



Production

ISSN: 0103-6513

production@editoracubo.com.br

Associação Brasileira de Engenharia de
Produção
Brasil

Yastami Nakamoto, Francisco; Eigi Miyagi, Paulo; dos Santos Filho, Diolino José
Automatic generation of control solution for resource allocation using Petri net model
Production, vol. 19, núm. 1, enero-abril, 2009, pp. 8-26
Associação Brasileira de Engenharia de Produção
São Paulo, Brasil

Available in: <http://www.redalyc.org/articulo.oa?id=396742035002>

- How to cite
- Complete issue
- More information about this article
- Journal's homepage in redalyc.org

redalyc.org

Scientific Information System

Network of Scientific Journals from Latin America, the Caribbean, Spain and Portugal

Non-profit academic project, developed under the open access initiative

Automatic generation of control solution for resource allocation using Petri net model

Francisco Yastami Nakamoto Centro Universitário Fundação Santo André

Paulo Eigi Miyagi EPUSP

Diolino José dos Santos Filho EPUSP

ABSTRACT

The flexible manufacturing system (FMS) executes multiple processes simultaneously using a limited set of resources. These processes can be blocked permanently due to the sequence of the activities, i.e., the processes sharing a finite set of resources may eventually lead to a deadlock state. The FMS belongs to the class of discrete event systems where it is not possible to determine when an event will occur and the sequence of events becomes undefined. Such systems can be modeled using Petri net, and this work presents a method that solves the deadlock problem considering the indeterminism of FMSs. The goal of this work is to introduce an algorithmic approach, which allows the implementation of a computer program based on FMS functional specification. The proposal is to automate the generation of the control strategy for resource utilization and the rules to avoid the deadlock situation.

KEY WORDS

Flexible manufacturing system, deadlock avoidance, Petri net, production system control, resource allocation.

Geração automática da solução de controle para alocação de recursos utilizando redes de Petri

RESUMO

O sistema flexível de manufatura (FMS flexible manufacturing system) executa múltiplos processos simultaneamente que utilizam um conjunto limitado de recursos. Estes processos podem ficar permanentemente bloqueados em função da sequência de atividades, isto é, o compartilhamento de um conjunto finito de recursos pode levar o sistema a um estado de autotravamento. O FMS pode ser visto como pertencente a uma classe de sistema a eventos discretos, onde não é possível determinar quando um evento ocorre e com isto a sequência de eventos fica indefinida. Este tipo de sistema pode ser modelado através de rede de Petri e assim explora-se esta técnica para desenvolver uma solução para o problema de autotravamento considerando o indeterminismo do FMS. O objetivo é introduzir uma abordagem algorítmica para a implementação de uma ferramenta computacional baseada na especificação funcional do FMS e que automatize a geração de uma estratégia de controle para a utilização de recursos e de regras para evitar o autotravamento.

PALAVRAS-CHAVE

Sistema flexível de produção, autotravamento de processos, rede de Petri, controle de sistemas de produção, alocação de recursos.

1. INTRODUCTION

Product life cycles in the manufacturing systems are changing constantly and the variability of these products has been increasing in order to meet the needs of the market. Therefore, it is necessary to use a system architecture that allows more flexibility. The flexibility concept should guarantee a level of adequate productivity to meet the needs of the market with a better utilization of resources. Consequently, this situation increases the complexity of the control system.

The Flexible Manufacturing System (FMS) is characterized by the simultaneous execution of several processes, sharing the same set of resources. The FMS belongs to the Discrete Events Dynamic Systems (DEDS) class (HO; CAO, 1991; CASSANDRAS, 1993; MIYAGI, 1996). The main characteristic of the DEDS is the dynamic behavior, which is based on the occurrence of instantaneous events that cause an abrupt transition from one discrete state to another. Moreover, it is possible to verify the occurrence of indeterminism, conflict and parallelism of the events.

The control of only one process in FMS is relatively simple since it is just to guarantee the sequencing of activities or stages of the process. However, the simultaneous execution of multiple processes with intense sharing of the finite set of resources may eventually result in a permanent blockage of the system. Such situation is called deadlock or deadlock state (VISWANADHAM et al., 1990; BANASZAK; KROGH, 1991; CHO, 1993; KUMARAN et al., 1994).

This phenomenon occurs when the flow of processes is permanently interrupted and/or when the operations of the processes can no longer be executed. There are three basic approaches to deal with the deadlock problems (FANTI et al., 2000; SANTOS FILHO, 2000): (i) *deadlock detection and recovery*, (ii) *deadlock avoidance* and (iii) *deadlock prevention*.

A brief analysis the approaches developed in the last ten years concerning the deadlock problems in FMS context basically introduces two research lines:

- The identification of a global process and development of the respective control system (*deadlock prevention*);
- The identification of undesirable states and development of the respective control system based on restrictions (*deadlock avoidance* and *deadlock detection and recovery*), i.e., the system does not allow an evolution to the undesirable states.

In a FMS, each process executes a pre-defined sequence of activities and executes each activity using a specific

resource. When two or more processes compete for a limited number of resources, the system can evolve into a deadlock situation.

The occurrence of deadlock in FMS is due to the presence of the following conditions (COFFMAN et al., 1971; BANASZAK; KROGH, 1990; CHO, 1993; FANTI et al., 1997; SANTOS FILHO, 2000): (i) Mutual exclusion, (ii) Await (hold and wait), (iii) No preemption and (iv) Circular wait condition.

The deadlock will occur if all the previous conditions are true at same time. Thus, the system will not evolve for the deadlock situation if at least one of the conditions above is never satisfied. In a FMS, the first three conditions can always occur, that is, two or more processes cannot use the same resource simultaneously and only the process that allocates a resource can release it. Consequently, after the conclusion of the activity of the process, the retention of resource occurs while the process awaits the release of resources held by another process in the same set. The occurrence of a cyclic chain of processes in this term constitutes the circular wait condition. Therefore, the circular wait condition remains as the condition that can be controlled.

The Flexible Manufacturing System (FMS) is characterized by the simultaneous execution of several processes, sharing the same set of resources.

This work aims to present an algorithmic approach based on the deadlock avoidance method for FMS proposed by Santos Filho (SANTOS FILHO, 2000) that allows the computational implementation to generate the resource control algorithm for FMS automatically. The resource control algorithm supervises the processes regarding the resources allocation and it accomplishes decisions in real time to avoid the deadlock situation.

2. RELATED WORK

Viswanadham (VISWANADHAM et al., 1990) proposed the implementation of deadlock prevention by using the reachability graph on Petri net models. However, due to the impossibility of identification of all reachable states, the author proposed the algorithm for a real-time controller using the deadlock avoidance method in Petri net-based model.

Interestingly, the approach adopted by Ezpeleta (EZPELETA et al., 1995) consists of the application of the

deadlock prevent method with an ordinary and conservative Petri net model called Systems of Simple Sequential Processes with Resources (S3PR). The authors analyze the structural element in the net called siphon (PETERSON, 1981; REISIG, 1985; MURATA, 1989). Basically, an algorithm identifies the siphons on the Petri net model and inserts a place to restrict and control the evolution of the marks in the siphon. Moreover, Ezpeleta (EZPELETA et al., 2002) proposed an extension of Banker's algorithm (CORMEN et al., 2003) applied in a S3PR model with flexible routing and multiple instances of different kinds of resource. Here we observe the increasing demand for more and more flexible systems. Accordingly, Ezpeleta (Ezpeleta and Recalde, 2004) presented the approach to the deadlock problem in non-sequential resource allocation systems (NS-RAS). This approach uses the deadlock prevention and avoidance concept based on partial reachability graph on Petri net models with an adaptation of the Banker's algorithm in flexible part routing presented in Ezpeleta (EZPELETA et al., 2002). In both of works presented by Ezpeleta (EZPELETA et al., 2002; EZPELETA; RECALDE, 2004) we can verify a high degree of difficulty to determine the siphons in the Petri net model.

When two or more processes compete for a limited number of resources, the system can evolve into a deadlock situation.

Li (LI; ZHOU, 2003; LI; ZHOU, 2004) proposed the concept of elementary siphons in S3PR. The siphon control is applied adding a control place for each elementary siphon to make sure that it is always marked. This approach reduces the number of control places because it is not necessary to add control places in the redundant siphons.

Using another approach based on deadlock prevent method, Banaszak (BANASZAK, 1997) investigated the deadlock problem in a class of FMS called sequential repetitive production systems. The author proposed a distributed buffer control policy that restricts the process entrance in a buffer of a local machine in a specific sequence. Basically, the rules identify the critical situation of resource allocation and restrict the process entrance according to the capacity of resources, transport timing and resource utilization. Moreover, Banaszak (BANASZAK; POLAK, 2002) presented a study of relationship between system resource capacities and an initial state with priority rules allocation in sequential repetitive production systems.

Basically, the control policy verifies the relation between initial state and the total capacity of the buffer and establishes the restriction of the entrance of the process in a buffer according to this verification.

The applications of the deadlock prevent method are interesting from the point of view of small size FMS. However, they are not feasible for real systems due to the quantity of variables, which can result in an explosion of reachable state of the system.

On the other hand, the deadlock avoidance method is characterized by an approach to the deadlock problems based on restriction policy. Banaszak (BANASZAK; KROGH, 1990) proposed a modeling of FMS using the Petri net model and the application of an algorithm to avoid the deadlock problems called deadlock avoidance algorithm (DAA). The DAA allows the evolution of the marks in the net according to the control policy.

