COMPARISON OF INCIDENCE MATRICES TO DETECT COMMON PATTERNS IN PETRI NETS

Juan-Ignacio Latorre-Biel [(a)], Emilio Jiménez-Macías[(b)]

[(a)] Public University of Navarre. Department of Mechanical Engineering, Energetics and Materials.
Campus of Tudela, Spain
[(b)] University of La Rioja. Industrial Engineering Technical School.
Department of Electrical Engineering. Logroño, Spain

[(a)] juanignacio.latorre@unavarra.es, [(b)] emilio.jimenez@unirioja.es

ABSTRACT

Given a discrete event system modeled by an alternatives Petri net system, the identification of common patterns is required in the incidence matrices in order to transform the model into another minimum one necessary to develop a more efficient optimization. Transformations of set of the alternatives Petri nets to be considered are two: aggregation and fusion. Aggregation is used to obtain alternatives Petri nets, and is performed by means of the following operations: identifying of shared subnets on the alternative nets, identification of binding transitions and unshared blocks, and aggregation of the incidence matrices. Fusion is used for obtaining a composed Petri net, and is made by means of the following operations: application of swaps to rows or columns to achieve an optimal configuration, and overlay of matrices. Those types of transformations on Petri nets are based on the equivalence class of the incidence matrices that can be formed by permuting or swapping the rows and the columns. This paper constitutes such a basis, by means of the analysis of this equivalence class.

Keywords: Petri nets, Incidence matrix, equivalence classes

1. INTRODUCTION

The decisions on discrete event systems under design and alternative structural configurations can be addressed by applying a family of formalisms based on Petri nets (Silva, 1993; Alla and David, 2005; Jensen and Kristensen, 2009) that include a set of mutually exclusive entities. This decision-making can be addressed through an optimization process based on simulation of the system model under different valid configurations. The optimization process efficiency depends on the speed with which the simulation is performed, that in the case of a model expressed by the formalism of Petri nets requires the solution of the state equation. Simulation therefore be the more efficient the smaller the system model, and in particular the size of the incidence matrices (Zimmermann et al., 2001; Tsinarakis et al. 2005;Jimenez et al., 2006, 2009; Latorre et al., 2013a).

Given a discrete event system modeled by an alternatives Petri net system, the identification of common patterns is required in the incidence matrices in order to transform this model into another minimum one necessary to develop a more efficient optimization (Berthelot, 1987; Haddad and Pradat-Peyre, 2006). Transformations of set of the alternatives Petri nets to be considered are two: aggregation and fusion (Latorre et al., 2009, 2011a).

a) Aggregation:

The aggregation of alternative Petri nets is used to obtain an alternatives Petri net, and aggregation is performed by means of the following operations: a.1) Identifying shared subnets on alternative Petri Nets (matching columns in various incidence matrices). At this stage it is possible to exchange between two rows and between two columns for optimal arrangement of the elements of each incidence matrix.

a.2) Identification of the binding transitions (columns that do not match with other incidence matrices having some nonzero element in the same row in which a shared subnet of the same incidence matrix has nonzero elements).

a.3) unshared blocks (other columns, ie columns mismatched with other incidence matrices having non-zero elements only, in which the matching columns in various matrices have nulls).

a.4) Aggregation: Building an aggregate incidence matrix as follows:

* The first block in the aggregate matrix is the first incidence matrix assembly.

* For each new incidence matrix of the set, they are added to the aggregate matrix binding their transitions, placing the non-zero elements in the matrix rows correspond to aggregate shared subnets.

* Also unshared blocks are added so that the non-zero elements in rows match corresponding to the nonzero elements and link transitions or new rows inserted in the same way as in the original array.

* The aggregate matrix voids are filled with zeros.

b) Fusion.

Merging alternative Petri nets used for obtaining a composed Petri net is made by means of the following operations.

b.1) Application of swaps to two rows or two columns to achieve an optimal configuration of the elements of each of the incidence matrices.

b.2) Overlay of matrices to obtain the merged matrix, element by element. Each element of the resulting array will be associated with a single value if overlapping elements coincide (this element is called defined parameter) or a set of possible values that will have many elements with different values coming from the original elements (this element is called undefined parameter).

