applied sciences

MDPI

Article

Enhancement of Computational Efficiency in Seeking Liveness-Enforcing Supervisors for Advanced Flexible Manufacturing Systems with Deadlock States

Yen-Liang Pan [1,2,*], Chun-Wang Tai [1], Ching-Yun Tseng [3] and Jong-Ching Huang [1]

1. Department of Electrical Engineering, National Kaohsiung University of Science and Technology, Kaohsiung 80778, Taiwan; apple15328@yahoo.com (C.-W.T.); ching@mail.ee.nkust.edu.tw (J.-C.H.)
2. Mathematics and Physics Division, General Education Center, Air Force Academy, Kaohsiung 82047, Taiwan
3. Department of Computer Science & Information Engineering, National Kaohsiung University of Science and Technology, Kaohsiung 80778, Taiwan; xxzz5801@gmail.com

* Correspondence: peterpan960326@gmail.com or yenliangpan@nkust.edu.tw; Tel.: +886-7-626-8846

Received: 25 February 2020; Accepted: 7 April 2020; Published: 10 April 2020

Featured Application: In the paper, we further propose two novel concepts called Pre Idle Transitions (PIT) and Pre Idle Places (PIP) for the IMFFP-2. Once PIT or PIP is identified from a deadlock Petri nets model, one can bypass all PIP under the process of solving MFFP, and the computational time can hence be shortened.

Abstract: In industry 4.0, all kinds of intelligent workstations are designed for use in manufacturing industries. Among them, flexible manufacturing systems (FMSs) use smart robots to achieve their production capacity under the condition of a high degree of resources sharing. As a result, deadlock states usually appear unexpectedly. For solving the damage deadlock problem, many pioneers have proposed new policies. However, it is very difficult to make systems maximally permissive even if their policies can solve the deadlock problem of FMSs. According to our survey, the Maximal number of Forbidding First Bad Marking (FBM) Problems (MFFP) seems to be the best technology to obtain systems' maximally permissive states in the existing literature. More importantly, the number of added control places (CP) is the smallest among the existing research works. However, when the complexity of a flexible manufacturing system increases, the computational burden rises rapidly. To reduce computational cost, we define a new concept named Pre Idle Places (PIP) to enhance the computational efficiency in Seeking Liveness-Enforcing Supervisors. We can bypass all PIP once they can be identified from a deadlock system under the process of solving MFFP. According to the data showed in three classical examples, our proposed Improved MFFP is better than conventional MFFP in terms of computational efficiency with the same controllers.

Keywords: Petri nets; Flexible manufacturing system; Discrete event dynamic systems; Liveness-Enforcing Supervisors; Deadlock Prevention

1. Introduction

Since flexible manufacturing systems (FMSs) use smart robots to achieve their production capacity under the condition of a high degree of resources sharing, deadlock states usually appear unexpectedly. For solving the deadlock problems of FMSs, many pioneers have focused on this issue [1–16].

Generally, three kinds of technologies are proposed to solve the deadlock problems of FMSs: deadlock avoidance, deadlock detection and recovery, and deadlock prevention [7]. Firstly, deadlock avoidance is adopted to prevent an FMS from reaching any deadlock state. Although it has a higher efficiency, it always fails to eliminate all deadlock states. The deadlock detection and recovery approach allows existing deadlock states to be reachable. Once deadlock occurs, the system detects and reallocates resources for recovery. Finally, deadlock prevention is proposed to prevent a deadlock situation from being reached. It is important to note that deadlock prevention does not need any run-time cost, since it solves the deadlock of FMSs in the design and planning stages.

In the deadlock prevention domain, two main analysis methods are designed to solve FMS deadlock problems: structural and reachability graph analysis, respectively. The former utilizes some structural items, such as all kinds of siphons [6,13,14,17–27]. Some generated siphons are found in uncontrolled models, which may become unmarked or partially marked. A siphon denotes a general set of places. In the case of a deadlock, the siphon cannot regain any new marking from the original ones, i.e., all transitions in this model are disabled, and a deadlock is recognized. To prevent a deadlock state from being reached, no places in these siphons may be empty. The objective is reached by designing and adding a control subnet to the model with its initial marking. Huang et al. [13] develop a formalized and iterative approach with two main stages. These two stages are required in each iteration in order to create new siphons after attaching them to the FMS model. Li and Zhou [18] propose a method with a much smaller control subnet than in previous works, with the same results. Huang et al. [14] introduce an iterative method, adopting two kinds of control places: ordinary and weighted control places. They attempt to prevent siphons from losing their tokens and guarantee more permissive systems. However, structural analysis seems suboptimal and leads to low utilization rates of system resources, although it does lower computational cost.

Reachability graph analysis [28–37] requires the enumeration of all generated markings. The set of these markings is hence referred to as the reachability graph (RG). The RG includes all reachable markings, which can be classified as one of two parts: the live zone (LZ) and deadlock zone (DZ). The DZ contains deadlock markings and critical bad markings, which inevitably lead to deadlock zones. The LZ contains all live markings and is also recognized as a deadlock-free-zone (DFZ). The objective is to forbid all markings in the DZ and to make sure all markings in the LZ are still reachable. According to the characteristics of RG and system complexity, marking explosion problems may occur during the computation of reachability graph analysis, implying that the increase of FMS model size leads to a corresponding increase in computational cost. Uzam [28] develops a deadlock prevention policy based on a theory of regions that proves maximal liveness performance. However, the policy fails to determine all sets of event-state separation problems (ESSPs), and its application seems limited to certain special nets only. Therefore, some works [34–37] aim to develop a more computationally efficient optimal deadlock control policy based on the theory of regions. Uzam and Zhou [5] propose a reduction technique to simplify large FMS models. To reduce computation, first-met bad markings (FBMs) are adopted in these works. FBMs are the set of markings in the DZ and the first entrance from the LZ to the DZ. However, as indicated by [38], this process requires the repeated calculation of reachability graphs.

Pirrodi et al. [39] consider selective siphons to reduce redundancy problems and provide small-size controllers. The control policy solves the deadlock problems of FMSs successfully. It also makes FMSs optimal. However, as indicated by [40], the process of eliminating all critical markings is time-consuming, since all legal markings must be checked in each iterative step. According to the above descriptions, it is clear that determining how to obtain optimal (maximally permissive) controllers for solving the FMS deadlock problem seems an extremely time-consuming and difficult issue. Therefore, in [33] Pan et al. also propose a policy called the enhancement of selective siphon control method to improve the efficiency of deadlock prevention in FMSs.

For improving the above disadvantage, Chen et al. [1] brought forward a new concept named MFFP (i.e., the Maximal number of Forbidding First Bad Marking (FBM) Problems) to obtain maximally permissive reachable markings with a much smaller number of controllers based on the reachability graph analysis method and Place Invariant (PI) [41]. It successfully solves the problem of the excessive consumption of time. However, when the complexity of a flexible manufacturing system increases, the computational burden rises rapidly. Besides, in our survey the MFFP could only be applied in certain special nets, and outside S^{3}PR [6], the method would fail. Therefore, in our previous paper [42], we proposed an iterative method called IMFFP-1 to solve the disadvantage of [1] so that it can be used in all general cases whether the controlled net is optimal or suboptimal. Subsequently, for reducing computational time, in [43] we proposed two novel concepts called Pre Idle Transitions (PIT) and Pre Idle Places (PIP). However, the proposed definitions seemed rough and immature.

Therefore, in this paper, we aim to redefine and give the formal definitions of PIT and PIP. Please note that we can bypass all PIP once they can be identified from a deadlock flexible manufacturing system under the process of solving MFFP. From the resulting data based on three classical examples, the further improved MFFP (called IMFFP-2) method seems to achieve a better computational efficiency with the same controllers. Please note that in this paper we focus on enhancement of computational efficiency for the conventional MFFP. Therefore, in this paper we consider the same kind of Petri Net (PN) class as that in [1,2]. On the other hand, some deadlock preventions that can recover all reachable markings using transition-based controllers [44–49] and inhibitor arcs [50] are also proposed. In the future, we will follow the work in [51] to discuss a divide-and-conquer-method and try to enhance their computational efficiency.

Section 2 gives a description of the basic definitions and properties of PNs. Section 3 presents the proposed policy. Section 4 shows examples and related experimental data. Section 5 makes comparisons with the existing literature. Finally, conclusions are drawn in Section 6.

2. Preliminaries

2.1. Petri Nets (PNs)

A Petri Net [52], also called a Place/Transition Net, is a four-tuple N = (P, T, F, W). Both P and T are non-empty sets. Generally, P is a finite set of places, and T is a finite set of transitions, disjointed to each other. The set $F \subseteq (P \times T) \cup (T \times P)$ is the arcs of this net, which is represented by one-way arrows from its nodes to denote a flow relation in a PN model. The mapping $W : (T \times P) \cup (P \times T) \to \mathbb{N}$ assigns a weight to an arc: $W(x, y) > 0$ if $(x, y) \in F$, and $W(x, y) = 0$ otherwise; where $x, y \in P \cup T$ and $\mathbb{N}$ is the set of non-negative integers. We call $x^\bullet = \{y \in P \cup T \mid (x, y) \in F\}$ the postset of x, and $^\bullet x = \{y \in P \cup T \mid (y, x) \in F\}$ is called the preset of x. A marking M is a multi-set of its places, which allocates tokens to each place of PNM. We denote $M(p)$ as the number of tokens in place p. The pair (N, M_0) is called a marked Petri net or a net system. A net is pure if $\forall (x, y) \in (P \times T) \cup (T \times P)$, $W(x, y) > 0$, implying that $W(y, x) = 0$. The incidence matrix $[N]$ of the pure net N is a $|P| \times |T|$ integer matrix with $[N](p, t) = W(t, p) - W(p, t)$.

