors constitute a tool for removing redundant information from a set of alternative Petri nets, leading in this way to a more compact model.

It is possible to define this concept in the following more formal description.

Definition 5 (natural choice color). A natural choice color can be defined as a pair (c, C) containing a natural number $c \in C$; a set $C = \{v_1, v_2, \dots, v_k \mid v_i \in \mathbb{N}^*, 1 \leq i \leq k\}$ is the type of c and the assignment of an actual value for $c, c = v_i \in C$ is the result of a decision.

Given a discrete event system with a set k of alternative structural configurations, it is possible to define a set C , where $|C| = k$. A decision on C implies that a value $v_s \in C$ is chosen to be assigned to the natural choice color $c = v_s$. Hence, this decision has a significant effect on the associated disjunctive colored Petri net model of the DES, since it has been chosen as a single alternative structural configuration and, for this reason, the behavior of the model of the system is the one of the discrete event system associated with the selected configuration. The disjunctive colored Petri net model is then ready for its use in, for example, a performance evaluation or a simulation.

The next result will deal with the fact that a natural choice color is a valid representation of a set of exclusive entities and, as a consequence, it is an essential component of a Petri net model able to describe a discrete event system with alternative structural configurations.

Proposition 6 (a natural choice color is a set of exclusive entities). Let S_R be a set of alternative Petri nets, feasible models of a discrete event system D .

Let the pair (c, C) , where $C \subseteq \mathbb{N}^*$ and $c \in C$, be a natural choice color.

If $|C| = |S_R| \Rightarrow (c, C)$ is a monotypic set of exclusive entities.

Proof. Let us recall Definition 4, which is about a monotypic set of exclusive entities.

By definition, the elements of the set C are exclusive, since only one of them can be assigned to c as a consequence of a decision. Then, (i) of Definition 4 is verified.

The elements of C are natural numbers; hence they belong to the same type. Thus, (ii) is verified.

Furthermore, as $|C| = |S_R|$ is possible to define a bijection $f : C \rightarrow S_R$, such that $\forall c_i \in C \exists! f(c_i) = R_i \in S_R$ and such that R_i is a feasible model for D , $\forall R_i \in S_R \exists! f^{-1}(R_i) = c_i \in C$ as a consequence (iii) is verified. \square

In the realistic decision making process considered in this paper, a decision cannot be changed. Once a decision has been made, it has to be kept in all the operations focusing on evaluating the quality of the choice made. In the same way, when a decision is made and a natural choice color takes a given value, this value should be kept constant during the complete process of quality evaluation of the associated alternative structural configuration.

Due to the previous considerations and considering that a state of the Petri net model is given by its marking, the set of markings reachable from an initial state of the net should be characterized by the same value of the natural choice color. That is to say, the tokens associated with a disjunctive colored Petri net once a decision has been made should present the same value for their attribute which is called natural choice color. The marking with the same value of the natural choice color in a given reachable state of a Petri net is called monochrome choice marking, as it can be seen below.

Definition 7 (monochrome choice marking). Let R be a marked colored Petri net. Let \mathbf{m} be any feasible marking of

R , reached from the initial marking \mathbf{m}_0 after the sequence of transitions σ is fired. Let c be a natural choice color.

If every token of \mathbf{m} verifies that c has the same value, then the marking \mathbf{m} of the Petri net R is said to be a monochrome choice marking.

However, for describing the uniformity of the value of the natural choice color associated with a disjunctive colored Petri net, after making a decision on the alternative structural configuration, it is necessary to extend the characterization of the choice marking to all the states reachable from the initial marking. The first step to deal with this description is to define a choice subset of reachable markings as it is detailed in the following.

Definition 8 (choice subset of reachable markings). Let R be a marked colored Petri net.

$S_r(R, c)$ is called choice subset of reachable markings of R if it contains all the reachable states of the model under a single chosen alternative structural configuration. The selected configuration is modeled by the natural choice color (c, C) and the elements of the subset by means of the markings of the Petri net.

As a second step, it should be specified that the natural choice color keeps the same value for every reachable state of the model.

Definition 9 (monochrome set of reachable markings). Let R be a marked colored Petri net.

A choice subset of reachable markings is called monochrome set of reachable markings when any token of every marking of the set shows the same choice color.

At this point, it is possible to define formally a disjunctive colored Petri net.

Definition 10 (marked disjunctive colored Petri net). A marked disjunctive colored Petri net is a nine-tuple $R = \langle P, T, F, \mathbf{m}_0, \Sigma, V, cs, g, e \rangle$, where

- (1) P and T are finite sets of places and transitions, respectively, that verify $P \cap T = \emptyset$;
- (2) $F \subseteq P \times T \cup T \times P$ is a set of directed arcs;
- (3) \mathbf{m}_0 is an initial monochrome choice marking;
- (4) Σ is a finite set of nonempty color sets and (c, C) , such that $c \in C$ is a natural choice color, where $C \in \Sigma$;
- (5) V is a finite set of typed variables such that type $[v] \in \Sigma$ for all variables $v \in V$;
- (6) $cs: P \rightarrow \Sigma$ is a color set function that assigns a color set to each place;
- (7) $g: T \rightarrow \text{EXPR}_V$ is a guard function that assigns a guard to each transition t such that type $[g(t)] = \text{boolean}$;
- (8) $e: F \rightarrow \text{EXPR}_V$ is an arc expression function that assigns an arc expression to each arc a .

Moreover, every choice subset of reachable markings is a monochrome set of reachable markings.

Not all the attributes associated with the tokens of a disjunctive colored Petri net are natural choice colors. In fact, there may be a lot of different colors, not related to decision making processes and alternative structural configurations. In this case, the property of monochrome choice marking does not apply to the attributes not related to the natural choice color. For this reason the nonchoice attributes may have any value.

In this section, a brief introduction in the formalism that will be used in this document for the development of a decision making methodology has been presented. In the following, an algorithm for constructing a model with a disjunctive colored Petri net and some application examples will be detailed.

2.2. The Construction of a Model. There are several possibilities for constructing a model of a discrete event system by using the formal language given by the disjunctive colored Petri nets. One of the causes of the differences between the mentioned methodologies lies on the source of the data used for creating the model.

