Enhanced Production Flow Schema for Modeling the Complex Resource Sharing System

K. Hasegawa¹, P.E. Miyagi^{2,}, K. Takahashi³,
L. Ma⁴, M. Sugisawa¹*

¹ *Toin University of Yokohama, Japan*

E-mail: hasek@aoba.toin.ac.jp

² *University of São Paulo, Escola Politécnica, Brazil*

E-mail: pemiyaqi@usp.br

³ *Tokyo Institute of Technology, Japan*

⁴ *Oriental Motors. Co., Japan*

Abstract

The authors propose in this paper a novel graphical representation “PFS-R” for complex resource sharing discrete production systems. The PFS-R is an augmented version of PFS (Production Flow Schema, previously proposed by the authors) that includes correspondences of system resources to each production step in PFS. The PFS-R is able to model system structures as well as system behaviors in a more simplified and transparent manner than ordinary Petri net representations. First, the paper introduces basic elements of PFS-R, and equivalent transformation of PFS-R model to a hybrid net and Petri nets, taking into account the net conservativity. Next, we present an example of discrete production system to illustrate the effectiveness of PFS-R. In addition, we deduce through the example that the correspondence between resources and production steps in process flows has the tree structure, which is effective to evaluate whether or not the objects designed have a well-defined structure.

Keywords

Discrete production systems, Petri nets, Production Flow Schema (PFS), Mark Flow Graph (MFG), resource sharing.

The original version of this chapter was revised: The copyright line was incorrect. This has been corrected. The Erratum to this chapter is available at DOI: [10.1007/978-0-387-35390-6_58](https://doi.org/10.1007/978-0-387-35390-6_58)

L. M. Camarinha-Matos et al. (eds.), *Intelligent Systems for Manufacturing*

© IFIP International Federation for Information Processing 1998

1 INTRODUCTION

Aiming at the design of discrete production systems, the authors have previously proposed a hybrid model that takes advantage of Petri nets, the PFS/MFG (Production Flow Schema/Mark Flow Graph) [e.g., Hasegawa, et al., 1988], along with its programming language [e.g., Miyagi, et al., 1988]. This model is employed during the stage of conceptual design, in which we initially represent the production process of discrete production systems using the Production Flow Scheme (PFS), then we use MFG to describe, in a hierarchical manner, the relationships among resources as well as more detailed behaviors of the system. This will then serve as a basis for building both simulation and control software.

The structure of discrete production systems with shared resources (herein called *resource sharing systems*), which offers flexibility in dealing with production of multiple types of parts or products, are getting more and more complex. Examples of these systems are the ones with assembling/disassembling capability, sorting/aggregation capability, and moving resources (e.g., mobile robot, pallet). Because of the increasing complexity of these systems, even with the assistance of a graphical representation such as PFS/MFG, the structure of graphs becomes more and more elaborated, decreasing considerably the visual and intuitive understanding of the system, and thus, eliminating the value as an interface for system design. This has been the biggest drawback of actual adoption of Petri nets as a design technique for *balanced automation systems* [Camarinha-Matos & Afsarmanesh, 1995; Camarinha-Matos & Afsarmanesh, 1996], despite the fact that its benefit has been already recognized.

This paper proposes the graphical technique PFS-R that is an enhanced version, from the aspect of resource arrangement, of the previously proposed PFS for modeling discrete production systems. PFS-R is a result of adding to the production steps of the original PFS a corresponding resource, which allows a concise representation of systems that include resources and thus, is effective as an interface among modelers. Here we will first present the structural elements of PFS-R, then establish a set of rules for the hybrid net models (as an intermediate representation), and show how to convert them to P/T net models (Place/Transition net or ordinary Petri net models). We will then present an example of PFS-R model of a workstation comprised of machine tools, robot or mobile robot, and buffers, indicating its structural conciseness and net conservativity, and in this way, elucidating its effectiveness as a model for evaluating systems during the conceptual design phase.

Since the MFG and all its extensions can be represented by P/T nets, in the following sections we will use P/T nets, instead of MFG, for generalization purposes.

2 PFS (PRODUCTION FLOW SCHEMA)

2.1 PFS and its Elements

PFS is used to graphically describe the processes (jobs) related to production of items (parts, products, information etc.) as a sequence of activity steps and distributing steps. As indicated in Figure 1, PFS consists of activity element nodes, distributing element nodes, and flow arcs that connect sequentially one type of node to the other.

Figure 1 Elements of PFS

The activity element, as shown in Figure 2, can be expanded into two transitions and one place (activity place). When it is necessary to indicate the onset and conclusion of an activity place, we distinguish the input transition (as the onset transition) from the output transition (as the ending transition) (Figure 2c). Moreover, m and n are respectively the simultaneous input and output amounts to and from an element; in P/T nets or hybrid representations, they are the weights of flow arcs. If there is no need in particular, or $m = n = 1$, they can be omitted.