A transition $t \in T$ is enabled at marking M if $\forall p \in {}^\bullet t$, $M(p) \geq W(p, t)$, which is denoted as $M[t\rangle$. Once an enabled transition t fires, it generates a new marking M', denoted as $M[t\rangle M'$, where $M'(p) = M(p) - W(p, t) + W(t, p)$. The set $M[\rangle$ represents all markings reachable from M by firing any possible sequence of transitions. The set $M_0[\rangle$ is the reachable markings of net N with initial marking M_0, often denoted as $R(N, M_0)$. It can be graphically indicated by a reachability graph, which can, in turn, be denoted as $G(N, M_0)$. It is a directed graph in which each node represents a marking in $R(N, M_0)$, and arcs are labeled by the fired transitions.

Let (N, M_0) be a net system with $N = (P, T, F, W)$. A transition $t \in T$ is live at M_0 if $\forall M \in R(N, M_0)$, $\exists M' \in R(N, M_0)$, $M'[t\rangle$. The pair (N, M_0) is live if $\forall t \in T$, t is live at M_0. It is dead at M_0 if $\nexists t \in T$, $M_0[t\rangle$.

A P-vector is a column vector $I : P \to \mathbb{Z}$ indexed by P. P-vector I is called a P-invariant (place invariant, PI for short) if $I \neq 0$ and $I^T[N] = 0^T$. P-invariant I is said to be a P-semiflow if $I \geq 0$. Let I be a PI of (N, M_0) and M be a reachable marking from M_0. Then, $I^T M = I^T M_0$.

2.2. Identify All FBMs

In the Reachability Graph (RG) method, all reachable markings in the initial reachability graph analysis can be divided into two main groups: illegal and legal markings [35]. Illegal markings include deadlock and quasi-deadlock markings. The set of deadlock markings M_D is defined as follows:

Definition 1 [35]**.** *The set of deadlock markings $\mathcal{M}_D$= {M∈ R(N, M_0)| at M; no transition can be enabled}.*

The quasi-deadlock markings are defined as follows:

Definition 2 [35]**.** *The set of quasi-deadlock markings $\mathcal{M}_Q$ = {M∈R(N, M_0)| M must eventually evolve into a dead one regardless of transition firing sequences}.*

The reachable markings, not including quasi-deadlock and deadlock markings, are called legal markings. The set of legal markings $\mathcal{M}_L$ is the maximal number of reachable markings of a system, from which the initial marking M_0 is reachable without leaving $\mathcal{M}_L$. Further, the set of legal markings $\mathcal{M}_L$ can be defined as follows:

Definition 3. $\mathcal{M}_L = \left\{MR(N,\ M_0) \wedge M\left(M_D \cup\ M_Q\right)\right\}$.

The $\left(M_D \cap\ M_Q\right)$ of Definition 3 signifies the set of illegal markings. Therefore, the set of illegal markings $\mathcal{M}_I$ can be defined as:

Definition 4. $\mathcal{M}_I = \left\{MR(N,\ M_0) \wedge M\left(M_D \cup\ M_Q\right)\right\}$.

Definition 4 means that all dead and quasi-dead markings can also be viewed as the set of illegal markings. Further, $\mathcal{M}_I$ can be reached directly from any legal marking in an RG of a PNM. Please note that the first-met illegal marking is the same as the first-met bad marking (FBM) [1]. In our opinion, the first-met illegal marking (FIM) seems to be superior to the FBM. Therefore, the definition of first-met illegal marking (FIM) is presented formally as follows:

Definition 5. $\mathcal{M}_{\mathrm{FIM}} = \{M'\big|MR(N,\ M_0), M \in \mathcal{M}_L, M' \in \mathcal{M}_I, tT,\ s.t.\ M[tM'\}$

According to an example, one can understand the $\mathcal{M}_{\mathrm{FIM}}$ easily. Referring to Figure 1, the PNM has two idle places (i.e., p_1 and p_8, both with three tokens in them), six operation places (p_2~p_6), and three resource places (p_9~p_{11}, all with one token in them). Figure 2 is its reachability graph. In Figure 2, there are 20 reachable markings in total. Further, according to Definitions 1~4, there are 15 $\mathcal{M}_L$ and 5 $\mathcal{M}_I$ (i.e., 2 $\mathcal{M}_D$ and 3 $\mathcal{M}_Q$) in Figure 2. In this example, all $\mathcal{M}_Q$ (i.e., deadlock marking M_8, quasi-deadlock marking M_{10}, quasi-deadlock marking M_{11}, quasi-deadlock marking M_{12} and deadlock marking M_{13}) are FIMs since they all fit Definition 5.

In the following section, an improved deadlock prevention policy with an enhancement of computational efficiency is proposed. In this deadlock policy, just a few FBMs need to be identified. Once the few FBMs in the reachability graph are forbidden, the system is deadlock-free. In other words, under the same number of controllers, the efficiency of our algorithm is better than the MFFP method [1]. The detailed information of the proposed policy is presented as follows.

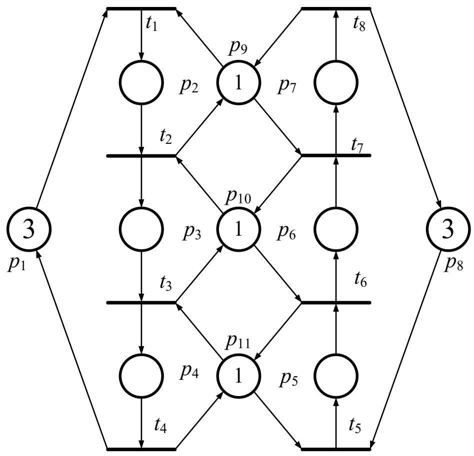

Figure 1. A simple flexible manufacturing system (FMS) model based on Petri Net (PN) theory.

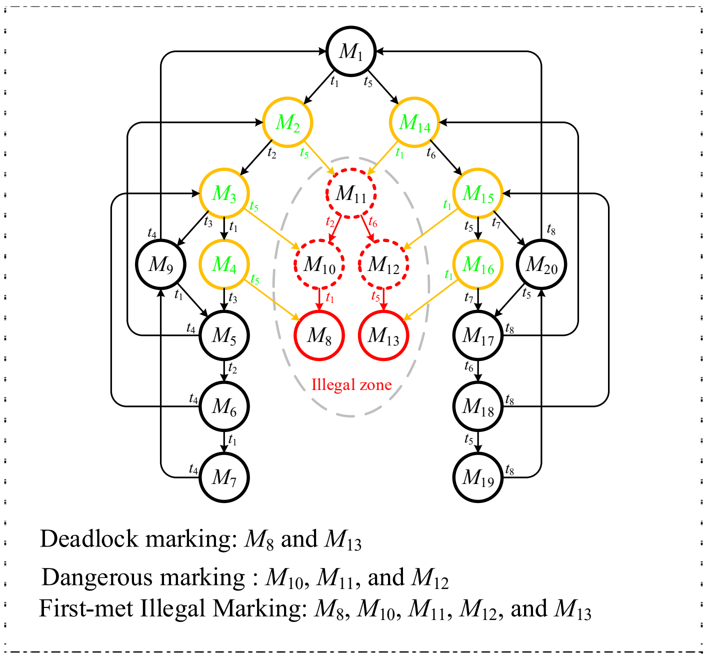

Figure 2. The reachability graph of Figure 1.

3. Improved Policy with Enhancement of Computational Efficiency

In this section, we will present our improved policy and further try to redefine and give the formal definitions of PIT, PIP and Crucial Vector Covering Approach for enhancing the computational efficiency of the conventional MFFP.

3.1. Finding Controllers from Place Invariant (PI) Concept

According to the definitions in [5,8], three types of places are given for FBM based on PN theory, including Idle places P^0, Resource places P_R, and Activity places P_A, respectively. The initial marking of a reachability graph of a PNM is denoted as M_0. The activity places must be considered among all kinds of places in using the PI concept [41].

Therefore, the PI concept is the most important factor and will help in designing maximally permissive controllers in this section. In [3], Yamalidou et al. first proposed the PI concept as follows:

$$\beta \geq \sum_{i=1}^{n} l_i \cdot \mu_i \tag{1}$$

where both β and l_i are non-negative integers, and μ_i denotes the number of tokens in place p_i. Furthermore, for calculating the controllers of a system, more non-negative integers μ_s are given and the new equality is as follows:

$$\beta = \sum_{i=1}^{n} l_i \cdot \mu_i \; + \mu_s \tag{2}$$

By using Equation (2) every time, one can obtain control places so that some illegal markings can be removed until the FMS becomes deadlock-free. However, it could simultaneously eliminate some legal markings. Therefore, Chen et al. [1,11] propose a novel concept called MFFP to improve the above PI control method to hold the maximally permissive live legal markings.

3.2. MFFP

By identifying a few useful FBMs based on conventional PI control concepts, Chen et al. [1,11] proposed a novel PI control method, MFFP, to identify all maximally permissive controllers (i.e., control places). Accordingly, there are two kinds of algorithms based on the MFFP presented in their papers [1,11]. However, according to our study, the two algorithms seem logically identical. Therefore, in this paper, we select the first type of MFFP algorithm as the object to improve and enhance its computational efficiency. In this paper, we still call the two kinds of algorithms MFFP.