In the present document a methodology that allows constructing a disjunctive colored Petri net from a set of alternative Petri nets will be described and applied. The reason for choosing a set of alternative Petri nets as initial formalism in which a discrete event system with alternative structural configurations is based is the fact that this set constitutes a natural but inefficient way of modeling such a DES. In fact, most of the case studies, present in the literature, that use Petri net models of DES with alternative structural alternatives, construct such models by means of sets of alternative Petri nets. See, for example, Tsinarakis et al. [28], Recalde et al. [26], Zimmermann et al. [27], or Zhou and Venkatesh [25].

The original alternative Petri nets may be colored Petri nets (high-level Petri nets) or low-level Petri nets.

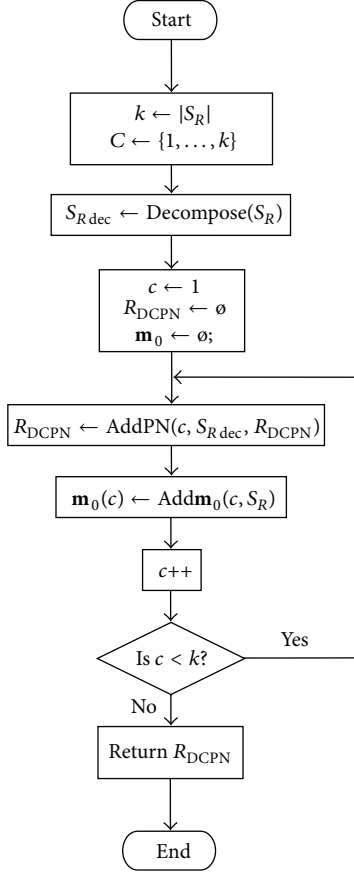
In the present section an algorithmic procedure will be introduced in order to describe in a formal and precise way the construction of a compact disjunctive colored Petri net model by successively folding or aggregating subnets from different alternative Petri nets.

In the algorithm, presented in Figure 1, the original set of alternative Petri nets is represented by S_R and its type of data is represented by SPN. These original alternative Petri nets are obtained as models of every alternative structural configuration for the freedom degrees of the discrete event system.

The first step of the algorithm consists of creating a choice color appropriate for the cardinality of the original set of alternative Petri nets S_R .

The second step is based on the application of the function decomposition (S_R). This function divides every alternative Petri net of S_R into a set of subnets and link transitions.

The third step creates three variables: c , which is part of the definition of natural choice color; R_{DCPN} , or disjunctive color Petri net model, which initially is an empty object but it will be created by the folding of different subnets and link transitions coming from the decomposed alternative

FIGURE 1: Algorithm S_R to DCPN(SPN: S_R).

Petri nets; finally \mathbf{m}_0 is the initial marking of the resulting disjunctive colored Petri net model.

The fourth stage is composed of various steps inside a loop. The loop will be executed so many times as the cardinality of the set of alternative Petri nets. The steps inside every execution of the loop are as follows.

- (a) The first step in the loop consists of calling the function $\text{AddPN}(c, S_{Rdec}, R_{DCPN})$. This function allows the folding of the c th alternative Petri net of S_R to the disjunctive colored Petri net in process of being constructed R_{DCPN} . This folding requires the following tasks.
 - (i) Aggregation of the subnets of the c th alternative Petri net is not shared by the disjunctive colored Petri net constructed so far. Moreover, the transitions not belonging to a subnet, also named link transitions, are also included in the model.
 - (ii) The natural choice color that corresponds to the c th alternative Petri net is associated with the places shared by the aggregated structure and the alternative Petri net.
 - (iii) Introduction of a guard function for the link transitions added to the model R_{DCPN} . In fact, the guard will be the choice color c .

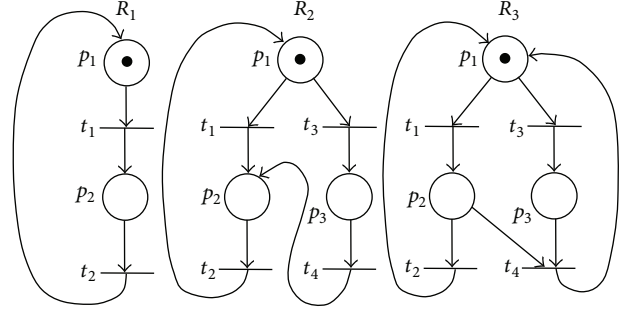


FIGURE 2: Alternative Petri nets.

- (b) The second step in the loop consists of calling the function $\text{Addm}_0(c, S_R)$. This function completes with the initial marking of the alternative Petri net, the initial marking of the R_{DCPN} . This newly associated initial marking is related to the value of the natural choice color.

2.3. Example of Application of the Algorithm. In this section, the previously presented algorithm for constructing a disjunctive colored Petri net from a set of alternative Petri nets is applied to a simple example with the purpose of illustrating the different steps that compose the procedure.

This example will begin with a set of three alternative Petri nets presented in Figure 2.

In the set of alternative Petri nets represented in Figure 2 three very simple low-level Petri nets (without attributes associated with the tokens) can be seen. The initial marking of all of them consists of a simple token in a place, which in the three alternative Petri nets is accidentally called with the same label p_1 .

The first step of the algorithm consists of making

- (a) $k \leftarrow |S_R|$; hence $k = 3$;
- (b) $C \leftarrow \{1, \dots, k\}$; hence $C = \{1, 2, 3\}$.

In the second step of the algorithm, it is necessary to decompose the alternative Petri nets into subnets and link transitions. This decomposition is performed with the objective of finding as many common subnets in the alternative Petri nets as possible.

In this example, in the same way as it is usual in the manufacturing or logistic sectors, it is possible to find a common subnet. In fact, the whole alternative Petri net R_1 can be considered a shared subnet, since, as it can be seen in Figures 3 and 4, it can be found, respectively, as subnet in R_2 and R_3 .

In fact, drawn with continuous line, the common subnet between R_1 and R_2 can be seen in Figure 3. Later on, in a subsequent paragraph, it will be seen how this subnet is also shared by R_3 . In this same figure, drawn with dotted line, the other subnet, p_3 , and the two link transitions, t_3 and t_4 , can be seen.

Similarly as it has been explained in the previous paragraph, Figure 4 shows the decomposition of the alternative Petri net R_3 into subnets and link transitions between them.