Fanti (FANTI et al., 1997) proposed a control strategy to be applied to the resources control using the graph-theoretical framework. The authors introduce two types of digraphs, named working procedure digraph and transition digraph. Moreover, Fanti (FANTI et al., 2000) presented the connections between the graph-theoretical approach and Petri net models to deal with the deadlock problems in FMS with deadlock avoidance method. Hsieh (HSIEH, 2000) proposed a framework for generating a deadlock avoidance control synthesis algorithm by using the controlled Petri net model, called controlled assembly Petri net (CAPN) model. The algorithm analyzes the liveness conditions, vulnerability and reconfigurability of CAPN for dealing with the addition and removal of processes, as well as the identification of the possible fragile part in the manufacturing systems to provide robustness.

Santos Filho (SANTOS FILHO, 2000) proposed the deadlock avoidance method for FMS based on the restriction model by division of the control system into two levels of control: (i) control of processes, and (ii) control of resources. Both levels are complementary and have distinct semantics. Basically, the algorithm monitors the system regarding the occurrence of the previous deadlock situation. Once the previous state is identified, the algorithm accomplishes actions to avoid the evolution of the system for this undesirable state.

More recently, Fanti (FANTI, 2004) proposed the application of the deadlock avoidance method using a hierarchical structure of control system of FMS with two levels using the colored timed Petri net model. The first level determines the restriction policies based on graph-

theoretical framework (FANTI et al., 1997; 2000) and the second level is responsible for the management of the flows of the processes. The division approach has been adopted by Wu e Zhou (WU; ZHOU, 2005; 2007). The authors applied the deadlock avoidance method for the solution of the deadlock problem in automated guided vehicles systems using the colored Petri net model.

The division approach of the control allows an improvement for the treatment of the different semantics involved in the control system of the FMS, that is, the sequence control of activities of the processes and the resources allocation control. Consequently, this approach facilitates the development of the control system.

As presented previously, the control system of FMS must adapt to the changes in the production environment quickly, according to market demand. We can note that the complexity is inherent to the FMS. Thus, for the systems addressed in this work, the deadlock avoidance method is the most indicated.

3. CONTROL OF FLEXIBLE MANUFACTURING SYSTEMS

The control system accomplishes previously established functions to reach a pre-defined objective (MIYAGI, 1996). In FMS context, such functions involve the execution of activities of the processes.

Some important aspects must be considered for the modeling of the control system: (i) the complexity of FMS, (ii) the architecture of the control system, and (iii) the modeling and analysis tools. The concept of complexity can be defined in many ways and in different contexts, i.e., there is no general definition for every situation. The term “complexity” comes from the Latin ‘complexus’, which means interlaced or twisted together. According to Edmonds (EDMONDS, 1995), the complexity is the property of representation in which the global behavior is hard to be formulated even when all information about the components and their interrelations are provided. In the FMS context, the complexity is a property that the global model of the system may have (SANTOS FILHO, 2000). In other words, complexity can arise when a certain behavior is imposed on FMS.

The imposition of a behavior for a single process is relatively simple, it is enough to guarantee the predetermined sequence of activities of the processes. However, in the global context, it is not possible to

determine the behavior of the system based on purely sequential manner of activities.

The complexity of the global process in FMS is due to the occurrence of two levels of indeterminism (SANTOS FILHO, 2000):

- Indeterminism regarding time: It is not possible to determine when a certain event is going to occur;
- Indeterminism regarding the sequence of events: It is not possible to determine which event precedes another. That situation is due to the execution of simultaneous processes sharing the same set of resources. Consequently, it is not possible to prescribe the system behavior as a whole in this context.

Both levels of indeterminism are increased according to the number of processes, number of resources and other variables of the manufacturing systems. The solution adopted in this work encompasses four fundamental aspects.

The first one is based on restriction model (SANTOS FILHO, 2000). Restriction model is considered because it is impossible to determine all reachable states of the system for the development of a global control.

The second aspect is related to the human element, i.e., the approach adopted in this work is the anthropocentric control systems (ITO, 1991); in other words, the human element is considered a fundamental component of the system. The anthropocentric systems incorporate new technologies in which the human element retains the knowledge and participates in requisite specification, development, implementation and monitoring. Whenever necessary, the human element executes actions that interfere in the dynamics of the system.

The resource control algorithm supervises the processes regarding the resources allocation and it accomplishes decisions in real time to avoid the deadlock situation.

The third aspect refers to the division of the control system in two levels of control (SANTOS FILHO, 2000):

- Control of Processes (CP): responsible for sequencing the activities of the processes, allocating available resources;
- Control of Resources (CR): responsible for managing the utilization of resources by processes.

Figure 1 presents the architecture of system control proposed by Santos Filho (SANTOS FILHO, 2000). The

operator monitors the CP through the monitor device and the operator will be able to interfere in the CP through the command device.

The CP must execute the sequence of activities of the process. The execution of the activities requires the allocation of the resources, thus, the CP requests the utilization of the resource for the CR. The CR can view of the global utilization of resource. Thus, according to the policy allocation

The division approach of the control allows an improvement for the treatment of the different semantics involved in the control system of the FMS.

(restriction policies or production rules), the CR will be able to authorize the request or not. The CP accomplishes the control (control object) through the detection device and actuator devices.

The fourth aspect is the automatic generation of the CR. As presented previously, the control system of FMS must adapt to the changes in the production environment quickly according to market demand. Thus, we observe the

importance of generating the control algorithm quickly form.

4. MODELING TOOLS

Considering the four aspects introduced previously, the adequate choice of the modeling tools can considerably influence the modeling process of the control systems of FMS. In this work, the modeling tools adopted for generation of the control policy rules and the control algorithm of CR are the Resource Allocation Graph (RAG) and Enhanced Mark Flow Graph (E-MFG).

4.1 Resource allocation graph

The RAG model is used to determine the circular wait condition (CWL). Basically, the RAG is an unmarked graph (CORMEN et al., 2003) which represents the resource allocation during the production processes. The graph consists of:

- Nodes, that represent resources used in the process;
- Arcs that represents the stages of the processes.

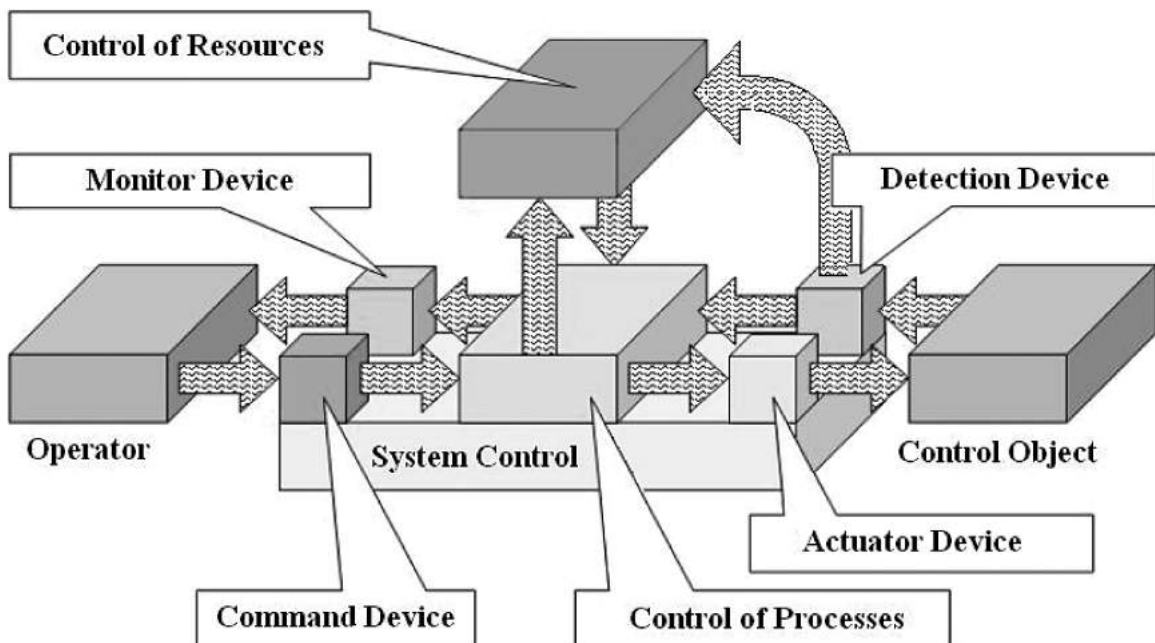


Figure 1: The architecture of system control with two levels of control

Fonte: Santos Filho, 2000.

The RAG is a set $G = (R, A)$, where R is a set of nodes (resources) $R = (R_1, R_2, R_3, \dots, R_n)$ and A is a set of arcs (process stages) $A = (a_{11}, a_{12}, \dots, a_{ij})$ where $a_{ij} = (R_i, R_j)$. The RAG defines an adjacency matrix $G[n \times n]$, where n represents the nodes.

For example, consider a process A that uses sequentially the resources $R1$ and $R2$. The A_{IN} represents the entrance station (loading station) and the A_{OUT} represents the exit station (unloading station). Figure 2 illustrates the RAG model and the matrix representation form (Figure 3) of process A .