In [1], for obtaining maximally permissive PI controllers, there are two constraints used to design the Integer Linear Programming Problem (ILPP). In other words, once the optimal PI controllers are added into the original system, all FBMs can then be excluded from all legal markings. This system becomes live and keeps the maximally permissive (or optimal) markings since just FBMs are forbidden. In addition, only activity places need to be considered since the ILPP belongs to PI control. Therefore, based on Equation (1), a set $N_A = \{i \middle| p_i \in P_A\}$ is further used to redefine Equation (1) as follows.

$$\beta \geq \sum_{i \in N_A}^{n} l_i \cdot \mu_i \tag{3}$$

In Equation (3), it is considered that each FIM can be forbidden by every relative controller added to the deadlock system. μ_i can be replaced by $M(p_i)$, and variable β can be redefined in Equation (4):

$$\beta = \sum_{i \in N_A}^{n} l_i \cdot M_I(p_i) - 1 \tag{4}$$

Following Equation (4), variable β must smaller than $\sum_{i \in N_A}^{n} l_i \cdot M(p_i)$ since $\beta = \sum_{i \in N_A}^{n} l_i \cdot M(p_i) - 1$ is true. Accordingly, one new constraint can be given as follows:

$$\beta < \sum_{i \in N_A}^{n} l_i \cdot \mu_i \tag{5}$$

On the other hand, for holding all legal markings, the calculation is as follows:

$$\beta \geq \sum_{i \in N_A}^{n} l_i \cdot M_L(p_i) \tag{6}$$

By using Equations (5) and (6), the optimal PI controller developed by Chen et al. [1,12] can then be obtained.

3.3. Crucial Vector Covering Approach

Under the process of calculating all PI controllers, the computational cost is very high if all the reachable markings need to be considered. Computational cost and computer time could be reduced once crucial legal markings and FBMs can be identified. Based on the above reason, a Crucial Vector Covering Approach (CVCA) is proposed and defined in this subsection. Firstly, the conventional Vector Covering Approach (VCA) is introduced as follows:

Definition 6 [1]. *Let there be any two markings M_a and M_b in $R(N, M_0)$. The marking M_a covers M_b if $\forall p \in P_A$, $M_a(p) \geq M_b(p)$, which is denoted as $M_a \geq M_b$.*

Theorem 1 [1]. *Let M_a and M_b be any two markings in $R(N, M_0)$ with $M_a \geq M_b$. If M_b is forbidden by one PI, M_a will be also forbidden. On the contrary, if M_a is not forbidden by a PI, M_b will not be forbidden either.*

Please note that Definition 6 seems imprecise according to our experimental results. For example (please refer to Figure 2), M_5 $(p_2 + p_4)$ and $M_{19}(p_5 + p_6 + p_7)$ are any two markings, with $M_5(p) = 2$ and $M_{19}(p) = 3$, respectively. For comparison, the detailed information of all reachable markings in Figure 2 is presented in Table 1.

Table 1. The properties of all reachable markings in Figure 2.

Marking No.	Classification	Marking Information
M_1	Initial Marking	$3p_1 + 3p_8 + p_9 + p_{10} + p_{11}$
M_2	Live Marking	$2p_1 + p_2 + 3p_8 + p_{10} + p_{11}$
M_3	Live Marking	$2p_1 + p_3 + 3p_8 + p_9 + p_{11}$
M_4	Live Marking	$p_1 + p_2 + p_3 + 3p_8 + p_{11}$
M_5	Live Marking	$p_1 + p_2 + p_4 + 3p_8 + p_{10}$
M_6	Live Marking	$p_1 + p_3 + p_4 + 3p_8 + p_9$
M_7	Live Marking	$p_2 + p_3 + p_4 + 3p_8$
M_8	Deadlock Marking	$p_1 + p_2 + p_3 + p_5 + 2p_8$
M_9	Live Marking	$2p_1 + p_4 + 3p_8 + p_9 + p_{10}$
M_{10}	Quasi-deadlock Marking	$2p_1 + p_3 + p_5 + 2p_8 + p_9$
M_{11}	Quasi-deadlock Marking	$2p_1 + p_2 + p_5 + 2p_8 + p_{10}$
M_{12}	Quasi-deadlock Marking	$2p_1 + p_2 + p_6 + 2p_8 + p_{11}$
M_{13}	Deadlock Marking	$2p_1 + p_2 + p_5 + p_6 + p_8$
M_{14}	Live Marking	$3p_1 + p_5 + 2p_8 + p_9 + p_{10}$
M_{15}	Live Marking	$3p_1 + p_6 + 2p_8 + p_9 + p_{11}$
M_{16}	Live Marking	$3p_1 + p_5 + p_6 + p_8 + p_9$
M_{17}	Live Marking	$3p_1 + p_5 + p_7 + p_8 + p_{10}$
M_{18}	Live Marking	$3p_1 + p_6 + p_7 + p_8 + p_{11}$
M_{19}	Live Marking	$3p_1 + p_5 + p_6 + p_7$
M_{20}	Live Marking	$3p_1 + p_7 + 2p_8 + p_{10} + p_{11}$

Accordingly, $M_{19}(p) \geq M_5(p)$ is checked. Therefore, according to Definition 6, M_5 must be covered by M_{19} since $M_{19}(p) \geq M_5(p)$. In addition, according to Theorem 1, if M_5 is forbidden by the PI controller, M_{19} must be also forbidden. However, in fact, M_{19} is not forbidden in the situation that M_5 is forbidden. Therefore, we revise Definition 6 so that it can be accurate and can fit Theorem 1. The new definition is presented as follows:

Definition 7. *Let M_a and M_b be any two markings in $R(N, M_0)$. The marking M_a is said to cover M_b if $\forall p \in P_A$, $(M_a(p) \geq M_b(p)) \wedge (M_a \supseteq M_b)$*, which is denoted as $M_a \geq M_b$

.

Please note that we add the extra condition $M_a \supseteq M_b$ in Definition 7. Further, $(M_a \cap M_b) = M_b$ so it can be precise and fit Theorem 1. Accordingly, we use Definition 7 instead of Definition 6 in this paper.

We also provide an example so that the readers can follow (please also refer to Figures 1 and 2). The markings M_6 and M_7 are any two markings in the reachability graph of Figure 1. Further, their properties are $M_6 = \{p_3, p_4\}$ and $M_7 = \{p_2, p_3, p_4\}$, respectively. The number of idle places in $M_6(p)$ is equal to 2 and $M_7(p)$ is equal to 3. Then, $M_7(p) \geq M_6(p)$. It is worth noting that $M_7 \supseteq M_6$ since $p_2 + p_3 + p_4 \supseteq p_3 + p_4$ and $(M_7 \cap M_6) = M_6$. Surely, $M_7(p) \geq M_6(p)$. We check if Definition 7 fits Theorem 1 by using $M_7(p) \geq M_6(p)$. It is obvious that once M_6 is forbidden, M_7 is forbidden. Theorem 1 is proved. In the following, the concept of a minimal covered set of FBMs is presented. In [2], Chen et al. use it to improve the conventional PI control method and determine the maximally permissive controllers. Therefore, we show its definition first. Please note that in this paper, we propose Pre Idle Places (PIP) so that the computational efficiency can then be enhanced further. First, we revise Definition 6 as follows:

Definition 8. *Set $\mathcal{M}^{\star}_{FIM} \subseteq \mathcal{M}_{FIM}$. $\mathcal{M}^{\star}_{FIM}$ is called the minimal covered set of $\mathcal{M}_{FIM}$ if the following conditions are satisfied:*

1) $\forall M_A \in \mathcal{M}_{FIM}$, $\exists M_B \in \mathcal{M}^{\star}_{FIM}$, *s.t.* $M_A \geq M_B$;

2) $\forall M_A \in \mathcal{M}^{\star}_{FIM}$, $\nexists M_C \in \mathcal{M}^{\star}_{FIM}$, *s.t.* $M_A \geq M_C$ and $M_A \neq M_C$.

Definition 9. *Set $\mathcal{M}^{\star}_{L} \subseteq \mathcal{M}_{L}$. $\mathcal{M}^{\star}_{L}$ is called the minimal covered set of $\mathcal{M}_{L}$ if the following two conditions are satisfied:*

1) $\forall M_P \in \mathcal{M}_L$, $\exists M_Q \in \mathcal{M}^{\star}_{L}$, *s.t.* $M_Q \geq M_P$;

2) $\forall M_P \in \mathcal{M}^{\star}_{L}$, $\nexists M_Q \in \mathcal{M}^{\star}_{L}$, *s.t.*$M_Q \geq M_P$ and $M_P \neq M_Q$.

According to Definitions 8 and 9, if all $\mathcal{M}_{FIM}$ are forbidden but all $\mathcal{M}_L$ are not, the system will reach optimal control.

3.4. Pre Idle Places (PIP)

Based on Part A in Section 3, all places in a PNM can be divided into three groups: P^0 (idle places), P_A (operation places) and P_R (resource places), and just the P_A is needed to identify Pre Idle Places (PIP). To reduce the computational cost of conventional MFFP, in this paper the authors give two definitions: Pre Idle Transitions (PIT, T_{PIT}) and Pre Idle Places (PIP, P_{PIP}). Once any PIP is found and identified in a PN model, the computational time of the MFFP deadlock prevention policy can then be reduced. In other words, the MFFP's computational efficiency can be enhanced further.

Definition 10. *Let all pre idle transitions be denoted as* $T_{PIT} = \{t \in T,\ {}^{\bullet}t \in P_A \wedge t^{\bullet} \in P^0 \wedge\ t^{\bullet} \in P_R)\}$.

Definition 11. *Let pre idle places be denoted as* $P_{PIP} = \{p \in P_A, p^{\bullet} \in T_{PIT}\}$.