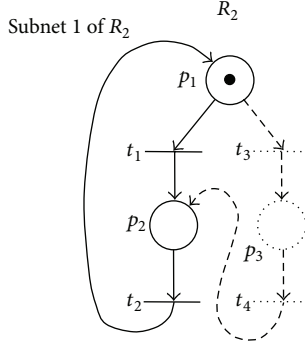


FIGURE 3: Alternative Petri net R_3 decomposed in subnets and link transitions.

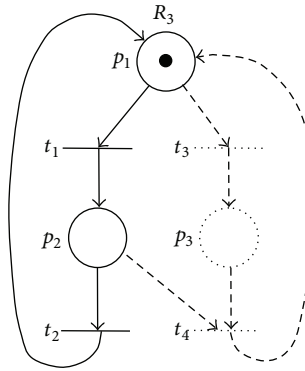


FIGURE 4: Alternative Petri net R_3 decomposed in subnets and link transitions.

In continuous line, the common subnet with R_2 and R_3 has been represented. In dotted line, the subnet p_3 , as well as the link transitions t_2 and t_3 , has been depicted.

Once a set of additional variables have been initialized, according to the algorithm, it is possible to begin with the loop.

The first step is $R_{DCPN} \leftarrow \text{AddPN}(c, S_{R_{dec}}, R_{DCPN})$, where $c = 1$. It consists in creating the first seed of the disjunctive colored Petri net model. In this step it will be equal to R_1 .

After creating the initial marking of the disjunctive colored Petri net with the token provided by R_1 , associated with the natural choice color $c = 1$, the natural choice color is increased ($c++$) and the ending condition of the loop is evaluated.

It is verified that $1 = c < k = 3$. As a consequence the loop is executed once more.

In the following iteration, $R_{DCPN} \leftarrow \text{AddPN}(c, S_{R_{dec}}, R_{DCPN})$ is performed, where $c = 2$. The result can be seen in Figure 5.

In this case, it is verified that $2 = c < k = 3$. As a consequence the loop is executed for the last time.

In the third iteration $R_{DCPN} \leftarrow \text{AddPN}(c, S_{R_{dec}}, R_{DCPN})$, where $c = 3$. The result has been represented in Figure 6.

It is possible to simplify the net by merging the transitions 3 and 6 in the way shown in Figure 7.

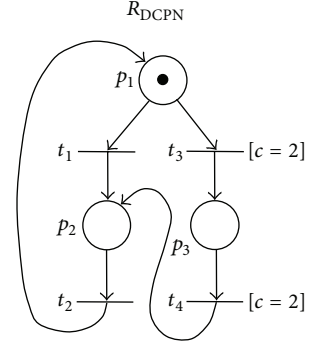


FIGURE 5: Disjunctive colored Petri net after the second iteration.

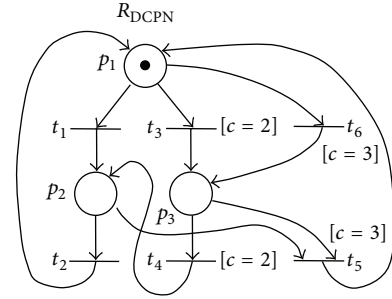


FIGURE 6: Disjunctive colored Petri net after the third iteration.

The disjunctive colored Petri net represented in Figure 7 is the resulting model obtained from the application of the algorithm. As it can be seen, its size is much more reduced than the size of the original set of alternative Petri nets depicted in Figure 2. Nevertheless, the information bearing both models is exactly the same and, hence, the description they perform of the original discrete event systems with alternative structural configurations is the same.

The initial marking can be defined in the following way:

if ($d=1$) then $M_0 = (1^1, 0, 0)$;
 else if ($d=2$) then $M_0 = (1^2, 0, 0)$;
 else if ($d=3$) then $M_0 = (1^3, 0, 0)$.

Meaning that, according to the value of the natural choice color, which in its term depends on the choice of the decision maker, the place p_1 will hold a single token with the attribute equal to the value of the natural choice color. The places p_2 and p_3 will not contain any token in the initial state of the disjunctive colored Petri net.

Once a simple example of application of the modeling algorithm has been discussed, it will be possible, in the following section, to apply this modeling methodology to a manufacturing facility.

2.4. Modeling of a Manufacturing Facility. In this section, the modeling methodology presented in the previous paragraphs for obtaining a disjunctive colored Petri net will be applied to a benchmark, developed by other authors [25] for the same purpose than the aim of this paper: showing a decision making procedure for the choice of an appropriate production strategy. A qualitative representation of the manufacturing

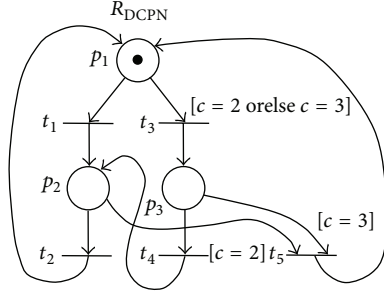


FIGURE 7: Simplified disjunctive colored Petri net after the third iteration.

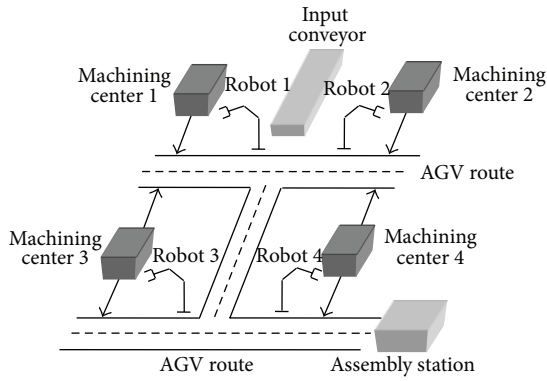


FIGURE 8: Qualitative layout of the manufacturing facility.

facility is presented in Figure 8, while in Figure 9 the raw materials mix and the sequence of required machines for the appropriate manufacturing of one of the final products PR_1 can be seen.

The first stage in the manufacturing facility is performed in parallel by two machining centers (1 and 2). Every one of these machining centers is able to process different types of raw materials, from a set of four types called A, B, C, and D (see Figure 9). In order to provide these two first machining centers with raw materials, there is an initial conveyor and a robot per machining center, to load the raw parts to the centers.