To manage efficiently those types of transformations on Petri nets, some intermediate goals are required, in concrete:

Algorithm 1 (optimization): search for common patterns in a set of matrices for minimizing the size of the aggregate matrix.

Algorithm 2 (multiobjective optimization): minimization of aggregate array size and the number of transitions link (decision variables).

Algorithm 3 (optimization) search for common patterns in a set of matrices to minimize the number of undefined parameters of the resulting array.

Algorithm 4 (multiobjective optimization): minimize the number of undefined parameters of the resulting matrix and the size of the containing sets of possible values for each parameter of the resulting matrix.

In addition, to developing the algorithms mentioned and provide evidence of correct operation (eg statistics) is included within the objectives to determine the computational complexity of the system.

Regarding distributed optimization, given a set of m matrices and a set of processors p, with p<m, the development of the following algorithms is tacked:

Algorithm 5 (optimization): determination of exchange operations pairs of rows and columns needed and the optimal partition of all the m matrices in p sets, to minimize the average size of the aggregate incidence matrices in each class partition and its variance.

Algorithm 6 (optimization): determination of exchange operations pairs of rows and columns needed and the optimal partition of all the m matrices in p sets, to minimize the average size of the incidence matrices resulting from the incidence matrices in each class of the partition and its variance.

Any of those algorithms constitute a goal and an advance in the state of the arte, with immediate applications and being the basis of other interesting issues. And all of them are based on the equivalence class of the equivalent matrices, that is, the equivalence class of the incidence matrices that can be formed by permuting the rows and the columns. This paper constitutes such a basis, by means of the analysis of the equivalence class of the incidence matrices ().

2. EQUIVALENCE CLASSES

2.1. Operations of incidence matrices

A decision problem based on an undefined Discrete Event System (DES) can be stated as an optimization problem based on an undefined Petri net. The performance of the optimization process can be influenced in a dramatic way by the representation considered for the undefined Petri net. Some operations allow transforming an alternative Petri net into another one that has an equivalent state space to the original PN and might be more appropriate for representing an undefined Petri net in an efficient optimization process. This relation of equivalence will guarantee that the equivalent Petri nets have isomorphous reachability graphs and the same set of reachable significant markings (Latorre et al., 2011b, 2013b, 22013c).

The transformation of one alternative Petri net into an equivalent one will be performed by means of the application of certain matrix-based operations to the incidence matrix. In fact, by the application of these operations to the alternative Petri nets it is possible to obtain adequate incidence matrices for their merging into a more compact compound Petri net. This compound Petri net will be equivalent representations of the same undefined PN in decision problems.

As a consequence, the equivalent Petri net of a certain alternative PN verifies that its incidence matrix can be obtained from the transformation of any other from the same set by means of the application of certain matrix-based operations. Any alternative Petri net of a well-constructed set define an equivalence class. This equivalence class is created by the application of the different feasible sequences of matrix-based operations to the incidence matrix of the alternative Petri net that creates it. From this alternative Petri net the equivalent Petri nets that can substitute a given alternative Petri net can be taken.

Definition 1. Operation of swapping two rows of a matrix.

The operation of swapping two rows of a matrix is defined as the following function:

swapr: $\mathbf{M}_{m\times n} \times \{1, 2, \ldots, m\} \times \{1, 2, \ldots, m\} \to \mathbf{M}_{m\times n}$

$(\mathbf{A}, i, j) \# \mathbf{B} \in \mathbf{M}_{m\times n}$

where, $\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ \cdots & \cdots & \cdots & \cdots \\ a_{i1} & a_{i2} & \cdots & a_{in} \\ \cdots & \cdots & \cdots & \cdots \\ a_{j1} & a_{j2} & \cdots & a_{jn} \\ \cdots & \cdots & \cdots & \cdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix} \in \mathbf{M}_{m\times n}$,

$$\mathbf{B} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ \cdots & \cdots & \cdots & \cdots \\ a_{j1} & a_{j2} & \cdots & a_{jn} \\ \cdots & \cdots & \cdots & \cdots \\ a_{i1} & a_{i2} & \cdots & a_{in} \\ \cdots & \cdots & \cdots & \cdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix} \in \mathbf{M}_{m\times n}.$$

□

In other words, **definition 1**, describes the swapping of the *i*-th and *j*-th rows in a matrix **A**. This operation is denoted by swapr(**A**, *i*, *j*).