According to Definitions 10 and 11, one can make sure that a PNM has $\{T_{PIT}, P_{PIP}\}$ or not. For example, in Figure 1, the PNM has $\{T_{PIT}, P_{PIP}\}$ in it. The detailed information is $T_{PIT} = \{t_4, t_8\}$ and $P_{PIP} = \{p_4, p_7\}$. Therefore, when the MFFP deadlock prevention policy is used to solve Integer Linear Programming Problems (ILPP), just four idle places $\{p_2, p_3, p_5, p_6\}$ (i.e., 6 (total idle places) - 2 (P_{PIP}) = 4) need to be considered. Please note that there are six idle places in total that need to be considered under the conventional MFFP deadlock prevention.

3.5. The Proposed Improved MFFP-2 (IMFFP-2)

This section presents the full Improved MFFP-2 (IMFFP-2) based on the definitions in Section 3. By using constraints (5) and (6), one can hold all legal markings and forbid all illegal markings.

IMFFP:

$max f^* = \sum f_k$

S.t.

$\sum l_i{\cdot}M_l(p_i) \le \beta,\ \forall M_l \in \mathcal{M}_L^{\star}$

$\sum l_i{\cdot}M_k(p_i) > \beta - Q(1 - f_k),\ \forall M_k \in \mathcal{M}_{FBM}^{\star}$

$l_i \in \mathbb{N},\ \ \forall i \in N_A, \forall i\ N_{PIP}$

$\beta \in \mathbb{N}.$

$f_k \in \{0,\ 1\}.$

Q is a huge positive integer constant.

Under the improved MFFP-2 (IMFFP-2), $\mathcal{M}_{FBM}^{\star}$ and $\mathcal{M}_L^{\star}$ are also listed to determine all constraints and to obtain the system's maximally permissive controllers. Only one set of controllers is processed by every iteration of IMFFP-2. Please note that not all controllers can be obtained just through one iteration.

4. Examples

The three classical PN models shown in Figure 3, Figures 7 and 8 are used to evaluate and check our proposed IMFFP-2 in this section, based on two famous analysis software PN-Tools [53] and INA [54].

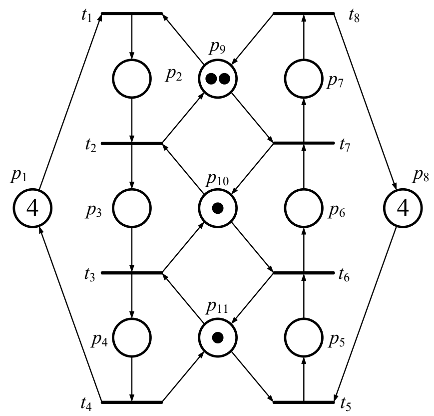

Figure 3. Another simple PN model of FMS.

The first model, shown in Figure 3, has two idle places (i.e., p_1 and p_8), six activity places (i.e., p_2-p_7), and three resource places (i.e., p_9-p_{11}). Additionally, there are 44 reachable markings (i.e., M_1-M_{44}) in total, including 8 FBMs (i.e., M_9, M_{11}, M_{17}, M_{18}, M_{23}, M_{25}, M_{30}, and M_{37}) and 36 legal markings in its reachability graph, respectively. Its reachability graph is also shown in Figure 4.

According to our proposed algorithm in Section 3, T_{PIT} and P_{PIP} must be identified first. The computational efficiency of the conventional MFFP can then be further enhanced, based on Definitions 5 and 6, and the $T_{PIT} = \{t_4,\ t_8\}$ and $P_{PIP} = \{p_4,\ p_7\}$ in Example 1.

Next, following the vector covering approach, one can obtain $\mathcal{M}_L^{\star} = \{M_{36}, M_{38}, M_{43}, M_{44}\}$ = {($2p_2$ + p_3 + p_4), (p_2 + p_5 + p_6 + p_7), (p_2 + p_3 + p_4 + p_7), (p_5 + p_6 + $2p_7$)} and $\mathcal{M}_{FBM}^{\star} = \{M_9, M_{11}, M_{18}\}$ = {($2p_2$ + p_5), (p_3 + p_5), ($2p_2$ + p_6)}, respectively. Please note that all the above markings present partial information. The detailed information is shown in Tables 2 and 3.

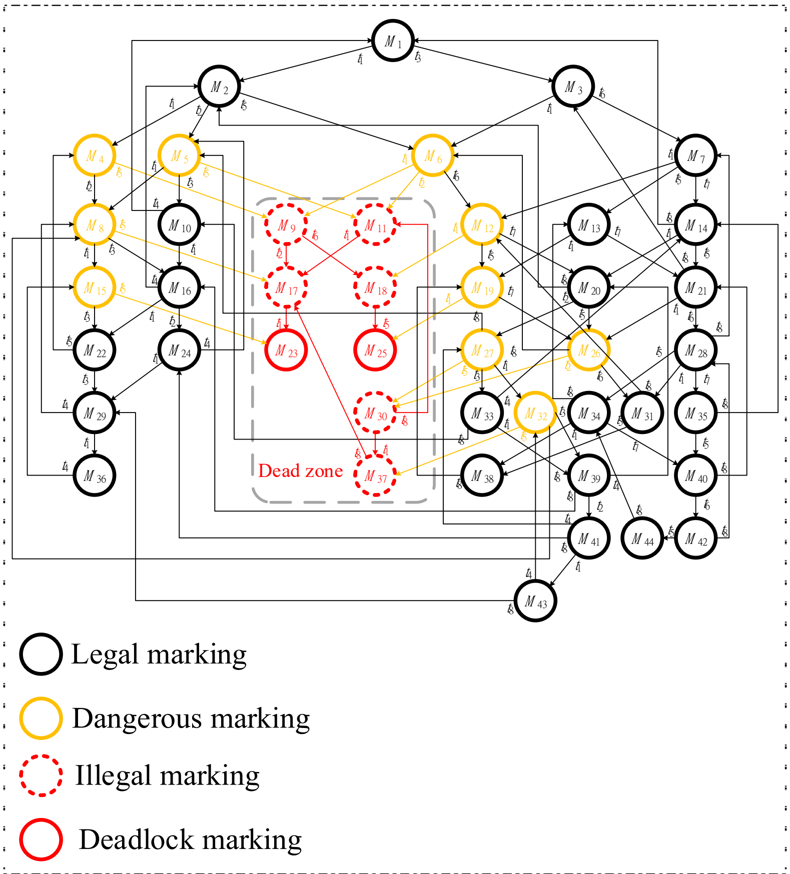

Figure 4. The initial and full reachability graph of Figure 3.

Table 2. Properties of eight First Bad Markings.

Marking No.	Classification	Marking Information $[p_1,p_2,p_3,p_4,p_5,p_6,p_7,p_8,p_9,p_{10},p_{11}]^T$
M_9	$M_Q/\mathcal{M}^{\star}_{FBM}$	$[2,\ 2,\ 0,\ 0,\ 1,\ 0,\ 0,\ 3,\ 0,\ 1,\ 0]^T$
M_{11}	$M_Q/\mathcal{M}^{\star}_{FBM}$	$[3,\ 0,\ 1,\ 0,\ 1,\ 0,\ 0,\ 3,\ 2,\ 0,\ 0]^T$
M_{17}	$M_Q/\mathcal{M}_{FBM}$	$[2,\ 1,\ 1,\ 0,\ 1,\ 0,\ 0,\ 3,\ 1,\ 0,\ 0]^T$
M_{18}	$M_Q/\mathcal{M}^{\star}_{FBM}$	$[2,\ 2,\ 0,\ 0,\ 0,\ 1,\ 0,\ 3,\ 0,\ 0,\ 1]^T$
M_{23}	$M_D/\mathcal{M}_{FBM}$	$[1,\ 2,\ 1,\ 0,\ 1,\ 0,\ 0,\ 3,\ 0,\ 0,\ 0]^T$
M_{25}	$M_D/\mathcal{M}_{FBM}$	$[2,\ 2,\ 0,\ 0,\ 1,\ 1,\ 0,\ 2,\ 0,\ 0,\ 0]^T$
M_{30}	$M_Q/\mathcal{M}_{FBM}$	$[3,\ 0,\ 1,\ 0,\ 1,\ 0,\ 1,\ 2,\ 1,\ 0,\ 0]^T$
M_{37}	$M_Q/\mathcal{M}_{FBM}$	$[2,\ 1,\ 1,\ 0,\ 1,\ 0,\ 1,\ 2,\ 0,\ 0,\ 0]^T$

Table 3. Properties of four Minimal Covering Sets.

Marking No.	Classification	Marking Information $[P_1,p_2,p_3,p_4,p_5,p_6,p_7,p_8,p_9,p_{10},p_{11}]^T$
M_{36}	$\mathcal{M}^{\star}_L$	$[0,2,1,1,0,0,0,4,0,0,0]^T$
M_{38}	$\mathcal{M}^{\star}_L$	$[3,1,0,0,1,1,1,0,0,0]^T$
M_{43}	$\mathcal{M}^{\star}_L$	$[1,1,1,1,0,0,1,3,0,0,0]^T$
M_{44}	$\mathcal{M}^{\star}_L$	$[4,0,0,0,1,1,2,0,0,0,0]^T$

Further, the IMFFP-2 for this model can then be obtained as follows.
The first iteration in our IMFFP-2:

$max\ f^* = f_1 + f_2 + f_3$ S. t.
$2l_2 + l_3 \le \beta$
$l_2 + l_5 + l_6 \le \beta$
$l_2 + l_3 \le \beta$
$l_5 + l_6 \le \beta$
$2l_2 + l_5 > \beta - Q(1 - f_1)$
$l_3 + l_5 > \beta - Q(1 - f_2)$
$2l_2 + l_6 > \beta - Q(1 - f_3)$
$l_i \in \mathbb{N},\ \ \forall i \in \mathbb{N}_A.$
$\beta \in \mathbb{N}.$
$f_k \in \{0,\ 1\}.$

In the following, solving the ILPP by using our proposed IMFFP-2, one can obtain two solutions. The first is $f_1 = 1$, $f_2 = 1$, $l_2 = 1$, $l_3 = 2$, $l_5 = 3$, and $\beta = 4$, and the second is $f_2 = 1$, $f_3 = 1$, $l_2 = 2$, $l_5 = 1$, $l_6 = 1$, and $\beta = 4$.