Figure 2 P/T net expansion of activity element

Similarly, the distributing element can also be expanded into a place (distributing place), and its output and input transitions, as indicated in Figure 3. Moreover, in case of expanding into P/T net two consecutive element nodes that are connected to each other, the output transition of the first step will be the same as the input transition of the following step. We will herein call both activity places and distributing places as *step places*.

Figure 3 P/T net expansion of distributing element

2.2 PFS and Resources

Each resource that comprises the production system is related to the progression of process (according to PFS) in the following way:

- [A] Each resource in a production system holds a specific role, related to the step of PFS.
- [B] A process advances systematically through allocation of a resource unit. The amount of resource unit matches the capacity of that resource.
- [C] The resource unit is either being utilized or discharged within the corresponding step.
- [D] The resource unit cannot violate the utilization constraints of the process to which it is related, before the process advances to the next step.

These conditions have often been used in Petri net models for deadlock analysis, which have been proposed in the past [e.g., Zhou, et al., 1992; Hsieh & Chang, 1994; Espeleta, et al., 1995].

Within the conditions previously mentioned, we can categorize the structural relationships of resources in [A] as follows:

- a) the activity steps will always correspond to an active resource such as machines and devices.
- b) the distributing elements can correspond to either active or passive resources. In other words,
 - b-i) they can correspond to a passive resource like a buffer or an activity element of temporary interruption, which denote a cessation of activities between two activity elements, and a pause in a specific place.
 - b-ii) when the resource is to be related indirectly. For instance, it can be a way to prevent the halting of a process within an activity step.

- c) when the step itself is a resource, it represents a moving resource such as mobile robot, AGV, or pallet, indicating a specific condition of the resource corresponding to the step.

The correspondences a) and b-i) are defined as *Step Correspondence (S-correspondence)* and the correspondence b-ii) is defined as *Condition Correspondence (C-correspondence)*.

In addition, there are steps that do not correspond to any of the resources. For instance, the dummy steps that denote control or transformation between two activities do not have any resource assigned to them.

In the next section we will discuss the graphical representations that target these kinds of correspondences within the PFS.

3 PFS WITH RESOURCES – PFS-R

3.1 Correspondence Between Step and Resource, and Resource Arc

Let's consider how to incorporate the presence of resources as a node in the PFS. As indicated in Figure 4, we represent the resource by a thick circle node, and connect it to the related step element through an arc. This arc is herein called *resource arc*.

Figure 4 Resource arc

3.2 Hybrid Net · P/T Net of PFS-R

As indicated in Figure 2 and Figure 3, the step element of PFS can be expanded into a P/T net. Applying this to the PFS-R of Figure 4, the hybrid representation of Figure 5(a) is used according to the needs. Figure 5(b) is the P/T net representation. Similar to step places, in these representations the resource node is called *resource place*. In addition, processes and resource units are represented as tokens, and the firing rules within the hybrid representation of Figure 5(a) follows the P/T net of Figure 5(b).

In the P/T net of Figure 5(b), we will represent the weight of flow arcs from a place p (transition t) to a transition t (place p) by $A(p, t)$ ($A(t, p)$). With respect to the weight of arcs in the transformation previously described, we will establish the following conditions: In Figure 5, $m = A(t1, p) = A(r, t1)$ and $n = A(p, t2) = A(t2, r)$.

Figure 5 Transform of PFS-R

The purposes of these conditions are to consider the tokens within a resource place as an available resource unit, and the token of a step place as a resource unit assigned to this step, and also to continually assure the conservativity of the total amount of resource units. If the weight of arc is two or more we will add this value to the arc, but if it is one or there is no need to specify it, we can omit the value.

When one resource corresponds to two consecutive step elements as indicated in the hybrid net of Figure 6(a), the transformation is performed as shown in Figure 6(b).

Figure 6 One resource for two steps in cascade connection

However, we will make the transition $t2'$ an instantaneous transition. In this way, when this type of hybrid net is transformed into the equivalent P/T net, the resulting representation will be either one of the Figure 7(a)~(c), depending on the difference of arc weights.

4 REPRESENTATION OF RESOURCE SHARING SYSTEMS

We present here the description of PFS-R that expresses the relationship between production flow and its resources, using an example of discrete production system.

Figure 7 P/T net representation of Figure 6

Consider here a discrete production system in which a robot is the shared resource, as indicated in Figure 8 and a batch process using pallets. The docking place of the pallet will be the home position (HP) of the mobile robot *RB*. First, *RB* takes up one pallet (*pu*), moves to the work place *WP1*, and then insert the pallet to input buffer *IB* (*ld/p*). The palletizing robot *RB1* loads three unprocessed parts into the pallet (*ld/w*). Now supplied with parts, the pallet is automatically placed inside the machine *MT*, processed in batch, and discharged to output buffer *OB*. The depalletizing robot *RB2* removes the three processed parts one by one from the pallet (*ul/w*), then the mobile robot *RB* unloads the empty pallet from *OB* (*ul/p*), transport it to the docking place of the pallet, and puts down onto the pre-established place (*pd*).