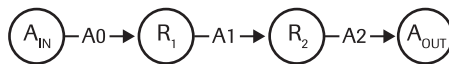


Figure 2: Example of RAG of process A.

$$G = \begin{matrix} & \begin{matrix} R1 \\ R2 \\ A_{IN} \\ A_{OUT} \end{matrix} \\ \begin{matrix} R1 \\ R2 \\ A_{IN} \\ A_{OUT} \end{matrix} & \begin{bmatrix} 0 & A1 & 0 & 0 \\ 0 & 0 & 0 & A2 \\ A0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \end{matrix}$$

Figure 3: Adjacency matrix of process A.

4.2 Enhanced mark flow graph

The Enhanced Mark Flow Graph (E-MFG) was proposed by Santos Filho (SANTOS FILHO, 2000). The E-MFG is an extension of Mark Flow Graph (MFG) deduced from the Petri net model proposed by Miyagi (MIYAGI, 1996).

Considering the functional behavior of the system and the accomplishment of control, the MFG allows the structural representation of complex sequential systems. Therefore, the system can easily be interpreted by observing the model. According to the Figure 4, the basic elements of MFG are: box, transition, output signal arc and external element, inhibitor arc, enabling arc and oriented arc.

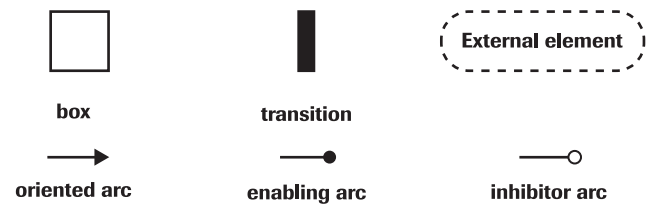


Figure 4: Basic elements of MFG

Fonte: Miyagi, 1996.

The modeling of the FMS using MFG model becomes limited due to the complexity of system. Consequently,

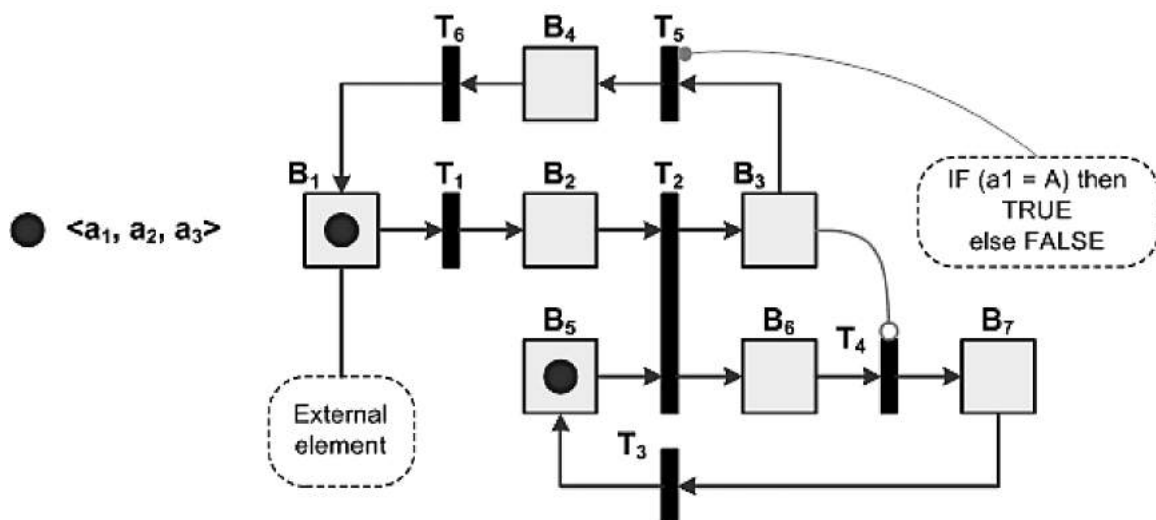


Figure 5: Example of E-MFG.

the E-MFG was introduced to overcome the limitation of MFG (SANTOS FILHO, 2000). The marks in the E-MFG have attributes, which allow the model a larger degree of abstraction. For example, in Figure 5 the box B_1 has a mark with three attributes $\langle A, 2, 4 \rangle$ and the box B_5 has an another mark with three attributes $\langle A, 2, 4 \rangle$.

The fire of the transition T_1 allows the evolution of the mark in the box B_1 to the box B_2 , which enables the transition T_2 . The fire of the transition T_2 allows the evolution of the mark in the box B_3 and the box B_6 . The mark in B_3 inhibits the transition T_4 through the inhibitor arc.

The external element is connected to the transition T_5 by an enabling arc. In this example, the rule associated to the external element is true. Therefore, the transition T_5 allows the evolution of the mark in the box B_4 . The mark

Whenever necessary, the human element executes actions that interfere in the dynamics of the system.

in B_1 allows the signal to the external element through the output signal arc.

5. SYNTHESIS OF THE METHOD TO DESIGN RESOURCE ALLOCATION CONTROL

The generation of the CR depends on the information about the set of available resources of the FMS and the sequences of resource allocation by the processes. The control policy rules are obtained by following the sequence of steps:

- I. Create the individual RAG for each process;
- II. Create the global RAG through of the fusion common nodes;
- III. Determine the CWL in the global RAG;
- IV. Deduce the control policy rules of resources allocation based on "IF... THEN..." model from the set of CWL.

The determination method of the CWL proposed by Santos Filho (SANTOS FILHO, 2000) was the sequence of adjacency matrix multiplication of RAG model. After each matrix multiplication, the CWL are presented in the main diagonal of the result matrix.

For example, consider a set of resources $R = \{R_1, R_2, R_3\}$ and a set of processes $P = \{A, B, C\}$ with the following resources utilization sequence presented in Figure 6.

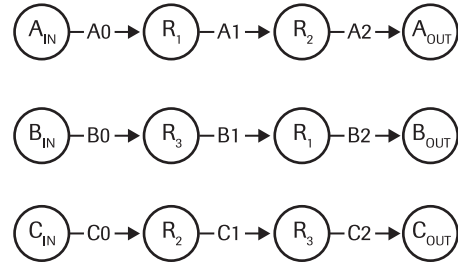


Figure 6: Adjacency matrix of process A.

The fusion of the common nodes of the individual RAG from each process generates another RAG model called Global RAG model. Figure 7 presents the Global RAG model of previous example and the respective CWL is formed by the set of resources R_1, R_2 and R_3 .

The control policy rules must avoid the deadlock situation. The previous deadlock situation is characterized when only one resource is available in the CWL. Under this condition, it is necessary to identify the entrance of the process that causes the deadlock situation.

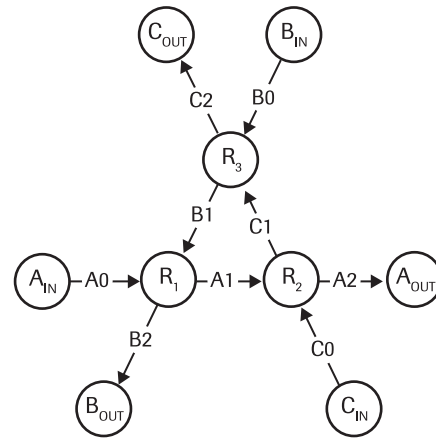


Figure 7: Example of Global RAG and the CWL.

Considering the example in Figure 7, we identify three previous deadlock situations in CWL:

- **IF** (R_1 is used by process A) **AND** (R_2 is used by process C) **THEN** (avoid the entrance of process B in the CWL);

- IF (R₁ is used by process A) AND (R₃ is used by process B)
THEN (avoid the entrance of process C in the CWL);
- IF (R₂ is used by process C) AND (R₃ is used by process B)
THEN (avoid the entrance of process A in the CWL).

Therefore, one rule is determined for each resource in the CWL that has an entrance of process. Behind

determine the control policy rules; the control algorithm of CR is obtained by following the sequence of steps:

- Create the individual E-MFG from individual RAG of the processes (Figure 8). This formalism was introduced by Santos Filho (SANTOS FILHO, 2000). Once that RAG is a transition graph that represents a finite auto-

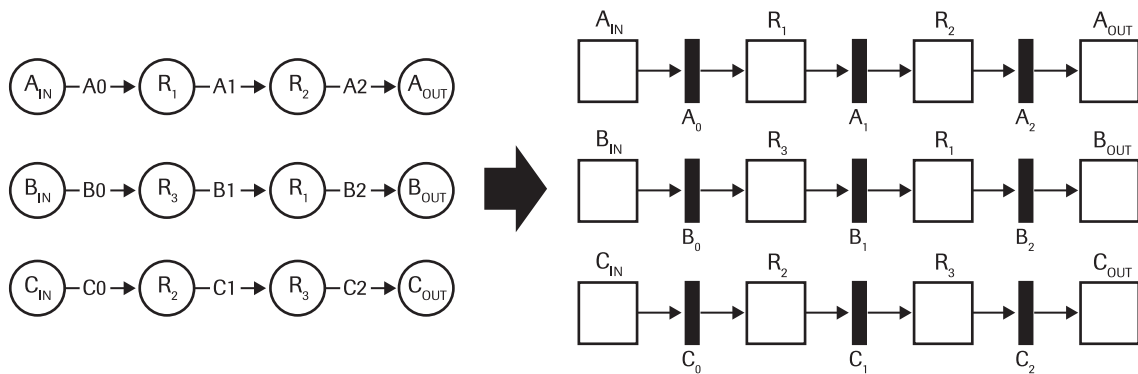


Figure 8: Individual E-MFG models from the RAG models.