Definition 2 . Operation of swapping two columns of a matrix.
The operation of swapping two columns of a matrix is defined as the following function:
swapc: $\mathbf{M}_{m\times n} \times \{1, 2, \ldots, n\} \times \{1, 2, \ldots, n\} \rightarrow \mathbf{M}_{m\times n}$
$(\mathbf{A}, i, j) \# \mathbf{B} \in \mathbf{M}_{m\times n}$
where,

$$\mathbf{A} = \begin{pmatrix} a_{11} & \cdots & a_{1i} & \cdots & a_{1j} & \cdots & a_{1n} \\ a_{21} & \cdots & a_{2i} & \cdots & a_{2j} & \cdots & a_{2n} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{m1} & \cdots & a_{mi} & \cdots & a_{mj} & \cdots & a_{mn} \end{pmatrix} \in \mathbf{M}_{m\times n},$$

and

$$\mathbf{B} = \begin{pmatrix} a_{11} & \cdots & a_{1j} & \cdots & a_{1i} & \cdots & a_{1n} \\ a_{21} & \cdots & a_{2j} & \cdots & a_{2i} & \cdots & a_{2n} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{m1} & \cdots & a_{mj} & \cdots & a_{mi} & \cdots & a_{mn} \end{pmatrix} \in \mathbf{M}_{m\times n}$$

□

In other words, **definition 2**, describes the swapping of the columns i and j in matrix **A**, which is denoted by swapc(**A**, i, j)

Remark 1. The state equation of a Petri net requires representing the characteristic vector that summarizes the information contained in the sequence of transitions fired. The characteristic vector (also called firing count vector) contains elements that are different to zero in the positions that correspond to the transitions fired. If an operation swapc is applied to an incidence matrix and the state equation is represented, the characteristic vector should be modified according to this same swapc operation.

Definition 3. Operation of adding a row of zeros to a matrix.
The operation of adding a row of zeros to a matrix is defined as the following function:
addr: $\mathbf{M}_{m\times n} \rightarrow \mathbf{M}_{(m+1)\times n}$
A # **B**, such that

Given $\mathbf{A} = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \cdots & \cdots & \cdots \\ a_{m1} & \cdots & a_{mn} \end{pmatrix} \in \mathbf{M}_{m\times n}$

$\Rightarrow$ addr(**A**) = $\mathbf{B} = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \cdots & \cdots & \cdots \\ a_{m1} & \cdots & a_{mn} \\ 0 & \ldots & 0 \end{pmatrix} \in \mathbf{M}_{(m+1)\times n}$

□

The operation described in **definition 3** is denoted by addr(**A**) and adds a row of zeros to the matrix **A**.

Remark 2. The operation addr applied to the incidence matrix of a Petri net implies the addition of a new place with a particular property: every input and output arc has weight zero. In other words, this new place is an isolated node of the Petri net.

The marking of the Petri net that results from the application of this operation should include the marking of the new place, which will occupy the last position of the vector. However, being isolated, the place cannot experience any variation of its initial marking in the evolution of the Petri net. Furthermore, the marking of other places will not be influenced by the added place, hence the marking of the new Petri net, excluding the added place, will be the same to the original one. If the new place is considered in this comparison it is possible to say that the significant marking is the same in both Petri nets; hence the graphs of reachable markings are isomorphous.

Definition 4. Operation of removing a row of zeros of a matrix.
The operation of removing a row of zeros of a matrix is defined as the following function:
removr: $S \rightarrow \mathbf{M}_{(m-1)\times n}$
A # **B**, such that
$S = \{\mathbf{A} \in \mathbf{M}_{m\times n} \mid a_{m*} = (0\ 0\ \ldots\ 0)\}$, in other words, S is the set of matrices whose m-th (last) row is a row of zeros.