Note that in fact the above two sets of controllers are the same since they can both make the system maximally permissive. Accordingly, one can just pick one set of controllers to process. For showing the two controllers are the same and the difference between this paper and [1], a second set of controllers is chosen by this paper. The second iteration is as follows.

The second iteration in the Improved MFFP-2:
$max f^* = f_1$ S. t.
$2l_2 + l_3 \le \beta$
$l_2 + l_5 + l_6 \le \beta$
$l_2 + l_3 \le \beta$
$l_5 + l_6 + 2l_7 \le \beta$
$2l_2 + l_5 > \beta - Q(1 - f_1)$
$l_i \in \mathbb{N},\ \ \forall i \in \mathbb{N}_A.$
$\beta \in \mathbb{N}.$
$f_k \in \{0,\ 1\}.$

By running the above IMFFP-2 again, one new controller is obtained such that $l_3 = 1$, $l_5 = 1$, $\beta = 1$, and $f_1 = 1$. Therefore, two PI controllers, $pc_1\ : 2\mu_2 + \mu_5 + \mu_6 \le 4$ and $pc_2\ : \mu_3 + \mu_5 \le 1$, are obtained. By putting the two control places into the original model (please refer to Figure 3), all 8 First Bad Markings are controlled while all 36 legal reachable markings are held. In other words, the PNM is not only live but also maximally permissive. Figures 5 and 6 show the controlled system and its reachability graph, respectively.

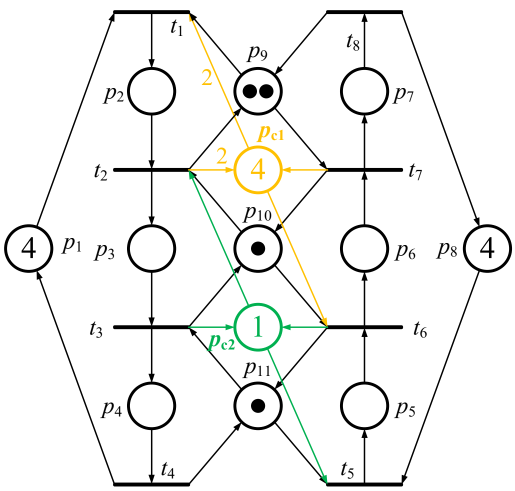

Figure 5. Two controllers are added into the PN model in Figure 3.

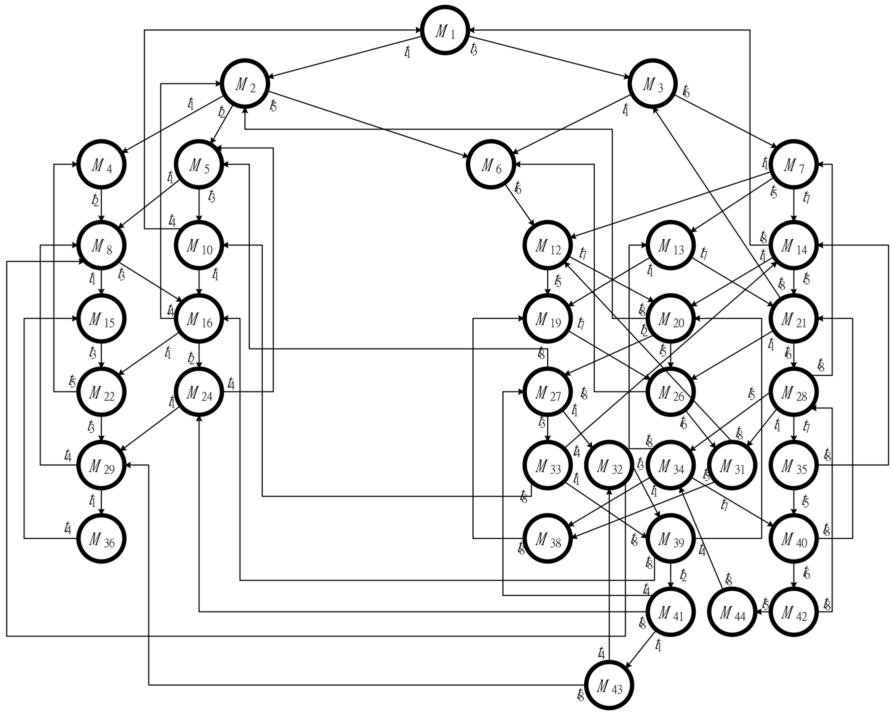

Figure 6. The maximally permissive reachable states of Figure 3.

The second PNM of the FMS model shown in Figure 7 has 282 reachable markings in total, including and 205 legal markings and 77 illegal markings. Besides, there are 54 FBMs in the RG of the PNM according to the definition. By the vector covering approach, the markings that need to be considered in computation can then be reduced to 8 $\mathcal{M}^{\star}_{FBM}$ and 26 $\mathcal{M}^{\star}_{L}$, respectively. Due to the limitation of space of this paper, the detailed information is presented in Tables 4 and 5, respectively.

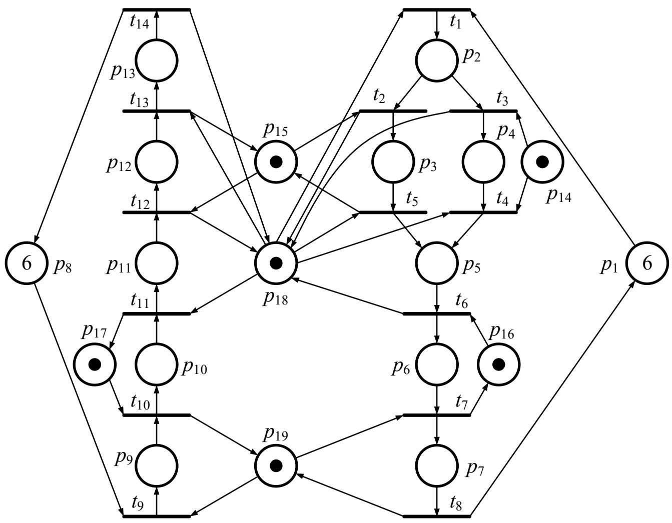

Figure 7. The second PN model of FMS.

Table 4. The properties of all First Bad Markings in Example 2.