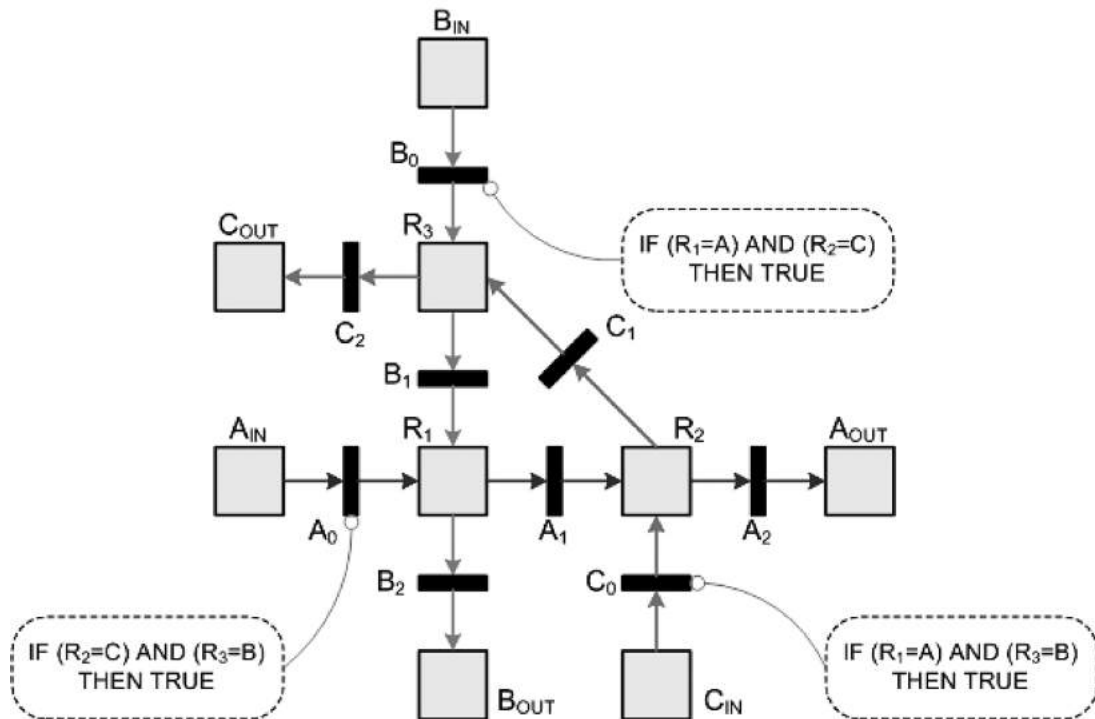


Figure 9: Global E-MFG with control policy rules.

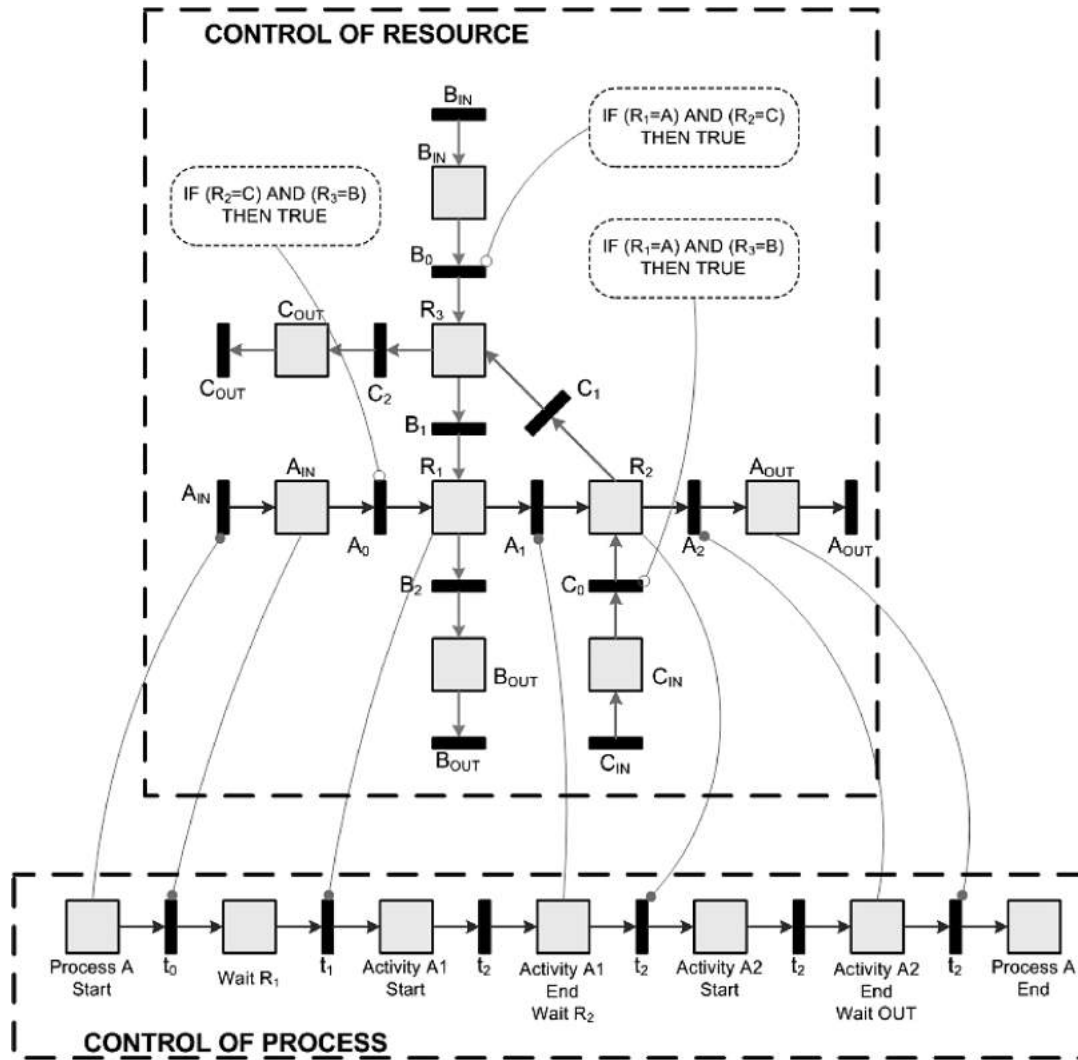


Figure 10: Communication of CR and CP.

mata (REISIG, 1985), It is possible mapping the graph into a net (CASSANDRAS, 1993);

- II. Map the global E-MFG from the individual E-MFG through of the fusion common boxes (SANTOS FILHO, 2000) (Figure 8);
- III. Associate the control policy rules to the transitions in global E-MFG (Figure 9). The E-MFG is represented by incidence matrix (DROZDEK, 2002; CORMEN et al., 2003; ZIVIANI, 2004).

The communication of the CP and CR is accomplished with the enabling arcs in the transitions of E-MFG Models.

The CP realizes the control of sequence of the activities of the process. Figure 10 presents the communication of CR and the control of process A in CP.

The validation of the models CR and CP were accomplished by Santos Filho (SANTOS FILHO, 2000) based on mapping of signal communication between models.

5.1 Algorithmic Approach

The computational resources are fundamental tools in several systems for accomplishing the data processing, data storage and the control execution. According to Ziviani

(ZIVIANI, 2004), the development of the computers program is basically built with the data structure and the algorithms. The form in which the data will be processed depends on the representation of these data. The computational evolution allows the access for faster processors and a larger storage of data with reduced costs. These facts allow the complex data structure implementation and algorithms, which were unfeasible before. In this section the data structures adopted to determine the CWL and of the additional rules of control will be introduced.

Considering the synthesis of the method presented previously, an algorithmic approach is applied for:

- The four steps to generate the CWL and the set of control policy rules to avoid deadlock situation, and;
- The three steps to generate the algorithm control of the CR.

The RAG is used to determine the condition of CWL. For example, Figure 11a presents an example of RAG

with three CWLs: $\{R_1, R_2, R_4 \text{ and } R_3\}$, $\{R_1, R_4 \text{ and } R_3\}$ and $\{R_1, R_2, R_5, R_4 \text{ and } R_3\}$. The representation of a RAG $G = (R, A)$ is based on the adjacency list (NAKAMOTO et al., 2007) instead of the adjacency matrix proposed in Nakamoto (NAKAMOTO et al., 2003). The adjacency list consists of an array Adj of $|R|$ lists, one for each resource in R . For each $R_i \in R$, the adjacency list $Adj[R_i]$ contains all the resource adjacent to R in G . Figure 11b and 11c presents an example of adjacency matrix representation and adjacency list representation.

The tabular form representation of the adjacency list is denominated the Star (DROZDEK, 2002), which can be direct (ForwardStar) or inverse (ReverseStar). The algorithm uses these formats to determine the previous and the posterior node in the RAG model. The great advantage of the adjacency list representation is the complexity of the algorithm to determine the CWL in RAG model, in comparison to the adjacency matrix adopted in Nakamoto (NAKAMOTO, 2002; NAKAMOTO et al., 2003).