Given $\mathbf{A} = \begin{pmatrix} a_{1,1} & \cdots & a_{1,n} \\ \cdots & \cdots & \cdots \\ a_{m-1,1} & \cdots & a_{m-1,n} \\ 0 & \ldots & 0 \end{pmatrix} \in \mathbf{M}_{m\times n} \Rightarrow$

removr(**A**) = $\mathbf{B} = \begin{pmatrix} a_{1,1} & \cdots & a_{1,n} \\ \cdots & \cdots & \cdots \\ a_{m-1,1} & \cdots & a_{m-1,n} \end{pmatrix} \in \mathbf{M}_{(m-1)\times n}$

□

The operation described in **definition 4** is denoted by removr(**A**) and removes the last row of a matrix **A**, which should contain only zeros.

Definition 5. Operation of adding a column of zeros to a matrix.
The operation of adding a column of zeros to a matrix is defined as the following function:

addc: $\mathbf{M}_{m\times n} \rightarrow \mathbf{M}_{m\times(n+1)}$
A # **B**, such that

Given $\mathbf{A} = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \cdots & \cdots & \cdots \\ a_{m1} & \cdots & a_{mn} \end{pmatrix} \in \mathbf{M}_{m\times n} \Rightarrow$

addc(**A**) = $\mathbf{B} = \begin{pmatrix} a_{11} & \cdots & a_{1n} & 0 \\ \cdots & \cdots & \cdots & \cdots \\ a_{m1} & \cdots & a_{mn} & 0 \end{pmatrix} \in \mathbf{M}_{m\times(n+1)}$

□

The operation described in **5** is denoted by addc(**A**) and adds a column of zeros to a matrix **A**.

Remark 3. The operation addc applied to the incidence matrix of a Petri net implies the addition of a new transition with a particular property: every one of its input and output arcs has weight zero. In other words, this new transition is an isolated transition of the Petri net. Moreover, the added transition will be associated to the last column of the incidence matrix.

Any characteristic vector associated to R should be modified before being associated as well to the Petri net R' that results from the application of the operation addc to its incidence matrix. This modification of the characteristic vector consists of the addition of a zero as the new last element. Thanks to this modification the size of the vector will fit with the one of the incidence matrix **B** in the state equation.

Definition 6. Operation of removing a column of zeros of a matrix.
The operation of removing a column of zeros of a matrix is defined as the following function:

removc: $S \rightarrow \mathbf{M}_{(m\text{-}1)\times n}$
 A # **B**, such that

$S = \{\mathbf{A} \in \mathbf{M}_{m\times n} \mid a_{*n} = [0\ 0\ \ldots\ 0]^{\mathrm{T}}\}$, in other words, S is the set of matrices whose nth (last) column is a column of zeros.

Given $\mathbf{A} = \begin{pmatrix} a_{1,1} & \cdots & a_{1,(n-1)} & 0 \\ \cdots & \cdots & \cdots & \cdots \\ a_{m,n} & \cdots & a_{m,(n-1)} & 0 \end{pmatrix} \in \mathbf{M}_{m\times n} \Rightarrow$

removc(**A**) = $\mathbf{B} = \begin{pmatrix} a_{1,1} & \cdots & a_{1,(n-1)} \\ \cdots & \cdots & \cdots \\ a_{m,1} & \cdots & a_{m,(n-1)} \end{pmatrix} \in \mathbf{M}_{m\times(n\text{-}1)}$

□

The operation described in **definition 6** is denoted by removc(**A**) and removes the last columns of a matrix **A**, which should contain only zeros.

It can be proven that none of the matrix-based operations defined in this section modify the form of the reachability graph of the Petri net, when thay are applied to its incidence matrix. Furthermore, the significant markings are the same.