The manufacturing facility counts with an undetermined number of automatic guided vehicles (AGV) for conveying the semifinished parts from this first stage to the second row of machining centers (3 and 4) and from these centers to a final assembly station. In fact, the AGV can remove all the semifinished parts, load machining centers 3 and 4, and carry the parts to the last assembly stage going over different routes. One of the objectives of the decision making problem consists of choosing an optimal number of AGV and the best route or routes to be followed by the AGV.

The modified benchmark, presented in this section, shows a number of freedom degrees, which define the feasible configurations that would lead the facility to different behaviors. The objectives of the decision makers, for

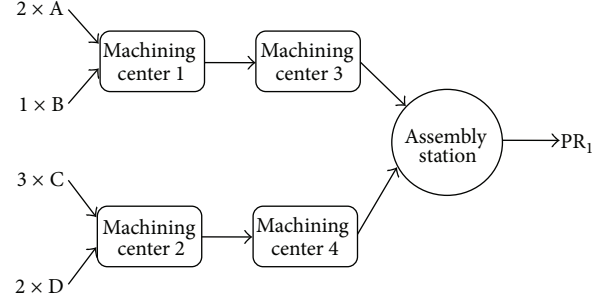


FIGURE 9: Sequence of production of PR_1 .

choosing one configuration or another, are diverse. In particular the following ones can be considered:

- (a) maximal yield;
- (b) minimal WIP (work in process);
- (c) maximal utilization rate of the production equipment.

These objectives can be difficult to reach simultaneously, since improving the behavior of the system according to one of the objectives might make this behavior with respect to another objective worse.

On the other hand, some of the freedom degrees present in the manufacturing facility are the following.

- (1) The production lot size.
- (2) The lot size conveyed by the automatic guided vehicles.
- (3) The manufacturing strategy. From a basic pull and push production policies proposed in the original benchmark [25], 6 different combinations of both approaches are performed with the purpose of removing bottlenecks and improving the degree of achievement of the objectives.
- (4) The layout and number of routes for the AGV. After analyzing in depth the distribution of the machinery and the sequence of manufacturing, three different layouts for the routes of the automatic guided vehicles are defined.
- (5) The number of AGV in each route.

Some of the freedom degrees are modeled by means of undefined marking parameters, while others are associated with different alternative structural configurations.

Following a classic approach, also followed by [25], the discrete event system associated with every alternative structural configuration can be modeled by independent models or alternative Petri nets. In fact, this is a very intuitive and natural way to obtain a model of a DES with alternative structural configurations, yet inefficient.

The freedom degrees of the discrete event system that leads to alternative structural configurations are the different layouts for the routes of the AGV (3 have been selected in this case study), as well as the diverse manufacturing and the feasible production policies (8 have been envisaged

as potentially efficient in total). The combination of every routing of the AGV and the manufacturing and production strategies produces 24 alternative structural configurations.

The main differences between the 24 alternative Petri nets that can be constructed from the manufacturing facility, taking into account every one of the alternative structural configurations, are the link transitions representing the different flow of information regarding each production policy. There are many shared subnets, representing the common models for the manufacturing centers or the assembly station.

In Figure 10 an example of alternative Petri net has been depicted. The representation is called an abbreviated representation, since it does not represent in detail the complete structure of the Petri net [35]. In fact, some details, needed for the precise definition of the Petri net model, are omitted. The purpose of this representation is to underline the important information for the decomposition of the alternative Petri net into subnets and link transitions as seen in the algorithm S_R to DCPN(SPN: S_R). Other information than subnets and link transitions is not indicated in an abbreviated representation of a Petri net.

The information given by the abbreviated representation of a Petri net will also be important for the iterative process of construction of a disjunctive colored Petri net model, since, as it will be remembered from the algorithm S_R to DCPN(SPN: S_R), the nonshared subnets and link transitions are aggregated to the model according to the successive iterations.

The abbreviated representation depicted in Figure 10 represents explicitly

- (a) link transitions, named t_i , where i is a natural number;
- (b) subnets represented by the picture of a cloud and named according to the following ideas. R_a , R_b , R_c , and R_d are the subnets corresponding to the machining centers. R_e is the model of the assembly station. Finally, R_h and R_g are the subnets of two different routes for the AGV;
- (c) other structural undefined parameters, which play the role of weights of some arcs and represent the size of the manufacturing and conveying lot size of every raw material or semifinished parts A, B, C, and D, in the form of S_a , S_b , S_c , and S_d , respectively.

The main differences between the alternative Petri nets are the link transitions and the subnets that correspond to the different feasible routes for the AGV. This fact can be seen in Figures 10 and 11, where the subnets of the machining centers R_a , R_b , R_c , and R_d , as well as the subnet of the assembly station R_e , are shared by every alternative Petri net. As a consequence, in the disjunctive colored Petri net 23×5 , subnets are removed from the model of the manufacturing facility, when compared to the set of alternative Petri nets.

Once all the 24 alternative Petri nets have been decomposed into subnets and link transitions, trying to find as many shared subnets as possible, it is possible to apply the algorithm S_R to DCPN(SPN: S_R) for obtaining the resulting disjunctive colored Petri net. The application of the mentioned algorithm implies an iterative process of adding the nonshared subnets

and the link transitions provided for every alternative Petri net to the disjunctive colored Petri net.

The third iteration in the aggregation of alternative Petri nets to the single model has been depicted in Figure 11. In this figure the guards of some transitions have been abbreviated in the following form:

- (a) c_1 means $[c = 1]$;
- (b) $c_1 + c_2$ means $[c = 1 \text{ or else } c = 2]$;
- (c) $c_2 + c_3$ means $[c = 2 \text{ or else } c = 3]$;
- (d) $c_1 + c_2 + c_3$ means $[c = 1 \text{ or else } c = 2 \text{ or else } c = 3]$.

As a result of the iterative process, a final disjunctive colored Petri net is constructed. Its size is larger than any of the 24 alternative Petri nets that represent the assignment of the different alternative structural configurations to the manufacturing facility. For this reason performing a simulation of the evolution of an alternative Petri net is faster than doing the same with the disjunctive colored Petri net.