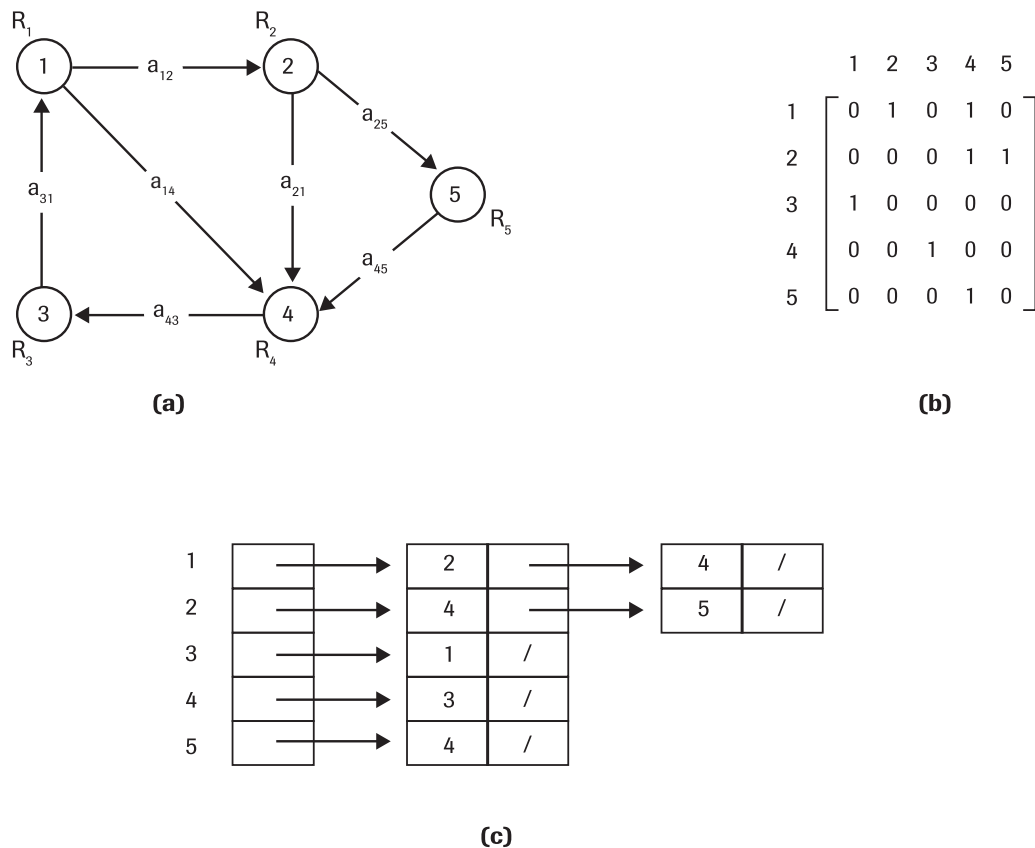


Figure 11: Example of RAG (a), the adjacency matrix (b) and adjacency list representation (c).

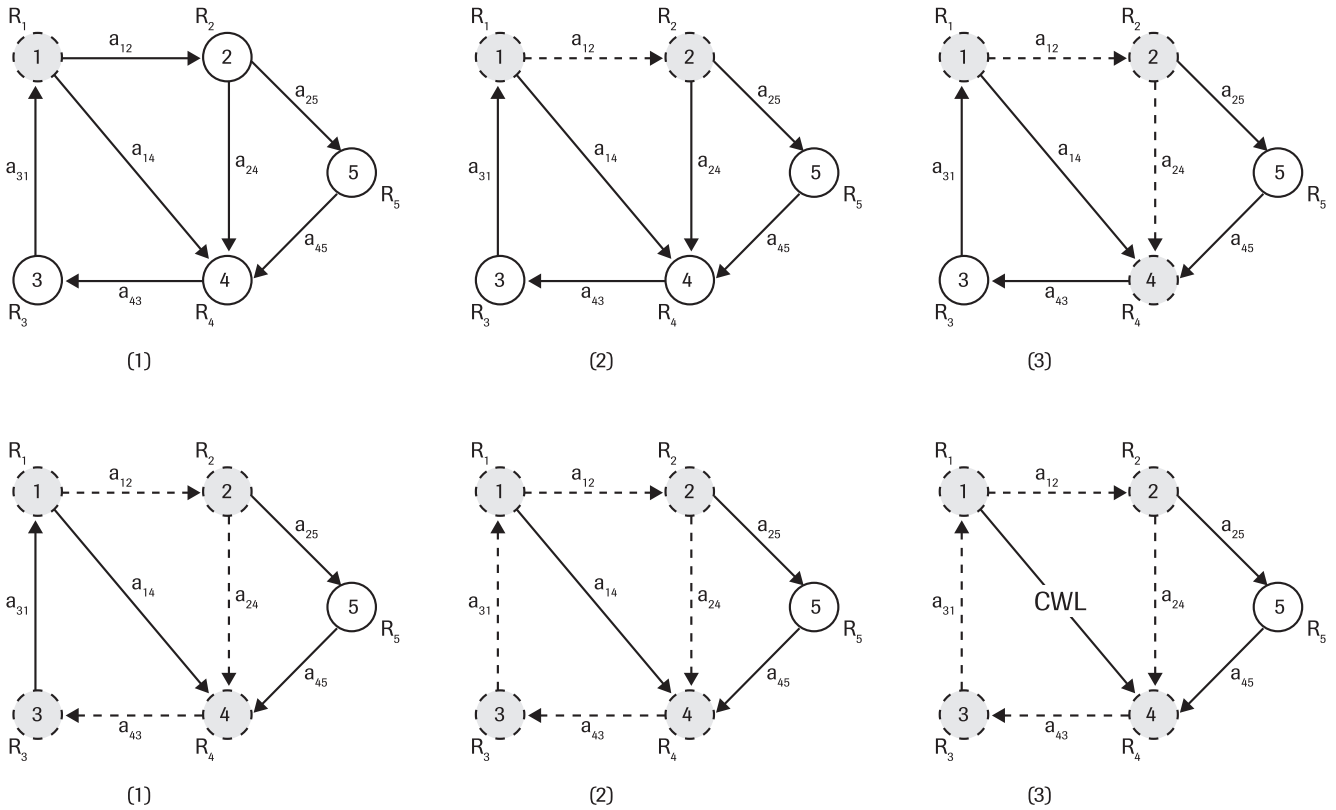


Figure 12: The sequence execution to determine the CWL.

The adjacency matrix multiplication algorithm proposed by Santos Filho (SANTOS FILHO, 2000) has the complexity $\Theta(R^3)$. Nakamoto (NAKAMOTO, 2002; NAKAMOTO et al., 2003) proposed the algorithm to determine the CWL based on depth-first search algorithm (CORMEN et al., 2003) applied in adjacency matrix, in which the complexity of this algorithm is $\Theta(R^2)$. In comparison, the same depth-first search algorithm applied in the adjacency list has the complexity $\Theta(R+A)$ (DROZDEK, 2002; CORMEN et al., 2003; ZIVIANI, 2004). Consequently, the adjacency list representation allows a better performance of computational processing time to determine the CWLs in RAG.

Once the data structure is defined, the algorithm executes a depth-first search in an adjacency list of RAG (searching CWLs). Basically, the strategy is to search the most depth possible in the RAG. The algorithm verifies to each node discovered if there are unexplored adjacent arcs. When all the adjacent arcs are explored, the algorithm returns to the

predecessor node and explores the rest of adjacent resource not yet explored. A CWL is found if the resource discovered was visited previously. Figure 12 illustrates the example of the sequence execution to determine the CWL formed by the resources $\{R_1, R_2, R_4 \text{ and } R_3\}$. The research sequence is initiated in the node 1. Node 1 owns two adjacent arcs a_{12} and a_{14} . From the arc a_{12} , the algorithm discovers node 2 that is inserted in the stack. From node 2, the algorithm discovers node 4 by the adjacent arc a_{24} . When the algorithm finds a node visited previously then it has discovered a CWL.

Figures 13 and 14 present the proposed algorithm and the auxiliary functions. The algorithm determines the CWL when a resource visited previously is discovered. Before inserting CWL in the list, the algorithm verifies if the CWL is already in the list. The algorithm uses the auxiliary data structures to find CWL and insert in the list: data structure stack and status vector of resource. The data

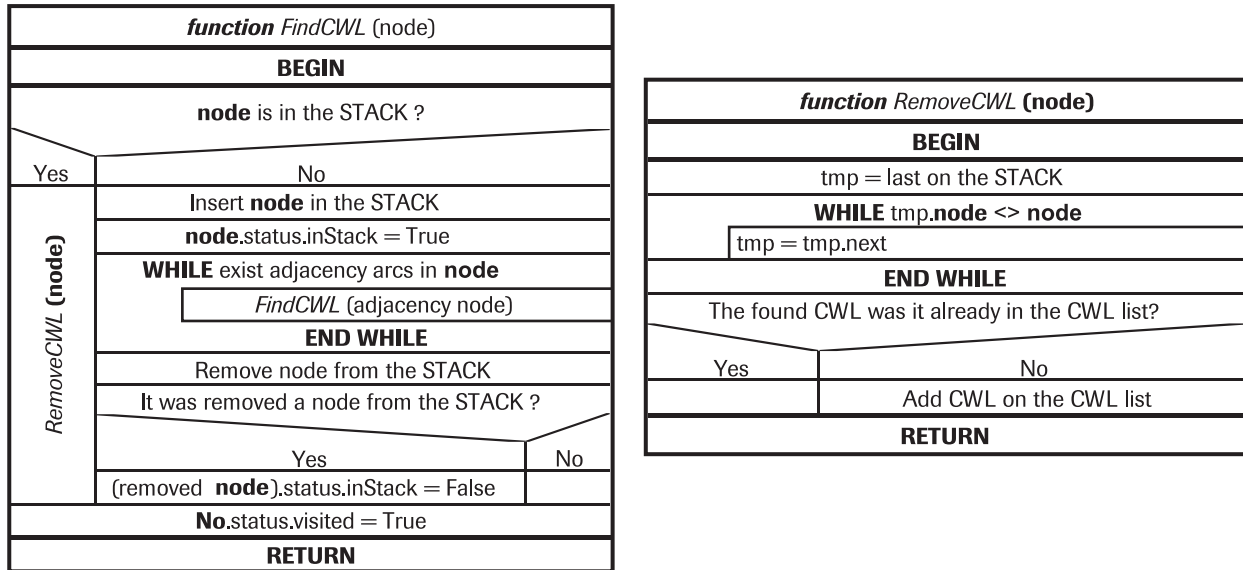


Figure 14: Auxiliary functions.

structure stack is used to store the discovered nodes in the RAG graph. For each node discovered, the respective attribute in the status vector is changed indicating that it was discovered.