Proposition 1.
Let R be a Petri net with an incidence matrix given by **A** $\in \mathbf{M}_{m\times n}$ and let swapr(**A**, i, j) = **B** $\in \mathbf{M}_{m\times n}$. The Petri net associated to **B** is R'. The initial markings of R and R' are respectively $\mathbf{m}_0$ and $\mathbf{m}_0$'.

$$\mathrm{rg}(R, \mathbf{m}_0) = \mathrm{rg}(R', \mathbf{m}_0')$$

□

Proposition 2.
Let R be a Petri net with an incidence matrix given by **A** $\in \mathbf{M}_{m\times n}$. and let swapc(**A**, i, j) = **B** $\in \mathbf{M}_{m\times n}$. The Petri net associated to **B** is R'. The initial markings of R and R' are respectively $\mathbf{m}_0$ and $\mathbf{m}_0$'.

$$\mathrm{rg}(R, \mathbf{m}_0) = \mathrm{rg}(R', \mathbf{m}_0')$$

□

Proposition 3.
Let R be a Petri net with an incidence matrix given by **A** $\in \mathbf{M}_{m\times n}$ and let addr(**A**) = **B** $\in \mathbf{M}_{(m+1)\times n}$. The Petri net associated to **B** is R'. The initial markings of R and R' are respectively $\mathbf{m}_0$ and $\mathbf{m}_0$'.

rg(R, $\mathbf{m}_0$) = rg(R', $\mathbf{m}_0$') for the marking of the connected places.

□

Proposition 4.
Let R be a Petri net with an incidence matrix given by **A** $\in \mathbf{M}_{m\times n}$ and let removr(**A**) = **B** $\in \mathbf{M}_{(m\text{-}1)\times n}$. The Petri net associated to **B** is R'. The initial markings of R and R' are respectively $\mathbf{m}_0$ and $\mathbf{m}_0$'.

rg(R, $\mathbf{m}_0$) = rg(R', $\mathbf{m}_0$') for the marking of the connected places.

□

Remark 4. What **proposition 3** and **proposition 4** mean in fact is that the graphs of reachable markings are isomorphous. They are only the same for the significant marking or more specifically for the marking of the connected places. Furthermore, the isolated places are associated to constant markings and they do not influence the evolution of the Petri net (the valid sequences of transition firing and the markings of the connected places). For this reason this relation between the graphs of reachable markings of the original Petri nets and the ones resulting from the application of the operations addr and removr is approximated with a high degree of reliability and usefulness in the applications for decision problems as being the same: rg(R, $\mathbf{m}_0$) = rg(R', $\mathbf{m}_0$').

Proposition 5.
Let R be a Petri net with an incidence matrix given by **A** $\in \mathbf{M}_{m\times n}$ and let addc(**A**) = **B** $\in \mathbf{M}_{m\times(n+1)}$. The Petri net associated to **B**, obtained from R is R'. The initial markings of R and R' are the same, in other words $\mathbf{m}_0$ = $\mathbf{m}_0$'.

$$\mathrm{rg}(R, \mathbf{m}_0) = \mathrm{rg}(R', \mathbf{m}_0')$$

□

Proposition 6.
Let R be a Petri net with an incidence matrix given by **A** $\in \mathbf{M}_{m\times n}$ such that $a_{*j} = 0$ (the elements of the last column are zeros) and let removc(**A**) = **B** $\in \mathbf{M}_{m\times(n\text{-}1)}$. The Petri net associated to **B** is R'. The initial markings of R and R' are the same ($\mathbf{m}_0$ = $\mathbf{m}_0$').

$$\mathrm{rg}(R, \mathbf{m}_0) = \mathrm{rg}(R', \mathbf{m}_0')$$

□

3. WELL-CONSTRUCTED SETS OF ALTERNATIVE PN AND EQUIVALENCE CLASSES

The matrix-based operations described in the previous section may be used to substitute one or several alternative Petri nets to find more convenient representations for solving the optimization problem in a more efficient way.

In order to proceed as described in the previous paragraph it is interesting to classify the sets of alternative Petri nets into two categories, which will be called the well-constructed sets and the redundant sets. It is possible to describe a well-constructed set as the one containing alternative Petri nets able to define equivalence classes from the equivalence relation given by the application of the matrix-based operations. In this case, the alternative Petri nets of the original set cannot be transformed into another Petri net of the same set, since they belong to different equivalence classes, which verify the property of being disjoint sets. This property of exclusiveness is associated to the idea of set of exclusive entities, which is the signature of the decision problems where there are structural alternatives as feasible solutions.