Nevertheless, the disjunctive colored Petri net is much smaller than the complete set of alternative Petri nets. As a consequence, performing a simulation of the evolution of the disjunctive colored Petri net is faster than calculating 24 simulations for the evolutions of the 24 alternative Petri nets.

In order to give an idea of the relative sizes of the different Petri net models, the size of the incidence matrices, which represent the structure of a Petri net, can be considered.

In this case study, the size of the disjunctive colored Petri net is 2.5 times larger than the size of a single alternative Petri net. It has to be considered that the size of every alternative Petri net is approximately the same.

Nevertheless, the disjunctive colored Petri net and the set of alternative Petri nets contain exactly the same information. The difference is that the shared subnets provide the set of alternative Petri nets with blocks of repeated data and, hence, redundantly inflating the size of the model without any advantage.

On the other hand, the disjunctive colored Petri net is a completely functional model to be used with any tool developed for the broadly used colored Petri nets, with the purpose of performing structural analysis, performance evaluation, simulation, or optimization.

The next section will be focused on the solution of the optimization problem stated on the disjunctive colored Petri net model of the manufacturing facility.

3. Results and Discussion

3.1. Solving the Decision Problem. The methodology proposed in this section aims to solve a decision making process, concerning a manufacturing facility with alternative structural configurations. The decision problem is to be solved by means of the simulation of a Petri net model of the system. A classic approach consists of solving as many optimization problems as alternative structural configurations can be selected, whereas the proposed approach reduces the process to a single optimization based on a single disjunctive colored Petri net model.

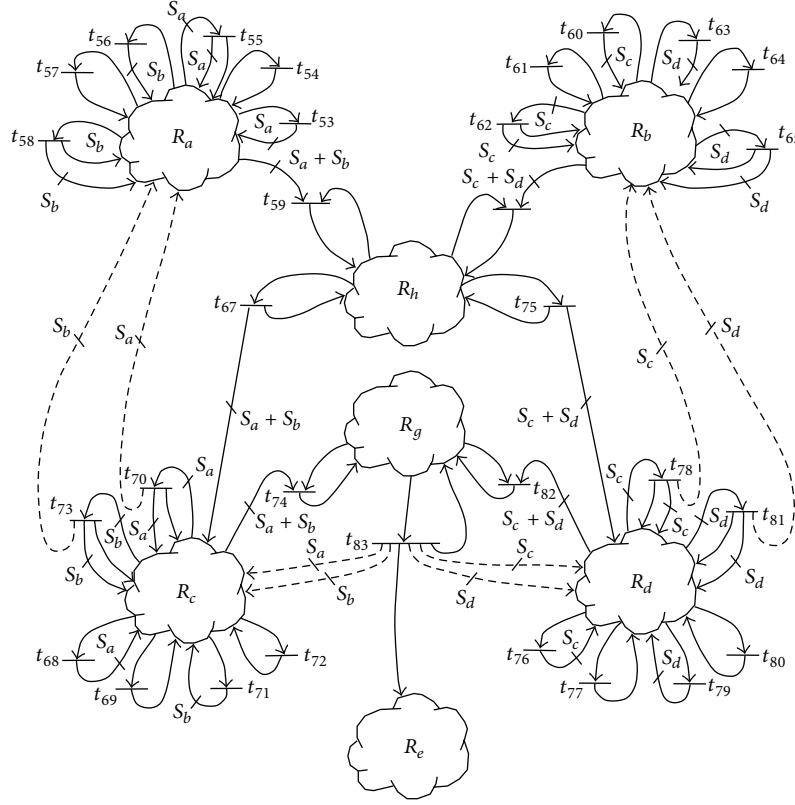


FIGURE 10: Example of alternative Petri net.

The decision making problem requires knowing the best values for the controllable parameters of the original discrete event system, also called freedom degrees. In other words, the goal of the process is to select the best alternative configuration of the manufacturing facility to achieve its objectives.

As a consequence, the decision problem can be translated into an optimization problem, where an automatic search in the solution space measures the quality of the different feasible solutions until the best one, or more modestly if the exploration is reduced to a promising region of the pool of solutions, a good enough one is found.

The quality of a feasible solution is given with regard to how close to the achievement of the objectives arrives the manufacturing facility, thanks to the associated alternative configuration. This quality is calculated by the objective or multiobjective function, whose evaluation requires knowing the evolution of the production system, performed by simulation.

More formally, it is possible to define an optimization problem appropriated for a disjunctive colored Petri net model in the following way.

Definition 11 (optimization problem based on a disjunctive colored Petri net). Maximize $f(x)$, where $f : D \rightarrow \mathbb{R}^n$ is called multiobjective function and D is the domain of f , constructed by the following constraints.

- R is a disjunctive colored Petri net model.
- S_c is a set of additional constraints, verifying that $S_c = \{g_1(x) = 0, \dots, g_{i_1}(x) = 0, g_{i_2}(x) < 0, \dots, g_{i_3}(x) < 0\}$, where $\forall g_k \in S_c, g_k : D_{g_k} \rightarrow \mathbb{R}$ and $D \subseteq \cap D_{g_k}$.
- There is a type of variable for every component of a solution. A solution variable of an optimization problem has the following form: $x = (x_1, x_2, \dots, x_{n_s})$, $x_i \in D_{x_i}$ and $\forall i \in \mathbb{N}^*$, such that $1 \leq i \leq n_s$, where $D \subseteq D_{x_1} \times D_{x_2} \times \dots \times D_{x_{n_s}}$.

This optimization problem can be solved by means of diverse methodologies. Due to the combinatorial explosion associated with the size of the solution space, the exhaustive search, also called brute force search, cannot be applied, since the computational resources required for its implementations lead to a nonpractical solving time.

A more realistic approach consists of a probabilistic guided search in the pool of solutions, which is the solving methodology adopted in this section.

For applying this methodology a metaheuristic, in particular a genetic algorithm, has been selected for guiding the search of promising solutions. In order to assure a fast convergence of the optimization process, some parameters of the genetic algorithm should be adjusted, such as the type of crossover, the length of the chromosome sections cut in the application of this type of crossover, and the mutation and mortality rates [13].