Marking No.	Classification	Marking Information $[p1 \sim p19]^T$	Marking No.	Classification	Marking Information $[p1 \sim p19]^T$
M_6	$M_Q/\mathcal{M}^{\star}_{FBM}$	[3 1 1 1 0 0 0 6 0 0 0 0 0 0 0 1 1 0 1]	M_{136}	M_Q	[4 1 0 1 0 0 0 4 0 1 0 1 0 0 0 1 0 0 1]
M_7	M_Q	[3 1 1 1 0 0 0 5 1 0 0 0 0 0 0 1 1 0 0]	M_{141}	M_Q	[3 1 0 1 0 1 0 4 1 0 0 1 0 0 0 0 1 0 0]
M_8	M_Q	[3 1 1 1 0 0 0 5 0 1 0 0 0 0 0 1 0 0 1]	M_{142}	M_Q	[3 1 0 1 0 1 0 4 0 1 0 1 0 0 0 0 0 0 1]
M_9	M_D	[3 1 1 1 0 0 0 4 1 1 0 0 0 0 0 1 0 0 0]	M_{143}	M_Q	[3 1 0 1 0 0 1 4 0 1 0 1 0 0 0 1 0 0 0]
M_{14}	M_Q	[2 1 1 1 0 1 0 6 0 0 0 0 0 0 0 0 1 0 1]	M_{150}	M_Q	[2 1 0 1 0 1 1 4 0 1 0 1 0 0 0 0 0 0 0]
M_{15}	M_Q	[2 1 1 1 0 0 1 6 0 0 0 0 0 0 0 1 1 0 0]	M_{162}	$M_Q/\mathcal{M}^{\star}_{FBM}$	[6 0 0 0 0 0 0 4 0 0 1 1 0 1 0 1 1 0 1]
M_{16}	M_Q	[2 1 1 1 0 1 0 5 1 0 0 0 0 0 0 0 1 0 0]	M_{198}	$M_Q/\mathcal{M}^{\star}_{FBM}$	[4 1 0 1 0 0 0 5 0 0 0 1 0 0 0 1 1 0 1]
M_{17}	M_Q	[2 1 1 1 0 1 0 5 0 1 0 0 0 0 0 0 0 0 1]	M_{203}	M_Q	[3 1 0 1 0 1 0 5 0 0 0 1 0 0 0 0 1 0 1]
M_{18}	M_Q	[2 1 1 1 0 0 1 5 0 1 0 0 0 0 0 1 0 0 0]	M_{204}	M_Q	[3 1 0 1 0 0 1 5 0 0 0 1 0 0 0 1 1 0 0]
M_{25}	M_Q	[1 1 1 1 0 1 1 6 0 0 0 0 0 0 0 0 1 0 0]	M_{210}	M_Q	[2 1 0 1 0 1 1 5 0 0 0 1 0 0 0 0 1 0 0]
M_{48}	M_Q	[1 1 1 1 0 1 1 5 0 1 0 0 0 0 0 0 0 0 0]	M_{250}	M_Q	[4 0 1 0 0 0 1 5 0 0 1 0 0 1 0 1 1 0 0]
M_{66}	$M_Q/\mathcal{M}^{\star}_{FBM}$	[4 0 1 0 1 0 0 4 1 1 0 0 0 1 0 1 0 0 0]	M_{251}	$M_Q/\mathcal{M}^{\star}_{FBM}$	[5 0 1 0 0 0 0 5 0 0 1 0 0 1 0 1 1 0 1]
M_{67}	$M_Q/\mathcal{M}^{\star}_{FBM}$	[4 0 1 0 0 1 0 4 1 1 0 0 0 1 0 0 0 1 0]	M_{257}	M_Q	[5 0 0 0 0 0 1 4 0 0 1 1 0 1 0 1 1 0 0]
M_{68}	M_Q	[3 1 1 0 0 1 0 4 1 1 0 0 0 1 0 0 0 0 0]	M_{258}	M_Q	[5 0 0 0 0 1 0 4 0 0 1 1 0 1 0 0 1 0 1]
M_{69}	M_Q	[3 0 1 1 0 1 0 4 1 1 0 0 0 0 0 0 0 1 0]	M_{259}	M_Q	[5 0 0 1 0 0 0 4 0 0 1 1 0 0 0 1 1 0 1]
M_{70}	M_D	[3 0 1 0 1 1 0 4 1 1 0 0 0 1 0 0 0 0 0]	M_{260}	M_Q	[5 0 0 1 0 0 0 3 1 0 1 1 0 0 0 1 1 0 0]
M_{71}	M_D	[3 0 0 1 1 1 0 4 1 1 0 0 0 0 1 0 0 0 0]	M_{261}	M_Q	[4 0 0 1 0 0 1 4 0 0 1 1 0 0 0 1 1 0 0]
M_{78}	$M_D/\mathcal{M}^{\star}_{FBM}$	[4 0 0 0 1 1 0 4 1 1 0 0 0 1 1 0 0 0 0]	M_{262}	M_Q	[4 0 0 1 0 1 0 4 0 0 1 1 0 0 0 0 1 0 1]
M_{87}	$M_Q/\mathcal{M}^{\star}_{FBM}$	[3 1 0 1 0 1 0 4 1 1 0 0 0 0 1 0 0 0 0]	M_{263}	M_Q	[3 0 0 1 0 1 1 4 0 0 1 1 0 0 0 0 1 0 0]
M_{94}	M_D	[4 1 0 1 0 0 0 3 1 1 0 1 0 0 0 1 0 0 0]	M_{265}	M_Q	[4 0 0 0 0 1 1 4 0 0 1 1 0 1 0 0 1 0 0]
M_{99}	M_D	[3 1 0 1 0 1 0 3 1 1 0 1 0 0 0 0 0 0 0]	M_{269}	M_Q	[4 0 1 1 0 0 0 4 1 0 1 0 0 0 0 1 1 0 0]
M_{100}	M_D	[4 0 0 0 1 1 0 3 1 1 0 1 0 1 0 0 0 0 0]	M_{270}	M_Q	[4 0 1 0 0 1 0 5 0 0 1 0 0 1 0 0 1 0 1]
M_{101}	M_Q	[4 0 0 1 0 1 0 3 1 0 1 1 0 0 0 0 1 0 0]	M_{272}	M_Q	[4 0 1 1 0 0 0 5 0 0 1 0 0 0 0 1 1 0 1]
M_{108}	M_Q	[5 0 0 0 0 1 0 3 1 0 1 1 0 1 0 0 1 0 0]	M_{273}	M_Q	[3 0 1 1 0 0 1 5 0 0 1 0 0 0 0 1 1 0 0]
M_{124}	M_Q	[6 0 0 0 0 0 0 3 1 0 1 1 0 1 0 1 1 0 0]	M_{274}	M_Q	[3 0 1 1 0 1 0 5 0 0 1 0 0 0 0 0 1 0 1]
M_{130}	M_Q	[5 0 1 0 0 0 0 4 1 0 1 0 0 1 0 1 1 0 0]	M_{277}	M_Q	[2 0 1 1 0 1 1 5 0 0 1 0 0 0 0 0 1 0 0]
M_{135}	M_Q	[4 1 0 1 0 0 0 4 1 0 0 1 0 0 0 1 1 0 0]	M_{278}	M_Q	[3 0 1 0 0 1 1 5 0 0 1 0 0 1 0 0 1 0 0]

Table 5. The properties of 26 Minimal Covering Sets.

Marking No.	**Information of Marking** $[\boldsymbol{p}_1 \sim \boldsymbol{p}_{19}]^T$
M_{268}	[4 0 0 1 0 1 0 3 1 1 1 0 0 0 1 0 0 0 0]
M_{97}	[4 1 0 0 0 1 0 3 1 1 0 1 0 1 0 0 0 0 0]
M_{98}	[4 0 0 1 0 1 0 3 1 1 0 1 0 0 0 0 0 1 0]
M_{46}	[2 1 1 0 0 1 1 5 0 1 0 0 0 1 0 0 0 0 0]
M_{54}	[2 1 0 1 0 1 1 5 0 1 0 0 0 0 1 0 0 0 0]
M_{107}	[4 0 0 1 0 1 0 3 1 1 0 0 1 0 1 0 0 0 0]
M_{47}	[2 0 1 1 0 1 1 5 0 1 0 0 0 0 0 0 0 1 0]
M_{49}	[2 0 1 0 1 1 1 5 0 1 0 0 0 1 0 0 0 0 0]
M_{50}	[2 0 0 1 1 1 1 5 0 1 0 0 0 0 1 0 0 0 0]
M_{148}	[3 1 0 0 0 1 1 4 0 1 0 1 0 1 0 0 0 0 0]
M_{149}	[3 0 0 1 0 1 1 4 0 1 0 1 0 0 0 0 0 1 0]
M_{151}	[3 0 0 0 1 1 1 4 0 1 0 1 0 1 0 0 0 0 0]
M_{264}	[3 0 0 1 0 1 1 4 0 1 0 0 1 0 1 0 0 0 0]
M_{64}	[4 1 1 0 0 0 0 4 1 1 0 0 0 1 0 1 0 0 0]
M_{132}	[4 1 0 1 0 0 0 4 1 1 0 0 0 0 1 1 0 0 0]
M_{65}	[4 0 1 1 0 0 0 4 1 1 0 0 0 0 0 1 0 1 0]
M_{40}	[3 1 1 0 0 1 0 5 1 0 0 0 0 1 0 0 1 0 0]
M_{191}	[3 1 0 1 0 1 0 5 1 0 0 0 0 0 1 0 1 0 0]
M_{85}	[4 0 0 1 1 0 0 4 1 1 0 0 0 0 1 1 0 0 0]
M_{41}	[3 0 1 1 0 1 0 5 1 0 0 0 0 0 0 0 1 1 0]
M_{42}	[3 0 1 0 1 1 0 5 1 0 0 0 0 1 0 0 1 0 0]
M_{279}	[3 0 0 1 1 1 0 5 1 0 0 0 0 0 1 0 1 0 0]
M_{95}	[5 0 0 0 1 0 0 3 1 1 0 1 0 1 0 1 0 0 0]
M_{90}	[4 0 0 1 0 0 1 4 0 1 1 0 0 0 1 1 0 0 0]
M_{144}	[4 0 0 0 1 1 0 4 1 0 0 1 0 1 0 0 1 0 0]
M_{276}	[3 0 0 1 0 1 1 5 0 0 1 0 0 0 1 0 1 0 0]

Additionally, based on Definitions 5 and 6, $T_{PIT} = \{t_8,\ t_{14}\}$ and $P_{PIP} = \{p_7,\ p_{13}\}$ in Example 2. Therefore, the first iteration for this example, using our IMFFP-2, is as follows:

$max f^* = f_1 + f_2 + f_3 + f_4 + f_5 + f_6 + f_7 + f_8$

S. t.

$l_4 + l_6 + l_9 + l_{10} + l_{11} \leq \beta$

$l_2 + l_6 + l_9 + l_{10} + l_{12} \leq \beta$

$l_4 + l_6 + l_9 + l_{10} + l_{12} \leq \beta$

$l_2 + l_3 + l_6 + l_{10} \leq \beta$

$l_2 + l_4 + l_6 + l_{10} \leq \beta$

$l_4 + l_6 + l_9 + l_{10} \leq \beta$

$l_3 + l_4 + l_6 + l_{10} \leq \beta$

$l_3 + l_5 + l_6 + l_{10} \leq \beta$

$l_4 + l_5 + l_6 + l_{10} \leq \beta$

$l_2 + l_6 + l_{10} + l_{12} \leq \beta$

$l_4 + l_6 + l_{10} + l_{12} \leq \beta$

$l_5 + l_6 + l_{10} + l_{12} \leq \beta$

$l_2 + l_3 + l_9 + l_{10} \leq \beta$

$l_2 + l_4 + l_9 + l_{10} \leq \beta$

$l_3 + l_4 + l_9 + l_{10} \leq \beta$

$l_2 + l_3 + l_6 + l_9 \leq \beta$

$l_2 + l_4 + l_6 + l_9 \leq \beta$

$l_4 + l_5 + l_9 + l_{10} \leq \beta$

$l_3 + l_4 + l_6 + l_9 \leq \beta$

$l_3 + l_5 + l_6 + l_9 \leq \beta$

$l_4 + l_5 + l_6 + l_9 \leq \beta$
$l_5 + l_9 + l_{10} + l_{12} \leq \beta$
$l_5 + l_6 + l_9 + l_{12} \leq \beta$
$l_4 + l_{10} + l_{11} \leq \beta$
$l_4 + l_6 + l_{10} \leq \beta$
$l_4 + l_6 + l_{11} \leq \beta$
$l_3 + l_{11} > \beta - Q(1 - f_1)$
$l_{11} + l_{12} > \beta - Q(1 - f_2)$
$l_2 + l_3 + l_4 > \beta - Q(1 - f_3)$
$l_2 + l_4 + l_{12} > \beta - Q(1 - f_4)$
$l_3 + l_5 + l_9 + l_{10} > \beta - Q(1 - f_5)$
$l_3 + l_6 + l_9 + l_{10} > \beta - Q(1 - f_6)$
$l_5 + l_6 + l_9 + l_{10} > \beta - Q(1 - f_7)$
$l_2 + l_4 + l_6 + l_9 + l_{10} > \beta - Q(1 - f_8)$
$l_i \in \mathbb{N}, \ \ \forall i \in \mathbb{N}_A.$
$\beta \in \mathbb{N}.$
$f_k \in \{0, \ 1\}.$