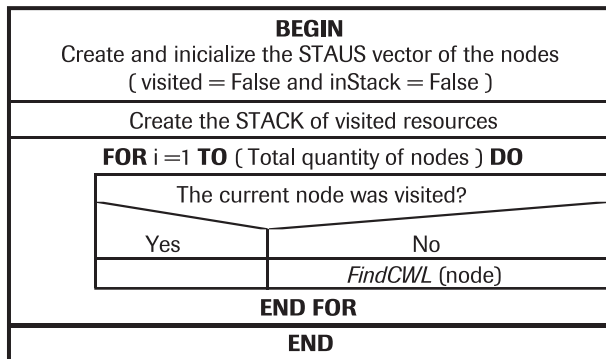


Figure 13: Algorithm to determine the CWL.

The control policy rules to avoid the deadlock situation are generated from the set of CWLs determined by the detection algorithm previously introduced. For each CWL, it is necessary to identify all entrances of the process into the cycle. The pre-deadlock condition is characterized when only one resource is available, considering that the others

resources in CWL are allocated by the processes. Under this condition, the control algorithm must avoid the entrance of the process that causes the deadlock. Figure 15 presents the algorithm to deduce the control policy rules to avoid the deadlock situation.

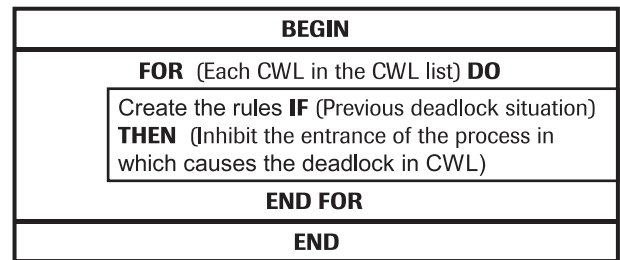


Figure 15: Algorithm to deduce the control policy rules.

The algorithmic approach allowed the development of computer program that generates automatically the control algorithm using the information of the resources, the information of the processes and the sequence of the activities. The software was developed in the IBM-PC Windows XP platform using Borland Delphi 6.

6. CASE STUDY

Consider the following system: processes (A, B, C, D, E, F and G) and resources (R_1 , R_2 , R_3 , R_4 , R_5 and R_6). The resources utilization sequence by the processes is presented in Table 1.

Table 1: Resource utilization sequence by the processes.

Process	Resource Utilization Sequence
A	R1, R2
B	R2, R1
C	R6, R3, R2
D	R3, R5
E	R5, R6
F	R4, R3
G	R1, R4

The following sequence is introduced using the computer program:

- a) The insertion of data about the resources (Figure 16);

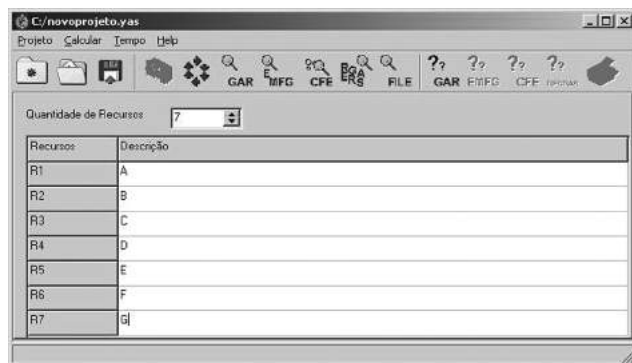


Figure 16: Interface to insert data on the resources.

- b) The insertion of data about the processes (Figure 17);

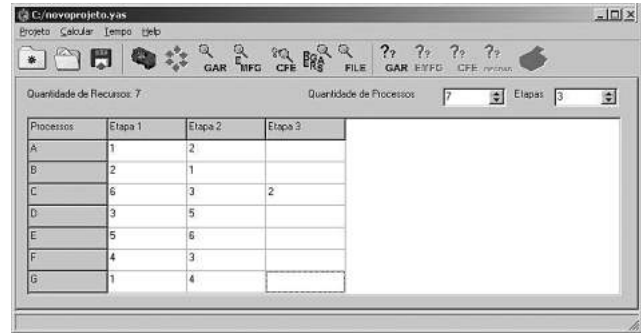


Figure 17: Interface to insert the sequence of activities of the processes.

- c) Determination of global RAG. Figure 18 presents the global RAG, Figure 19 presents the adjacency matrix and Figure 20 the interface of the program with the adjacency matrix of global RAG;
- d) The determination of the CWL. The algorithm is applied to adjacency matrix of global RAG that determines all CWL. Figure 18 presents the three CWLs in global RAG;
- e) The determination of global E-MFG. The global E-MFG is generated from the global RAG, according to the Figure 22. Figure 23 presents the incidence matrix of global E-MFG determined by the program;
- f) The determination of the control policy rules to avoid the deadlock in FMS. Figure 23 presents the CWL with the control rules and Figure 24 presents the rules determined by the program.

7. CONCLUSION

The main contribution of this work is the algorithm approach of the systematization of resource control design for FMS. The results of this work show that it is possible to automatically generate the algorithm control of resource allocation in a simple and fast way. Based on this algorithmic approach, it is possible to implement a modular computational tool that would integrate a computer-aided tool for FMS control system design. The further work needs to be done on the automatic generation of the control of processes integrated with the control of resources.

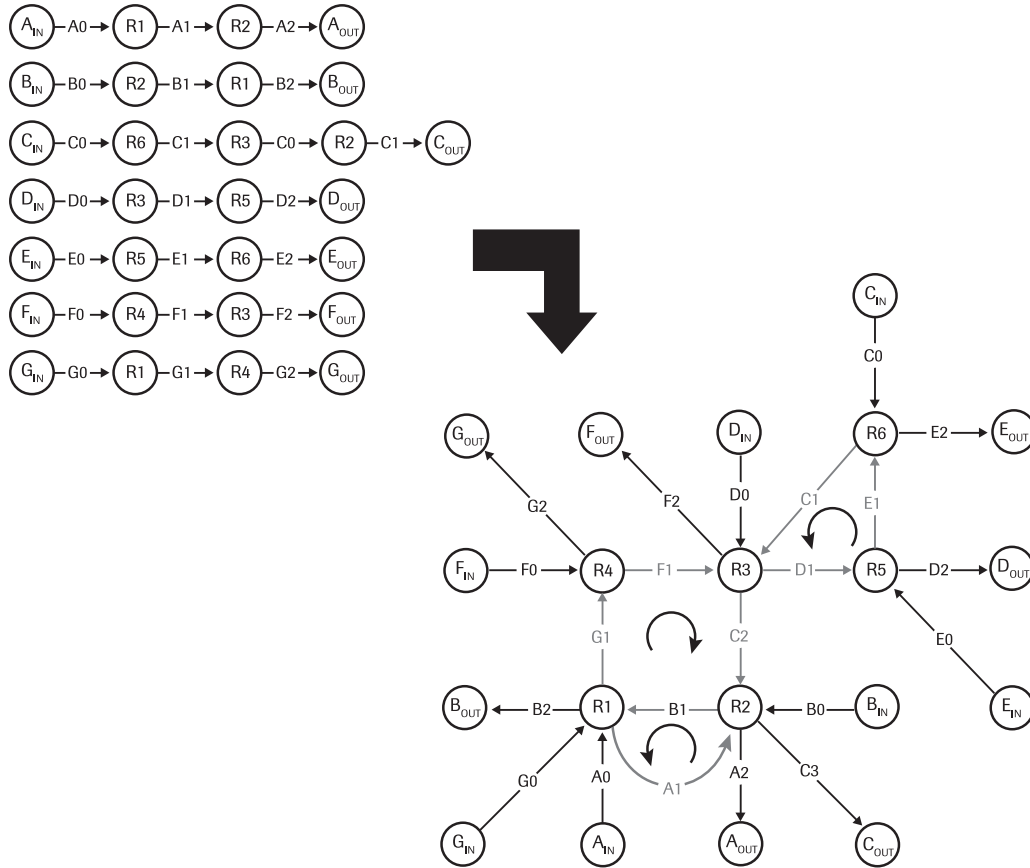
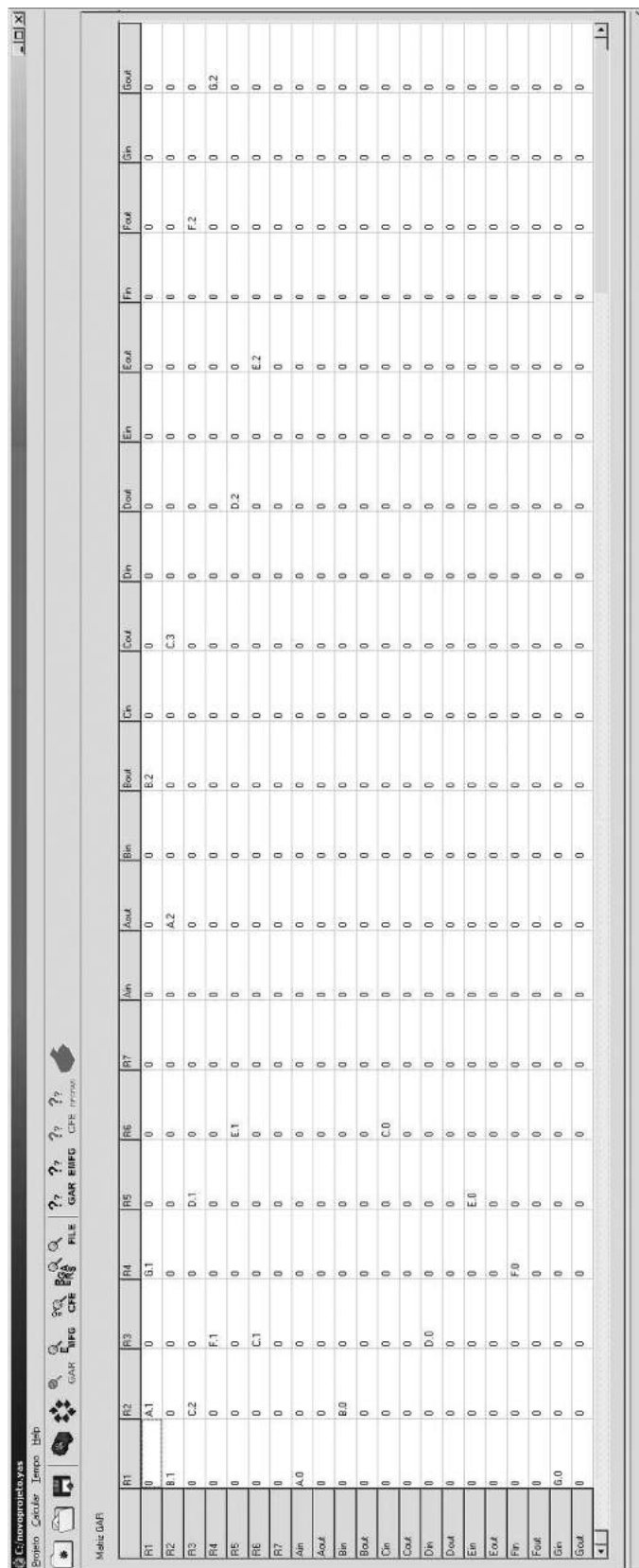