The name "redundant" arises from the fact that at least one alternative Petri net can be transformed into another one from the same. subsequently both of them belong to the same equivalence class. As a consequence, the couple of PN is related to the same solution of the decision problem.

3.1. Well-constructed sets of alternative PN and equivalence classes: definitions

Definition 7. Set of equivalence operations.
The set of equivalence operations is S_{OP} = { swapr, swapc, addr, removr, addc, removc } set of all matrix-based operations defined previously in this chapter.

□

Definition 8. Feasible sequence of operations.
A feasible sequence of operations is a finite set of the form

$$S_{secop} = \{ \text{op}_1, \text{op}_2, \ldots, \text{op}_{nop} \mid \text{op}_i \in S_{OP} ,\ 1 \le i \le n_{op} \}$$

□

Definition 9. Set of feasible sequence of operations.
The set of feasible sequence of operations is
$S_{SOP} = \{ S_{secop} \}$

□

Definition 10. Application of a sequence of operations to an incidence matrix.
Let $\mathbf{W}_a$ be the incidence matrix of a Petri net.
Let S_{secop} be a feasible sequence of operations such that $S_{secop} \in S_{SOP}$.
The application of S_{secop} to $\mathbf{W}_a$ is called $S_{secop}(\mathbf{W}_a)$ and is performed in the following way

$$S_{secop}(\mathbf{W}_a) = \text{op}_{nop}(\ldots\text{op}_2(\text{op}_1(\mathbf{W}_a)))$$

The way of applying this sequence of operations is the following: first of all it is calculated the operation $\text{op}_1(\mathbf{W}_a) = \mathbf{W}_a'$, in a second step it is calculated the operation $\text{op}_2(\mathbf{W}_a') = \mathbf{W}_a'' = \text{op}_2(\text{op}_1(\mathbf{W}_a))$ and so on until the last operation op_{nop} is applied.

□

Definition 11. Equivalence relation between two Petri nets.
Let us consider the Petri nets R_a and R_b , whose incidence matrices are $\mathbf{W}_a$ and $\mathbf{W}_b$ respectively.
The equivalence relation ~ is defined in the following way:

$R_a \sim R_b$ iif $\exists\, S_{secop} \in S_{SOP}$ such that $S_{secop}(\mathbf{W}_a) = \mathbf{W}_b$.

□

Definition 12. Equivalence class defined by an alternative Petri net.
Given a set of alternative Petri nets $S_R = \{ R_1, R_{2,\ldots,} R_{nr} \}$, the binary equivalent relation ~ , defined in **definition 6.21**, allows to create an equivalence class for every alternative Petri net, such that

Let $R_i \in S_R$, the equivalence class created by R_i is $[R_i] = \{ R \text{ PN} \mid R_i \sim R \}$.

□

Definition 13. Well-constructed set of alternative Petri nets.
Given a set of alternative Petri nets $S_R = \{ R_1, R_{2,\ldots,} R_{nr} \}$. S_R is said to be well constructed iif $\forall\, R_i , R_j \in S_R$, $i \neq j$,it does not exist any sequence of operations $S_{secop} \in S_{SOP}$ such that $S_{secop}(\mathbf{W}_i) = \mathbf{W}_j$.

□

In other words, for a set of alternative Petri nets to be well constructed set it is a necessary and sufficient condition that none of the Petri nets of the set has an incidence matrix that can be transformed, by means of equivalence operations, into the incidence matrix of another Petri net of the same set.

On the other hand, it has been mentioned before as well that the definition of this category of well-defined sets of alternative Petri nets do not compromise the applicability of the methodologies developed in this thesis to real cases. On the contrary, the correct models developed for undefined DES will not contain isolated places or transitions (which do not contribute to the evolution of the system) or different order of the rows or columns between them (due to the assignment of different names for the same real items modelled by nodes in the PN). Subsequently, this condition do not prevent the representation of most of the different possible real or academic cases that can arise in a decision problem, but it is a guarantee of the correct development of a model of an undefined DES for a decision problem. Furthermore, this small restriction will have important implications in the improvement of the efficiency of the algorithms to solve decision problems in the scope of this thesis.