In the first iteration, there is a solution that $l_2 = 4$, $l_3 = 8$, $l_4 = 4$, $l_5 = 5$, $l_6 = 0$, $l_7 = 0$, $l_9 = 1$, $l_{10} = 1$, $l_{11} = 8$, $l_{12} = 7$, $l_{13} = 0$, $\beta = 14$, and $f_1 \sim f_5 = 0$. In other words, the first maximally permissive solution is $pc_1 \ : 4\mu_2 + 8\mu_3 + 4\mu_4 + 5\mu_5 + \mu_9 + \mu_{10} + 8\mu_{11} + 7\mu_{12} \leq 14$, so five $\mathcal{M}^{\star}_{\text{FBM}}$ are hence forbidden. Then, the second iteration is considered, and a solution is obtained that $l_2 = 1$, $l_3 = 2$, $l_4 = 1$, $l_5 = 2$, $l_6 = 2$, $l_7 = 0$, $l_9 = 3$, $l_{10} = 3$, $l_{11} = 0$, $l_{12} = 0$, $l_{13} = 0$, $\beta = 9$, and $f_6 \sim f_8 = 0$. In other words, the second maximally permissive solution is $pc_2 \ : \mu_2 + 2\mu_3 + \mu_4 + 2\mu_5 + 2\mu_6 + 3\mu_9 + 3\mu_{10} \leq 9$. In the second iteration, the last three $\mathcal{M}^{\star}_{\text{FBM}}$ are forbidden. Finally, all eight $\mathcal{M}^{\star}_{\text{FBM}}$ are forbidden. Two controllers shown in Table 6 can then be obtained by the above two solutions. After adding the two control places into the second PN model, all the FBMs are then forbidden. Additionally, all illegal markings are also controlled. Two maximally permissive control places are shown in Figure 8.

Table 6. The two optimal controllers of Example 2.

Additional C_{p_i}	$M_0(C_{p_i})$	$\cdot(C_{p_i})$	$(C_{p_i})\cdot$
C_{p_1}	14	$3t_5,\ 5t_6,\ t_{12},\ 7t_{13}$	$4t_1,\ 4t_2,\ t_4,\ t_9,\ 7t_{11}$
C_{p_2}	9	$2t_7,\ 3t_{11}$	$t_1,\ t_2,\ t_4,\ 3t_9$

The third PN model taken from [6] is presented in Figure 9. Due to the limitation of space in this paper, the detailed information concerning the process of seeking controllers is not presented here. According to the definitions in this paper, there are 16 activity places that need to be calculated in MFFP. However, there are just 13 activity places that need to be calculated in IMFFP-2, since three PIPs (i.e., p_4, p_{10}, and p_{19}) can be identified in Example 3.

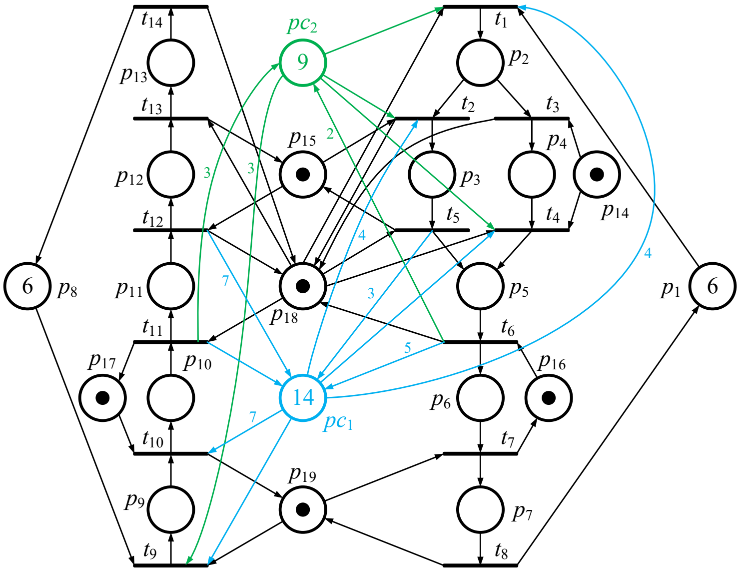

Figure 8. Controlled system of the Petri model in Figure 7.

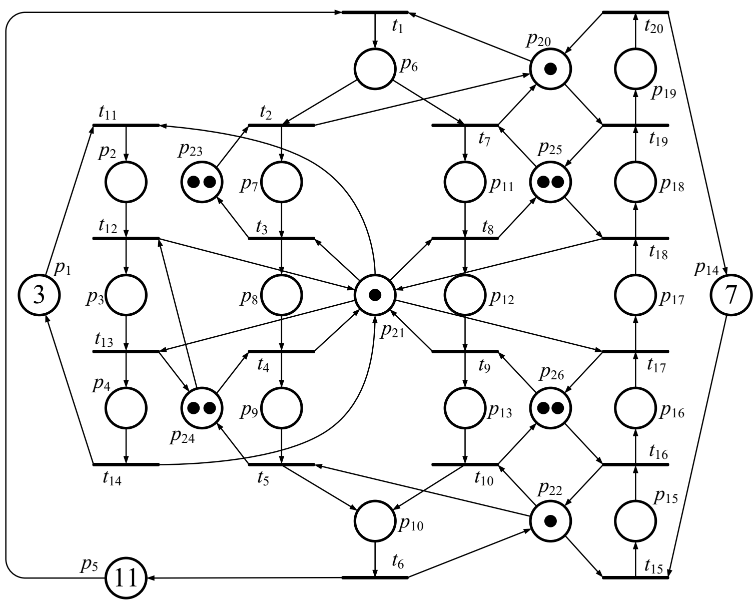

Figure 9. The third Petri net model of FMS.

5. Comparison

In this section, we make a comparison with the conventional MFFP [1,2] in terms of the computational efficiency by comparing (1) system complexity and (2) time complexity. Here, $|P_A|_{\text{MFFP}}$ represents the number of activity places needed by the MFFP method , $|P_{PIP}|$ represents the number of P_{PIP} identified in one PN-based system, and $|P_A|_{\text{IMFFP}}$ represents the number of activity places needed by the IMFFP-2 method. Further, $\text{O(n?)}_{\text{MFFP}}$ and $\text{O(n?)}_{\text{IMFFP}}$ represent time complexity under MFFP and IMFFP-2, respectively. Firstly, based on Table 7 one can realize that there are 6, 11, and 16 variables (i.e., activity places) in Examples 1, 2, and 3, respectively, that need to be considered under the conventional MFFP control policy. However, there are just 4, 9, and 12 variables to consider since there are 2, 2, and 4 P_{PIP} identified in Examples 1, 2, and 3, respectively, under IMFFP-2.

Table 7. Comparison using system/time complexity in different examples.

Ex.	$\lvert P_A\rvert_{MFFP}$	$\lvert P_A\rvert_{IMFFP\text{-}2}$	$\lvert P_A\rvert_{MFFP} - \lvert P_A\rvert_{IMFFP\text{-}2}$	$O(n?)_{MFFP}$	$O(n?)_{IMFFP\text{-}2}$	$O(n?)_{MFFP}$ / $O(n?)_{IMFFP\text{-}2}$	Computational Efficiency
1	6	4	2	n^6	n^4	n^6/n^4	100 times
2	11	9	2	n^{11}	n^9	n^{11}/n^9	100 times
3	16	13	3	n^{16}	n^{13}	n^{16}/n^{13}	1000 times

Firstly, from the viewpoint of system complexity, the computational efficiency of IMFFP-2 must be better than conventional MFFP since the number of variables of IMFFP-2 is much less than in the conventional MFFP method. On the other hand, according to the book *Introduction to the theory of computation* (second edition) [55] and Wikipedia [56], the definition of time complexity for computer science is as follows: the time complexity of an algorithm can be expressed commonly by using big O notation "O", which excludes coefficients and lower-order terms. When expressed this way, time complexity is said to be described asymptotically, i.e., as the input size goes to infinity. For instance, if the time required by an algorithm on all inputs of size n is at most $6n^3 + 2n^2 + 20n + 45$ for any n (n is a variable), then obviously, there are four terms in total and the highest order term is $6n^3$. Thus, the asymptotic time complexity is $O(n^3)$. Therefore, the time complexity of Chen's MFFP method in Example 1 could be viewed as $O(n^6)$. The time complexity of our IMFFP-2 is viewed as $O(n^4)$ based on Table 7. Similarly, the time complexity of Chen's MFFP method in Example 2 is viewed as $O(n^{11})$, and our IMFFP-2 is $O(n^9)$; the time complexity of Chen's MFFP method in Example 3 is $O(n^{16})$, and our IMFFP-2 is $O(n^{13})$. Accordingly, in Examples 1 and 2, the computational efficiency of the proposed IMFFP-2 is 100 times better than that of the conventional MFFP control policy. In addition, in Example 3, the computational efficiency of the proposed IMFFP-2 is 1000 times better than that of the conventional MFFP method.