Figure 18 Global RAG.

	R1	R2	R3	R4	R5	R6	A _{IN}	A _{OUT}	B _{IN}	B _{OUT}	C _{IN}	C _{OUT}	D _{IN}	D _{OUT}	E _{IN}	E _{OUT}	F _{IN}	F _{OUT}	G _{IN}	G _{OUT}
R1	0	A1	0	G1	0	0	0	0	0	B2	0	0	0	0	0	0	0	0	0	0
R2	B1	0	0	0	0	0	0	A2	0	0	0	C3	0	0	0	0	0	0	0	0
R3	0	C2	0	0	D1	0	0	0	0	0	0	0	0	0	0	0	0	F2	0	0
R4	0	0	F1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	G2
R5	0	0	0	0	0	E1	0	0	0	0	0	0	0	D2	0	0	0	0	0	0
R6	0	0	C1	0	0	0	0	0	0	0	0	0	0	0	0	E2	0	0	0	0
A _{IN}	A0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
A _{OUT}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
B _{IN}	0	B0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
B _{OUT}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C _{IN}	0	0	0	0	0	C0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C _{OUT}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D _{IN}	0	0	D0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D _{OUT}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
E _{IN}	0	0	0	0	E0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
E _{OUT}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
F _{IN}	0	0	0	F0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
F _{OUT}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
G _{IN}	G0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
G _{OUT}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 19: Adjacency matrix of global RAG.



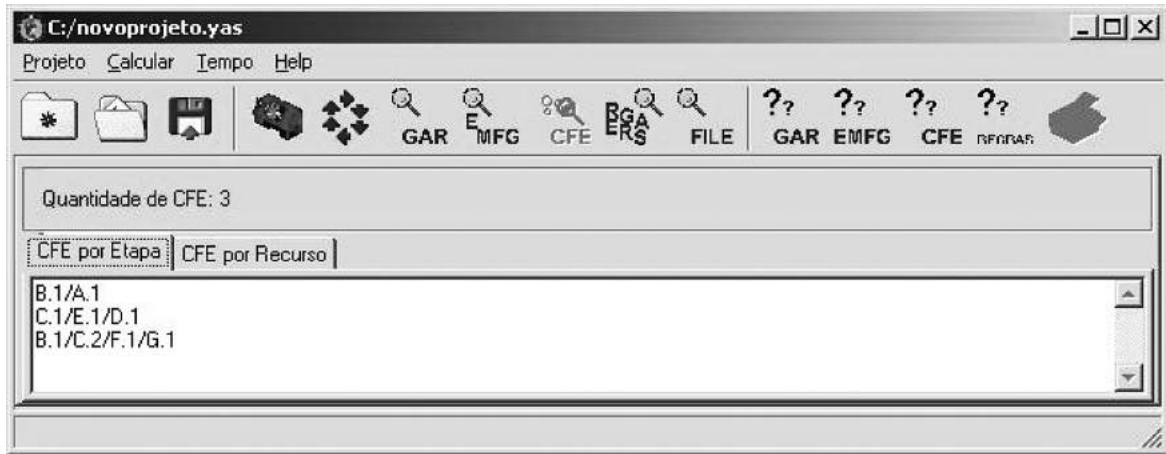


Figure 21: Determination of CWL.

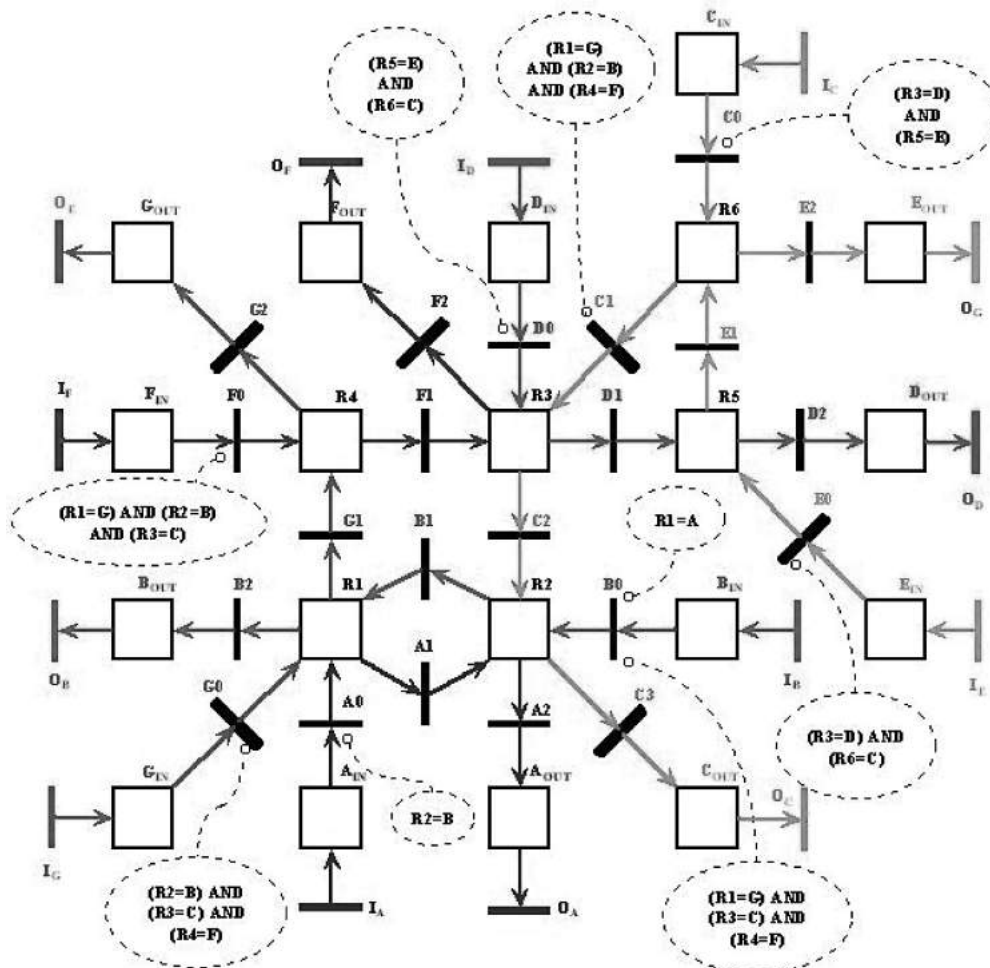


Figure 22: Global E-MFG.