Figure 9 corresponds to the PFS-R that represents the work content described above.

Figure 8 Lay-out of discrete production system

Figure 9 PFS-R of the discrete production system in Fig. 8

Figure 10 was obtained by transforming the PFS-R of Figure 9 to P/T net, and as we can see, it lacks considerably the property of transparency (resource identification x itens flows) compared to the representation of Figure 9.

Figure 10 Equivalent P/T net of Figure 9

5 RESOURCE ARC AND TREE STRUCTURE

We will here consider the tree representation (herein called *correspondence tree*, or simply *tree*) of the step places and connected resources bearing in mind the resource arcs appeared in the PFS-R obtained in the example presented in 4. Figure 11 shows the set of trees obtained from the example in Figure 9. The root of this tree (represented by ● in the figure) is a resource, and the leaves are step nodes. We will establish that the tree is connected from top to bottom. The nodes in the middle of the correspondence tree are steps and resources at the same time. A node with “.” on its right side and its neighbor node of the right indicate that they lie next to each other within the PFS-R with a precedence relationship.

Figure 11 Set of resource sharing trees obtained from Figure 9

In this way the number of resources, shared resources, dedicated resources are arranged clearly, and also the hierarchical structure of the resource utilization is indicated.

The set of distributing nodes that belong to the same correspondence tree is transformed to P/T net, thus becoming a set of places. However, it is clear that the total number of tokens in this set remains unchanged. Therefore, this set corresponds to the minimum support of P-invariant. This implies that if the step that corresponds to the PFS-R of the system does not have a resource assigned to it, there will be no guarantee that the conservativity of the P/T net representing the system will be maintained. Therefore the correspondence tree is an effective tool to evaluate if the net structure is well-defined in terms of resources.

6 CONCLUSION

In this paper, we presented how the incorporation of system resource information can make the graphical representation of complex resource sharing systems remarkably concise, facilitating the interface among modelers/planners. In addition, we showed here that the correspondence tree that indicates the

relationship between resources and production steps is effective to evaluate whether or not the objects designed have a well-defined structure. It is anticipated that the PFS-R obtained here can be effective in deadlock analysis and in specifying the means for its avoidance, but this will be left for future work.

7 REFERENCES

- Camarinha-Matos, L., Afsarmanesh, H. (Eds.) (1995) *Balanced Automation Systems - Architectures and Design Methods*, Chapman & Hall, London, UK.
- Camarinha-Matos, L., Afsarmanesh, H. (Eds.) (1996) *Balanced Automation Systems II - Implementation Challenges for Antropocentric Manufacturing*, Chapman & Hall, London, UK.
- Ezpeleta, J., et al. (1995) A Petri net based deadlock prevention policy for flexible manufacturing systems, *IEEE Trans. of Robotics and Automation*, **11-2**, 173-184.
- Hasegawa, K., et al. (1988) Application of the Mark Flow Graph to represent discrete event production systems and system control, *Trans. of SICE*, **24-1**, 69-75.
- Hsieh, F.S., Chang, S.C. (1994) Dispatching driven deadlock avoidance controller synthesis for flexible manufacturing systems, *IEEE Trans. of Robotics and Automation*, **10-2**, 196-209.
- Miyagi, P.E., et al. (1988) A programming language for discrete production systems based on Production Flow Schema and Mark Flow Graph, *Trans. of SICE*, **24-2**, 183-190.
- Zhou, M., et al. (1992) A Hybrid methodology for synthesis of Petri net models for manufacturing systems, *IEEE Trans. of Robotics and Automation*, **8-3**, 350-361.

8 BIOGRAPHY

Kensuke Hasegawa is currently a professor of Toin University of Yokohama, and prior a professor of Tokyo Institute of Technology (TITech), Japan, from 1972 to 1990. He received the Dr.Eng. degree in Electrical Eng. from TITech, in 1962. Currently his main research interest is the modelling and control of discrete production systems and discrete event system control based on Petri nets.

Paulo E. Miyagi is currently a professor of University of São Paulo, Brazil. He received the Dr.Eng. degree in Control Eng. from TITech, Japan, in 1988. Currently his main research interest is the modelling and control of discrete event systems based on Petri nets.

Kouji Takahashi is currently a associate professor of TITech, Japan,. He received the Dr.Eng. degree in Control Eng. from TITech, in 1985. Currently his main research interest is the analysis and control of event driven systems based on Petri nets.

Li-qu Ma is presently an engineer of Oriental Motors Co., of Japan, He received the M.Eng. degree in Control Eng. from TITech, in 1987. His main area of interests is the modelling and control of discrete production systems based on Petri nets and the simulation of production systems.

Makiko Sugisawa is presently a Ph.D. student of Toin University of Yokohama, Japan. She received the M.Eng. degree in Control and Systems Eng. from the same University, in 1995. Her main area of interests is the modelling and deadlock avoidance for discrete production systems based on Petri nets.