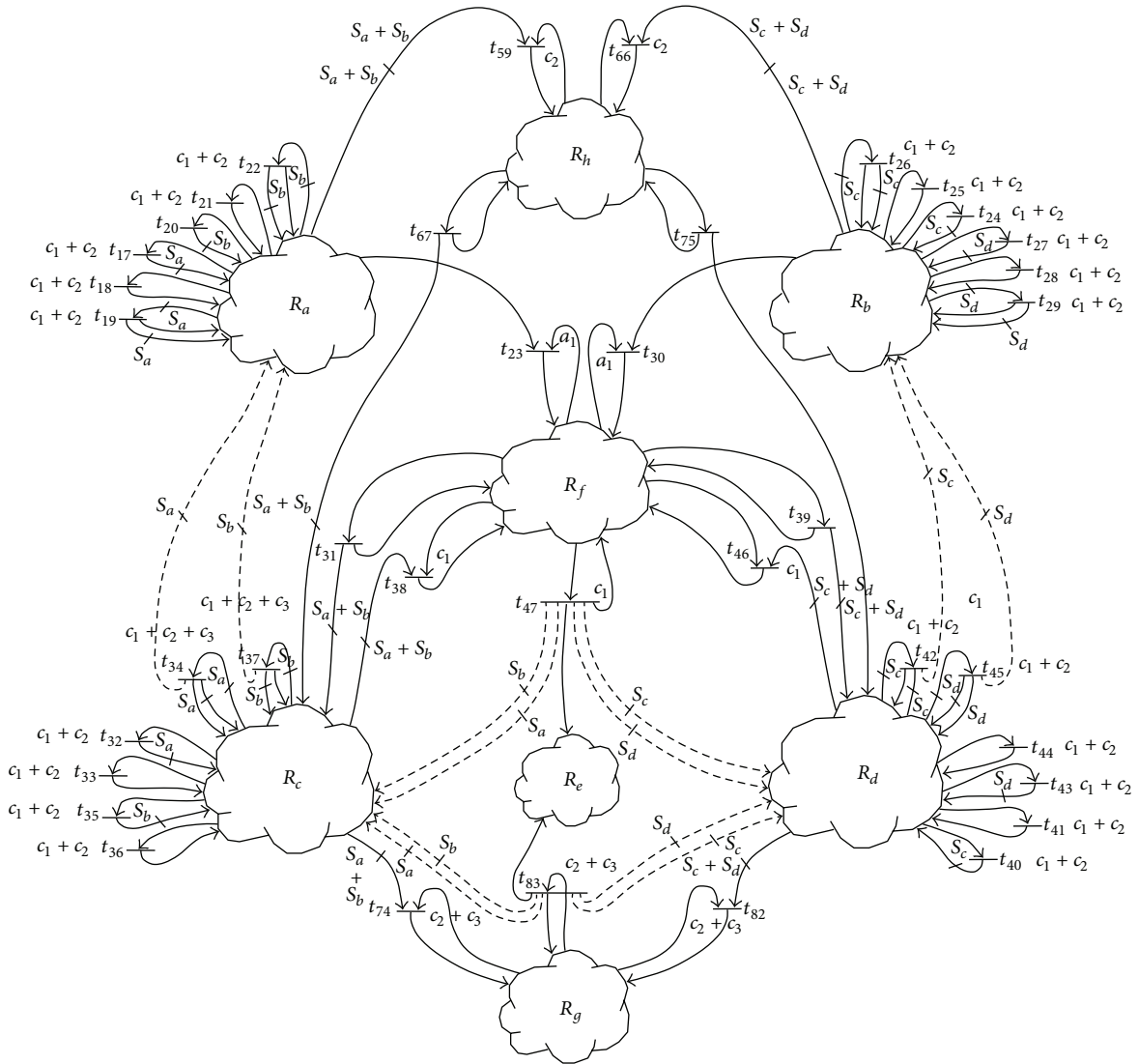


FIGURE 11: Basic abbreviated representation of the third step in the folding of subnets to configure the disjunctive colored Petri net.

The structure of a single chromosome has been composed by means of the concatenation of the values that a certain alternative configuration assigns to every freedom degree of the manufacturing facility.

In fact, this solving methodology has been applied to solve the decision problem two times. The first one following the classic approach of stating and solving 24 optimization problems, so many as alternative Petri net models, and the second one by stating and solving a single optimization problem based on the disjunctive colored Petri net model. The comparative results of both solution processes are presented in the following section.

3.2. Comparing the Classic and New Approaches. As it has been mentioned in the previous section, two decision making processes have been implemented for comparing their relative performance.

- (a) The first is classic methodology, where the model of the DES consists of a set of 24 alternative Petri nets. In this case, the decision problem has been tackled by means of the statement and solution of 24 optimization problems; one for every element of the set of alternative Petri nets. In each one of the optimization problems, the genetic algorithm has been implemented by the calculation of a population of 50 solutions during 15 generations. As a consequence, 750 solutions of the solution space have been evaluated and their quality has been calculated.
- (b) The second is the proposed methodology where the model of the DES consists of a single disjunctive colored Petri net. In this case, the decision problem has been addressed by the statement and solution of a single optimization problem. For solving this

problem, several combinations in the size of the population and the number of generations are considered.

The results of the application of both methodologies are compared in the following paragraphs.

Two parameters have been calculated for a better comparison between the performances of both methodologies.

(a) *Quality of the Solution*. This parameter has been calculated in % of variation of the best value of the objective function of the proposed methodology with respect to the classic one, divided by this last one. More in detail, let us define the following terms:

f_1 = best value of the objective function from the methodology based on 24 alternative Petri net models;

f_2 = best value of the objective function from the methodology based on a single disjunctive colored Petri net model.

Hence, the relative quality in the solution found by both methodologies, Q , has been defined in the following way:

$$Q = \left(\frac{f_2 - f_1}{f_1} \right) \cdot 100. \quad (2)$$

The values obtained in the implementation of both methodologies can be seen in Table 1. In this table, the first column represents the number of solutions that have been explored by means of the methodology based on the disjunctive colored Petri net. It has to be noted that the classic methodology evaluates always 750 solutions per alternative Petri net, meaning $750 \times 24 = 18000$ solutions in the whole process.

As it can be seen in Table 1, when the methodology based on the disjunctive colored Petri net explores (calculates the quality of) a reduced number of solutions, the best value for the objective function is slightly worse than using the classic approach. However, as the number of explored solutions rises, the trend changes significantly.