C:\novoprojeto.yas

Projeto Calcular Tempo Help

GAR EMFG CFE BRS FILE GAR EMFG CFE

Matriz E-MFG

Quantidade total de etapas: 29

	R1	R2	R3	R4	R5	R6	R7	Ain	Aout	Bin	Bout	Cin	Cout	Din	Dout	Ein	Eout	Fin	Fout	Gin	Gout
A0	1	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	0	0	0	0	0
A1	-1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
A2	0	-1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
B0	0	1	0	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	0	0	0
B1	1	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
B2	-1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
C0	0	0	0	0	0	1	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	0
C1	0	0	1	0	0	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C2	0	1	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C3	0	-1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
D0	0	0	1	0	0	0	0	0	0	0	0	0	0	-1	0	0	0	0	0	0	0
D1	0	0	-1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D2	0	0	0	0	-1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
E0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	-1	0	0	0	0	0
E1	0	0	0	0	-1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
E2	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
F0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	0	0	0
F1	0	0	1	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
F2	0	0	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
G0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	0
G1	-1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
G2	0	0	0	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Figure 23: Incidence matrix of global E-MFG.

CWL	Rules to avoid <i>deadlock</i> state
E1/C1/D1	IF (R3=D) AND (R5=E) THEN C0=FALSE
	IF (R5=E) AND (R6=C) THEN D0=FALSE
	IF (R3=D) AND (R6=C) THEN E0=FALSE
F1/C2/B1/G1	IF (R1=G) AND (R2=B) AND (R3=C) THEN F0=FALSE
	IF (R1=G) AND (R2=B) AND (R4=F) THEN C1=FALSE
	IF (R1=G) AND (R3=C) AND (R4=F) THEN B0=FALSE
	IF (R2=B) AND (R3=C) AND (R4=F) THEN G0=FALSE
B1/A1	IF (R1=A) THEN B0=FALSE
	IF (R2=B) THEN A0=FALSE

Figure 24: CWL with the control rules.

Artigo recebido em 08/04/2008

Aprovado para publicação em 10/10/2008

ACKNOWLEDGEMENTS

The authors would like to thank CAPES, CNPq, FAPESP, CUFSa and ABEPRO (Associação Brasileira de Engenharia de Produção) for their partial support to this work.

REFERENCES

- BANASZAK, Z. A.; KROGH, B. H. Deadlock Avoidance In Flexible Manufacturing Systems with Concurrently Competing Process Flows. In: *IEEE Transactions on Robotics and Automation*, v. 6, p. 724-734, 1990.
- BANASZAK, Z. A.; POLAK, M. Deadlock-free Distributed Control for Repetitive Flows. *Proceedings of the Sixth International Workshop on Discrete Event Systems*, p. 273-278, 2002.
- CALINESCU, A.; EFSATHIOU, J.; SIVADAN, S., SCHIRN, J.; HUACCHO HUATUCI, L. Complexity in Manufacturing: An Information Theoretic Approach. In: *Proceedings of the International Conference on Complex Systems and Complexity in Manufacturing*, Warwick University, Warwick, 2000.
- CASSANDRAS, C. G. *Discrete Event Systems – Modeling and Performance Analysis*, Richard D. Irvin, Inc. and Aksent Associates, Incorporated Publishers, p. 1-139, 1993.
- CHO, H. An Intelligent Workstation Controller for Computer Integrated Manufacturing, Doctor dissertation, Texas A&M University, p. 83-88, 1993.
- COFFMAN, E. G. JR.; ELPHICK, M. J.; SHOSHANI, A. System Deadlocks. In: *Computing Surveys*, v. 3, n. 2, p. 67-78, June 1971.
- CORMEN, T. H., LEISERSON, C. E., RIVEST, E. L.; STEIN, C. *Introduction to Algorithms*, 2nd Edition, McGraw-Hill, 2003.
- DROZDEK, A. *Estrutura de Dados e Algoritmos em C++*, Editora Pioneira Thomson Learning, 2002.
- EDMONDS, B. What is Complexity? – The Philosophy of Complexity per se with Application to some Examples. In: *Evolution, Symposium: The Evolution of Complexity*, University of Brussels, Brussels, Belgium, 1995.
- EZPELETA, J.; TRINCAS, F.; GARCIA VALLES, F.; COLOM, J. M. A Banker's Solution for Deadlock Avoidance in FMS with Flexible Routing and Multiresource States. In: *IEEE Transactions on Robotics and Automation*, v. 18, p. 621-625, 2002.
- EZPELETA, J.; RECALDE, L. A Deadlock Avoidance Approach for Nonsequential Resource Allocation Systems. In: *IEEE Transactions on Systems, Man and Cybernetics, Part A*, v. 34, p. 93-101, 2004.
- FANTI, M. P.; MAIONE, B.; MASCOLO, S.; TURCHIANO, B. Event-Based Feedback Control for Deadlock Avoidance in Flexible Production Systems. In: *IEEE Transaction on Robotics and Automation*, v. 13, n. 3, p. 347-363, 1997.
- FANTI, M.P.; MAIONE, B.; MASCOLO, S.; TURCHIANO, B. Comparing Digraph and Petri Net Approaches to Deadlock Avoidance in FMS. In: *IEEE Transaction on Systems, Man, and Cybernetics*, v. 30, n. 5, p. 783-798, 2000.
- HSIEH, F. S. Reconfigurable Fault Tolerant Deadlock Avoidance Controller Synthesis for Assembly Production Processes. In: *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics*, v. 4, p. 3045-3050, 2000.
- HSIEH, F. S. Modeling and Control of Holonic Manufacturing Systems Based on Extended Contract Net Protocol. In: *Proceeding of the American Control Conference*, v. 6, p. 5037-5042, 2002.
- HO, Y. C.; CAO, X. R. *Perturbation Analysis of Discrete Event Dynamic Systems*, Kluwer Academics Publishers, 1991.
- ITO, Y. A Dismal Production Structure Looking Toward the 21st Century – Anthropocentric Intelligence-Bases Manufacturing. XI Congresso Brasileiro de Engenharia Mecânica, São Paulo, Brazil, p. 23-32, 1991.
- KUMARAN, T. K.; CHANG, W.; CHO, H.; WYSK, R. A. A Structured Approach to Deadlock Detection, Avoidance and Resolution in Flexible Manufacturing Systems. In: *International journal of production research*, v. 32, n. 10, p. 2361-2379, 1994.
- LAWLEY, M.; REVELIOTIS, S.; FERREIRA, P. Design Guidelines for Deadlock Handling Strategies in Flexible Manufacturing Systems. In: *The International Journal of Flexible Manufacturing Systems*, v. 9, p. 5-30, 1997.
- MIYAGI, P. E. *Programmable Control: Fundamentals of Discrete Event System Control*. São Paulo: Ed. Edgard Blucher, 1996, (In Portuguese).
- MURATA, T. Petri Nets: Properties, Analysis and Applications. In: *Proceedings of the IEEE*, v. 77, n. 4, p. 541-580, 1998.
- NAKAMOTO, F. Y. Sistematização do Projeto do Controle de Sistemas Produtivos, Dissertação de

Mestrado. Escola Politécnica da Universidade de São Paulo, São Paulo, Brazil, 2002. (In Portuguese.)

NAKAMOTO, F. Y.; MIYAGI, P. E.; SANTOS FILHO, D. J. Systematization of the Project of the Production System Control. *IEEE International Symposium on Industrial Electronics*, v. 2, p. 868-873, 2003.

NAKAMOTO, F. Y.; MIYAGI, P. E.; SANTOS FILHO, D. J. Resource Allocation Control in Flexible Manufacturing Systems Using the Deadlock Avoidance Method. In: *Proceedings of COBEM 2007 – 19th International Congress*

of Mechanical Engineering, November 5-9, 2007, Brasília-DF.

PALAZZO, L. A. M.; CASTILHO, J. M. V. *Complex Systems and Self-Organization*. VI Regional School of Informatics, Curitiba, PR, Brazil, 1998. (In Portuguese.)

PETERSON, J. L. *Petri Net Theory and the Modeling of Systems*, Prentice-Hall, 1981.

REISIG, W. (1985) *Petri Nets, an Introduction*. Springer-Verlag, 1985. Berlin Heidelberg.

SANTOS FILHO, D. J. *Aspectos do Projeto de Sistemas Produtivos*. Tese de Livre Docência, Escola Politécnica da Universidade de São Paulo, São Paulo, Brazil, 2000. (In Portuguese).

VISWANADHAM, N.; NARAHARI, Y.; JOHNSON, T. L. Deadlock Prevention and Deadlock Avoidance in Flexible Manufacturing Systems Using Petri Net Models. In: *IEEE Transactions on Robotics and Automation*, v. 6, n. 6, p. 713-723, 1990.

ZIVIANI, N. *Projeto de Algoritmos com implementações em PASCAL e C*, Editora Thomson, 2004.

ABOUT THE AUTHORS

Francisco Yastami Nakamoto

Centro Universitário Fundação Santo André
Tel.: (11) 4979-3300
E-mail: yastami@fsa.br; francisco.nakamoto@poli.usp.br

Paulo Eigi Miyagi

Escola Politécnica da Universidade de São Paulo
Tel.: (11) 3091-5580 Fax.: (11) 3091-5461
E-mail: pemiagi@usp.br

Diolino José dos Santos Filho

Escola Politécnica da Universidade de São Paulo
Tel.: (11) 3091-6025 Fax.: (11) 3091-5461
E-mail: diolino.santos@poli.usp.br