3.2. Sufficient conditions for a set of alternative Petri nets to be well constructed.

The next topic to be considered is how to check that a set of alternative Petri nets is a well-constructed one and, hence, able to create so many different equivalence classes as the cardinality of the set.

Proposition 7. Sufficient condition to identify a well-constructed set of alternative Petri nets.
Let D be a DES.
Let R^U be an undefined Petri net developed as model for D.

Let $S_R = \{ R_1, R_{2, \dots,} R_{nr}\}$ be a set of alternative Petri nets developed as representation of R^U.

If the following conditions are verified

a) $\forall\ R_i \in S_R \wedge \forall\ p_j \in P_i$ it is verified that p_j is a connected place.
b) $\forall\ R_i \in S_R \wedge \forall\ t_j \in T_i$ it is verified that t_j is a connected place.
c) $\forall\ R_i, R_j \in S_R \wedge \forall\ p_{k1} \in P_i$, $p_{k2} \in P_j$ such that $X(p_k)$ is the item in D modelled by p_{k1} and p_{k2} then $k_1 = k_2$.
d) $\forall\ R_i, R_j \in S_R$ it is verified that $\mathbf{W}(R_i) \neq \mathbf{W}(R_j)$.

Then S_R is a well-constructed set of alternative Petri nets.

□

A sufficient condition for a set of alternative Petri net to have incidence matrices that cannot be transformed one into another by means of “add” or “remov” operations is that there is not any isolated place or transition in any of the alternative Petri nets. As it has been mentioned before, this is a usual case in the application of PN found in the literature so far, since in the definition of PN it is usually imposed the condition of connectivity in every node. A way to detect this situation is to search for rows or columns of zeros.

A sufficient condition for a set of alternative Petri net to have incidence matrices that cannot be transformed one into another by means of “swap” operations is to give in the model the same reference name to the same physical item that is modelled as a place or transition, to associate an alias to every reference name with the same subindex and to compare the incidence matrices element by element. If there is a pair (or more) elements which are different in the incidence matrices of different Petri nets, the set of alternative Petri nets is well constructed.

Another sufficient condition for a set of alternative Petri net to be well constructed is given below.

Proposition 8. Sufficient condition to identify a well-constructed set of alternative Petri nets.
Let $S_R = \{ R_1, R_{2, \dots,} R_{nr}\}$ be a set of alternative Petri nets, where $\mathbf{W}_k$ is the incidence matrix of $R_k \in S_R$, $\mathbf{W}_k \in \mathbf{M}_{mk\times nk}$ and $a^k_{i,j} = \mathbf{W}_k[i,j]$

$$\text{Let } sumt_k = \sum_{\substack{i,j=1}}^{\substack{i=m_k \\ j=n_k}} a^k_{i,j}$$

If $\forall\ R_{k1}, R_{k2} \in S_R$, $sumt_{k1} \neq sumt_{k2} \Rightarrow S_R$ is well constructed

□

As it has been proven, a sufficient condition for a set of alternative Petri nets to be well constructed is that the sums of all the elements of the incidence matrix are different for the different alternative Petri net.

3.3. Redundant set of alternative Petri nets.
It has been explained previously that a set of alternative Petri nets that is not well constructed may lead to the creation of the same equivalence class by several different alternative PN of the set.

The detection of this fact by means of the sufficient conditions described earlier can allow the removal of the redundant alternative Petri nets and hence the simplification of the statement of the decision problem based on a reduced set of alternative PN. A simplification in the representation of an undefined Petri net may lead to a more efficient optimization process based on a model of reduced size.