It has to be remembered that the objective function measures the degree of achievement of the objectives of the manufacturing facility.

As a conclusion, with a number of explored solutions being significantly lower, the methodology based on the disjunctive Petri net model provides a better solution than the classic approach.

(b) *Speed-Up*. This parameter summarizes the computer resources required to finish the optimization process by dividing the time required by the different methodologies to complete the process and obtain the best solution. Again, some parameters will be defined:

t_1 = time required by the methodology based on a set of 24 alternative Petri nets to be completed;

t_2 = time required by the methodology based on a disjunctive colored Petri net to be completed.

TABLE 1: Relative quality in the solution.

Number of evaluated solutions	Q
750	-0.637%
1000	-0.297%
1500	0.224%
2000	0.699%
2250	2.039%
3000	3.679%
7500	7.471%

TABLE 2: Speed-up.

Number of evaluated solutions	su
750	9.501
1000	6.520
1500	5.121
2000	4.613
2250	4.058
3000	2.976
7500	2.162

The speed-up defined in this comparative has been calculated in the following way:

$$su = \frac{t_2}{t_1}. \quad (3)$$

The values obtained in the implementation of both methodologies can be seen in Table 2. As it has been mentioned in the case of Table 1, in Table 2, the first column represents the number of solutions that have been explored by means of the methodology based on the disjunctive colored Petri net.

The second parameter of this comparative process of both tested methodologies, the speed-up, indicates that, in all the tests, the methodology based on the disjunctive colored Petri nets outperforms the classic approach.

Combining both Tables 1 and 2, it is possible to find cases with a speed-up between 2 and 4 and improvements in the quality of the solution between 0.7 and 7.4%.

As it has been seen, in this decision making process, a compact single model based on the formalism of the disjunctive colored Petri nets outperforms a classic approach based on a set of alternative Petri nets.

4. Conclusions

This paper outlines the basis, details, and the practical application of a methodology for the development of decision making support tools based on the statement and solution of optimization problems by means of the simulation of Petri net models of the real system.

More specifically, the decisions are made on real systems with alternative structural configurations, which traditionally have been faced by constructing a set of alternative Petri nets

in a semimanual process, which rely on the subjectivity of expensive human experts.

In order to overcome the limitations of the classic approach and to develop an automatic procedure, a special kind of formalisms, based on the Petri nets, is proposed. All of them are characterized by the so-called set of exclusive entities. One of these formalisms, the disjunctive colored Petri net, is defined and a construction algorithm, from a set of alternative Petri nets, is presented.

A benchmark, based on the decision making related to the management of a manufacturing facility, is described and an optimization problem is presented for solving the decision making process.

Both, a classic methodology based on a set of alternative Petri nets and the proposed methodology based on a single disjunctive colored Petri net are applied to the benchmark.

The results, compared by means of two parameters, the quality of the best solution found, and the speed-up, show that the proposed methodology outperforms the classic one.

The explanation for these good results of the methodology that implements a single optimization based on a disjunctive colored Petri net can be summarized in the following.

- (a) The single optimization focuses all the computational resources on the most promising area of the solution space, while the classic process with n optimizations decomposes the solution space in n different regions that are explored investing computational resources despite the fact that some of them might not give good solutions. For this reason, the single optimization based on the disjunctive colored Petri net model profits more efficiently of the resources; hence, it requires less time to find better solutions.
- (b) The success of the proposed methodology in this benchmark does not imply its success in the decision making of any discrete event system with alternative structural configurations. In fact, the existence of shared subnets between the alternative Petri nets is critical. Otherwise, the disjunctive colored Petri net model might be so large, compared with the set of alternative Petri nets that the simulation time required for evaluating the quality of the solutions would be too large comparing to the simulation of a single alternative Petri net. However, the experience tells that the existence of shared subnets between alternative Petri nets in practical modeling of industrial and logistic systems is common.

As a consequence of the previous statements, the methodology based on a single disjunctive colored Petri net arises as a promising methodology for the decision making process on discrete event systems with alternative structural methodologies.

As a future research line it is proposed to continue in the application of this methodology and variations of it to real case studies.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

- [1] J. I. Latorre, E. Jiménez, J. Blanco, and J. C. Sáenz-Díez, "Decision support in the rioja wine production sector," *International Journal of Food Engineering*, vol. 9, no. 3, pp. 267–278, 2013.
- [2] A. G. Bruzzone and F. Longo, "An advanced system for supporting the decision process within large-scale retail stores," *Simulation*, vol. 86, no. 12, pp. 742–762, 2010.
- [3] C. G. Cassandras and S. LaFortune, *Introduction to Discrete Event Systems*, Springer, New York, NY, USA, 2nd edition, 2008.
- [4] M. Liu, S. Wang, and Z. Li, "Supervisor reconfiguration for deadlock prevention by resources reallocation," *Journal of Applied Mathematics*, vol. 2013, Article ID 315894, 11 pages, 2013.
- [5] J. I. Latorre, E. Jiménez, and M. Pérez, "Sequence of decisions on discrete event systems modeled by Petri nets with structural alternative configurations," *Journal of Computational Science*, 2013.
- [6] E. Jiménez, A. Tejada, M. Pérez, J. Blanco, and E. Martínez, "Applicability of lean production with VSM to the Rioja wine sector," *International Journal of Production Research*, vol. 50, no. 7, pp. 1890–1904, 2012.
- [7] E. Jiménez and M. Pérez, "Simulation and optimization of logistic and production systems using discrete and continuous Petri nets," *Simulation*, vol. 80, no. 3, pp. 142–152, 2004.
- [8] M. M. Mota and M. A. Piera, "A compact timed state space approach for the analysis of manufacturing systems: key algorithmic improvements," *International Journal of Computer Integrated Manufacturing*, vol. 24, no. 2, pp. 135–153, 2011.
- [9] M. A. Piera and G. Mušič, "Coloured Petri net scheduling models: timed state space exploration shortages," *Mathematics and Computers in Simulation*, vol. 82, no. 3, pp. 428–441, 2011.
- [10] B. Yao, C. Yang, J. Hu, J. Yao, and J. Sun, "An improved ant colony optimization for flexible job shop scheduling problems," *Advanced Science Letters*, vol. 4, no. 6-7, pp. 2127–2131, 2011.
- [11] Y. L. Pan, Y. S. Huang, Y. S. Weng, W. Wu, and M. Jeng, "Computationally improved optimal control methodology for linear programming problems of flexible manufacturing systems," *Journal of Applied Mathematics*, vol. 2013, Article ID 294835, 11 pages, 2013.
- [12] T. Sun and X. Ye, "A model reduction method for parallel software testing," *Journal of Applied Mathematics*, vol. 2013, Article ID 595897, 13 pages, 2013.
- [13] J. I. Latorre, E. Jiménez, and M. Pérez, "A genetic algorithm for decision problems stated on discrete event systems," in *Proceedings of the 12th UKSim International Conference on Modelling and Simulation (UKSim '10)*, pp. 86–91, Cambridge, UK, March 2010.
- [14] F. Longo, L. Nicoletti, A. Chiurco, A. Solis, M. Massei, and R. Diaz, "Investigating the behavior of a shop order manufacturing system by using simulation," in *Proceedings of the Emerging M and S Applications in Industry and Academia Symposium (EAI '13)*, article 7, pp. 47–54, April 2013.
- [15] M. Silva, "Introducing Petri nets," in *Practice of Petri Nets in Manufacturing*, F. di Cesare, Ed., pp. 1–62, Chapman & Hall, 1993.

- [16] S. Huang, Z. Xiang, and H. R. Karimi, "Stabilization and controller design of 2D discrete switched systems with state delays under asynchronous switching," *Abstract and Applied Analysis*, vol. 2013, Article ID 961870, 12 pages, 2013.
- [17] A. Kuchkarov, G. Ibragimov, and A. Sotvoldiev, "Linear discrete pursuit game problem with total constraints," *Abstract and Applied Analysis*, vol. 2013, Article ID 840925, 5 pages, 2013.
- [18] Y. Yin, Y. Liu, and F. Liu, "A simplified predictive control of constrained Markov jump system with mixed uncertainties," *Abstract and Applied Analysis*, vol. 2014, Article ID 475808, 7 pages, 2014.
- [19] X. Hu, L. Zhao, and W. Li, "Petri net-based R&D process modeling and optimization for composite materials," *Journal of Applied Mathematics*, vol. 2013, Article ID 306704, 12 pages, 2013.
- [20] B. Barzegar, H. Motameni, and H. Bozorgi, "Solving flexible job-shop scheduling problem using gravitational search algorithm and colored Petri net," *Journal of Applied Mathematics*, vol. 2012, Article ID 651310, 20 pages, 2012.
- [21] K. Jensen and L. M. Kristensen, *Colored Petri Nets. Modelling and Validation of Concurrent Systems*, Springer, 2009.
- [22] M. Mujica, M. A. Piera, and M. Narciso, "Revisiting state space exploration of timed coloured Petri net models to optimize manufacturing system's performance," *Simulation Modelling Practice and Theory*, vol. 18, no. 9, pp. 1225–1241, 2010.
- [23] G. Mušič, "Petri net based scheduling approach combining dispatching rules and local search," in *Proceedings of the 21st European Modeling and Simulation Symposium (EMSS '09)*, vol. 2, pp. 27–32, Puerto de la Cruz, Spain, September 2009.
- [24] G. Mušič, T. Löscher, and F. Breiteneker, "Simulation based scheduling applying Petri nets with sequences and priorities," in *Proceedings of the UKSim 10th International Conference on Computer Modelling and Simulation (EUROSIM/UKSim '08)*, pp. 455–460, April 2008.
- [25] M. Zhou and K. Venkatesh, *Modelling, Simulation and Control of Flexible Manufacturing Systems. A Petri Net Approach*, World Scientific, 1999.
- [26] L. Recalde, M. Silva, J. Ezpeleta, and E. Teruel, "Petri nets and manufacturing systems: an examples-driven tour," in *Lectures on Concurrency and Petri Nets: Advances in Petri Nets*, J. Desel, W. Reisig, and G. Rozenberg, Eds., vol. 3098 of *Lecture Notes in Computer Science*, pp. 742–788, Springer, 2004.
- [27] A. Zimmermann, D. Rodriguez, and M. Silva, "A two phase optimization method for Petri net models of manufacturing systems," *Journal of Intelligent Manufacturing*, vol. 12, no. 5-6, pp. 409–420, 2001.
- [28] G. J. Tsinarakis, N. C. Tsoveloudis, and K. P. Valavanis, "Petri net modeling of routing and operation flexibility in production systems," in *Proceedings of the 13th Mediterranean Conference on Control and Automation (MED '05)*, pp. 352–357, June 2005.
- [29] J. I. Latorre, E. Jiménez, and M. Pérez, "Colored Petri nets as a formalism to represent alternative models for a discrete event system," in *Proceedings of the 22nd European Modelling and Simulation Symposium (EMSS '10)*, pp. 247–252, Fez, Morocco, 2010.
- [30] C. Lin and J. He, "Amutual-evaluation genetic algorithm for numerical and routing optimization," *Journal of Applied Mathematics*, vol. 2013, Article ID 214814, 14 pages, 2013.
- [31] G. Balbo and M. Silva, Eds., *Performance Models for Discrete Event Systems with Synchronizations: Formalisms and Analysis Techniques*, Kronos, Saragossa, Spain, 1998.
- [32] R. David and H. Alla, *Discrete, Continuous and Hybrid Petri Nets*, Springer, Berlin, Germany, 2005.
- [33] J. I. Latorre and E. Jiménez, "Simulation-based optimization of discrete event systems with alternative structural configurations using distributed computation and the Petri net paradigm," *Simulation*, vol. 89, no. 11, pp. 1310–1334, 2013.
- [34] J. I. Latorre, E. Jiménez, and M. Pérez, "The optimization problem based on alternatives aggregation Petri nets as models for industrial discrete event systems," *Simulation*, vol. 89, no. 3, pp. 346–361, 2013.
- [35] J. I. Latorre, E. Jiménez, M. Pérez, and J. Blanco, "The problem of designing discrete event systems. A new methodological approach," in *Proceedings of the 21st European Modelling and Simulation Symposium (EMSS '09)*, vol. 2, pp. 40–46, Puerto de la Cruz, Spain, September 2009.