In case that a redundant set of alternative Petri net is not detected and used to state an optimization problem, the different alternative Petri nets that create the same equivalence class will be treated as if their respective equivalence classes were different. The behaviour and the structure of the Petri nets will not be different from one of these alternative PN and the others that create the same equivalence class. As a consequence, the optimization algorithm might duplicate the computational effort by considering twice the same alternative Petri net.

4. CONCLUSIONS

Given a discrete event system modeled by an alternatives Petri net system, the identification of common patterns is required in the incidence matrices in order to transform the model into another minimum one necessary to develop a more efficient optimization. Transformations of set of the alternatives Petri nets to be considered are two: aggregation and fusion.

This paper has analyzed the equivalence class of the incidence matrices, that is, of matrices that can be obtained by swapping rows and columns. This knowledge, and the basis of the definitions and properties, constitute the starting point to analyze the optimization of several incidence matrices that are wanted to be merged, as it happens with alternatives Petri nets.

ACKNOWLEDGMENTS
This paper has been partially supported by a grant of the University of La Rioja and Banco Santander.

REFERENCES
Berthelot, G., 1987. Transformations and decompositions of nets. In: Brauer, W., Reisig,
David, R., Alla. H., 2005. *Discrete, Continuous and Hybrid Petri nets*, Springer.
Haddad, S. and Pradat-Peyre, J.F., 2006. New Efficient Petri Nets Reductions for Parallel Programs Verification. *Parallel Processing Letters*, pages 101-116, World Scientific Publishing Company.
Jensen, K., Kristensen, L.M., 2009. *Coloured Petri nets. Modelling and Validation of Concurrent Systems*. Springer.

Jimenez, E., Perez, M., Latorre, J.I., 2006. Industrial applications of Petri nets: system modelling and simulation. *Proceedings of International Mediterranean Modelling Multiconference* 2006, pp. 159-164

Jimenez, E., Perez, M., Latorre, J.I., 2009. Modelling and simulation with discrete and continuous PN: semantics and delays. *Proceedings of 21st European Modeling and Simulation Symposium, Vol II*, pp. 14-19

Latorre, J.I., Jimenez, E., Blanco, J., Sáenz-Díez, J.C., 2013a. Integrated methodology for efficient decision support in the Rioja wine production sector. *International Journal of Food Engineering*, (In press).

Latorre, J.I., Jiménez, E., Pérez, M., 2011a. Matrix-based operations and equivalence classes in alternative Petri nets. *Proceedings of the 23rd European Modelling and Simulation Symposium* (EMSS 11). Rome, Italy, pp. 587-592.

Latorre, J.I., Jiménez, E., Pérez, M., 2011b. Petri net transformation for decision making: compound Petri nets to alternatives aggregation Petri nets. *Proceedings of the 23rd European Modelling and Simulation Symposium* (EMSS 11). Rome, Italy, pp. 613-618.

Latorre, J.I., Jimenez, E., Perez, M., 2013b. Simulation-based Optimisation of Discrete Event Systems by Distributed Computation. *Simulation-Transactions of the Society for Modeling and Simulation International*, (In press).

Latorre, J.I., Jimenez, E., Perez, M., 2013c. The optimization problem based on alternatives aggregation Petri nets as models for industrial discrete event systems. *Simulation-Transactions of the Society for Modeling and Simulation International*, 89 (3), 346–361.

Latorre, J.I., Jimenez, E., Perez, M., Blanco, J., 2009. The problem of designing discrete events systems. A new methodological approach. *Proceedings of 21st European Modeling and Simulation Symposium, Vol II*, pp. 40-46

Silva, M., 1993. Introducing Petri nets. In *Practice of Petri Nets in Manufacturing*, Di Cesare, F., (editor), pp. 1-62. Ed. Chapman&Hall.

Tsinarakis, G. J., Tsourveloudis, N. C., and Valavanis, K. P., 2005. Petri Net Modeling of Routing and Operation Flexibility in Production Systems. *Proceedings of the 13th Mediterranean Conference on Control and Automation*, pages 352-357.

Zimmermann, A.; Freiheit, J.; Huck, A. 2001. A Petri net based design engine for manufacturing systems. *International Journal of Production Research*, Vol. 39, No. 2, pages 225-253.