6. Conclusions

Solving flexible manufacturing systems' deadlock problems is extremely difficult work. Just a few excellent mathematicians or researchers can work in this field. For decades, these outstanding pioneers have concentrated their efforts into proposing various algorithms to solve the deadlock problems of flexible manufacturing systems. In the authors' previous study, the conventional MFFP method proposed by Chen et al. [1] seemed to be the best one (i.e., least control places and relative arcs) among all control place-based deadlock prevention policies in the existing literature. The greatest advantage of conventional MFFP is that just a few controllers are needed to hold the maximally permissive states in solving deadlock problems. In fact, the method could become better if (1) it could be applied for all kinds of nets not just for certain special nets or (2) we could further enhance its computational efficiency. Based on the above reason, firstly, in our previous paper we proposed an improved iteration method called IMFFP-1 [42] so that it could be used for all Petri nets' general systems.

In the present paper, we further propose two novel concepts, namely, PIT and PIP, for the IMFFP-2. Once PIT or PIP is identified from a deadlock Petri nets model, one can bypass all PIP under the process of solving MFFP, and the computational time can hence be shortened. Undoubtedly, computational cost can then be decreased. Three examples prove that the IMFFP-2 is more efficient than the conventional MFFP in obtaining the same maximally permissive control places. Finally, and most importantly, in this paper, we redefine the conventional vector covering approach technology, named the improved vector covering approach technology (IVCAT), so that the definition can be solidified. We also identify the correct solution in the second example of Chen's paper [1] so that all readers and researchers can easily follow Chen's paper. In summary, the IMFFP-1 makes the conventional MFFP able to be used in all kinds of nets, and IMFFP-2 enhances the computational efficiency. In the future, we will further merge IMFFP-1 and IMFFP-2 so that it can be used for all kinds of PN-based FMSs and still have very great computational efficiency.

In the past, our research team has focused on developing more efficient algorithms in the existing methods, and obtained quite good results [9,15,34–37,40,42,43,47–49,57]. In our future works, four novel technologies [46,49–51] in this domain will be considered and we will aim to enhance their computational efficiency.

Author Contributions: Conceptualization, Y.-L.P. and C.-Y.T.; methodology, Y.-L.P., C.-W.T., and C.-Y.T.; software, C.-Y.T.; validation, Y.-L.P.; formal analysis, Y.-L.P.; investigation, Y.-L.P. and C.-W.T.; resources, C.-W.T. and J.-C.H.; data curation, Y.-L.P. and C.-Y.T.; writing—original draft preparation, Y.-L.P.; writing—review and editing, Y.-L.P.; visualization, C.-W.T. and J.-C.H.; supervision, Y.-L.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Chen, Y.; Li, Z.; Zhou, M. Behaviorally Optimal and Structurally Simple Liveness-Enforcing Supervisors of Flexible Manufacturing Systems. *IEEE Trans. Syst. Man Cybern. Part A Syst. Hum.* **2012**, *42*, 615–629. [CrossRef]
2. Chen, Y.; Li, Z.; Khalgui, M.; Mosbahi, O. Design of a Maximally Permissive Liveness- Enforcing Petri Net Supervisor for Flexible Manufacturing Systems. *IEEE Trans. Autom. Sci. Eng.* **2011**, *8*, 374–393. [CrossRef]
3. Yamalidou, K.; Moody, J.; Lemmon, M.; Antsaklis, P. Feedback control of petri nets based on place invariants. *Automatica* **1996**, *32*, 15–28. [CrossRef]
4. Coffman, E.G.; Elphick, M.; Shoshani, A. System Deadlocks. *ACM Comput. Surv.* **1971**, *3*, 67–78. [CrossRef]
5. Uzam, M.; Zhou, M. An Iterative Synthesis Approach to Petri Net-Based Deadlock Prevention Policy for Flexible Manufacturing Systems. *IEEE Trans. Syst. Man Cybern. Part A Syst. Hum.* **2007**, *37*, 362–371. [CrossRef]
6. Ezpeleta, J.; Colom, J.; Martinez, J. A Petri net based deadlock prevention policy for flexible manufacturing systems. *IEEE Trans. Robot. Autom.* **1995**, *11*, 173–184. [CrossRef]
7. Kumaran, T.K.; Chang, W.; Cho, H.; Wysk, R.A. A structured approach to deadlock detection, avoidance and resolution in flexible manufacturing systems. *Int. J. Prod. Res.* **1994**, *32*, 2361–2379. [CrossRef]
8. Uzam, M.; Zhou, M.C. An improved iterative synthesis method for liveness enforcing supervisors of flexible manufacturing systems. *Int. J. Prod. Res.* **2006**, *44*, 1987–2030. [CrossRef]
9. Pan, Y.L.; Huang, Y.S.; Jeng, M.; Chung, S.L. Enhancement of an efficient control policy for FMSs using the theory of regions and selective siphon method. *Int. J. Adv. Manuf. Technol.* **2012**, *66*, 1805–1815. [CrossRef]
10. Luo, J.; Wu, W.; Su, H.; Chu, J. Supervisor Synthesis for Enforcing a Class of Generalized Mutual Exclusion Constraints on Petri Nets. *IEEE Trans. Syst. Man Cybern. Part A Syst. Hum.* **2009**, *39*, 1237–1246.
11. Chen, Y.; Li, Z. Design of a maximally permissive liveness-enforcing supervisor with a compressed supervisory structure for flexible manufacturing systems. *Automatica* **2011**, *47*, 1028–1034. [CrossRef]
12. Zhou, M.; Dicesare, F. Parallel and sequential mutual exclusions for petri net modeling of manufacturing systems with shared resources. *IEEE Trans. Robot. Autom.* **1991**, *7*, 515–527. [CrossRef]
13. Huang, Y.; Jeng, M.; Xie, X.; Chung, S. Deadlock prevention policy based on Petri nets and siphons. *Int. J. Prod. Res.* **2001**, *39*, 283–305. [CrossRef]
14. Huang, Y.S.; Jeng, M.; Xie, X.; Chung, D.H. Siphon-Based Deadlock Prevention Policy for Flexible Manufacturing Systems. *IEEE Trans. Syst. Man Cybern. Part A Syst. Hum.* **2006**, *36*, 1248–1256. [CrossRef]
15. Pan, Y.L.; Jeng, M.D.; Chung, S.L.; Guo, Y.X. Computationally Improved Optimal Deadlock Prevention Policy for Linear Programming Problems of Flexible Manufacturing Systems. In Proceedings of the 2013 IEEE International Conference on Systems, Man, and Cybernetics, Manchester, UK, 13–16 October 2013.
16. Zhou, M.; Mcdermott, K.; Patel, P. Petri net synthesis and analysis of a flexible manufacturing system cell. *IEEE Trans. Syst. Man Cybern.* **1993**, *23*, 523–531. [CrossRef]
17. Li, Z.; Wang, A.; Wei, N. Liveness-Enforcing Supervisors for Flexible Manufacturing Systems with Multiple Resource Acquisitions. In Proceedings of the 2006 IEEE International Conference on Networking, Sensing and Control, Ft. Lauderdale, FL, USA, 23–25 April 2006.
18. Li, Z.; Zhou, M. Elementary Siphons of Petri Nets and Their Application to Deadlock Prevention in Flexible Manufacturing Systems. *IEEE Trans. Syst. Man Cybern. Part A Syst. Hum.* **2004**, *34*, 38–51. [CrossRef]
19. Iordache, M.; Moody, J.; Antsaklis, P. Synthesis of deadlock prevention supervisors using Petri nets. *IEEE Trans. Robot. Autom.* **2002**, *18*, 59–68. [CrossRef]

20. Moody, J.O.; Antsaklis, P.J. *Supervisory Control of Discrete Event Systems Using Petri Nets*; Kluwer Academic Publishers: Boston, MA, USA, 1998.
21. Jeng, M.D. A Petri net synthesis theory for modeling flexible manufacturing systems. *IEEE Trans. Syst. Man Cybern. Part B (Cybern.)* **1997**, *27*, 169–183. [CrossRef]
22. Huang, Y.S. Design of Deadlock Prevention Supervisors Using Elementary Siphons. In Proceedings of the Multi Conference on "Computational Engineering in Systems Applications", Beijing, China, 4–6 October 2006.
23. Huang, Y.S. Deadlock Prevention for Sequence Resource Allocation Systems. *J. Inf. Sci. Eng.* **2007**, *23*, 215–231.
24. Li, Z.W.; Hu, H.S.; Wang, A.R. Design of Liveness-Enforcing Supervisors for Flexible Manufacturing Systems Using Petri Nets. *IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.)* **2007**, *37*, 517–526. [CrossRef]
25. Iordache, M.; Moody, J.; Antsaklis, P. A method for the synthesis of liveness enforcing supervisors in Petri nets. In Proceedings of the 2001 American Control Conference. (Cat. No. 01CH37148), Arlington, VA, USA, 25–27 June 2001.
26. Park, J.; Reveliotis, S. Algebraic synthesis of efficient deadlock avoidance policies for sequential resource allocation systems. *IEEE Trans. Robot. Autom.* **2000**, *16*, 190–195. [CrossRef]
27. Li, Z.W.; Liu, G.Y.; Hanisch, M.H.; Zhou, M.C. Deadlock prevention based on structure reuse of petri net supervisors for flexible manufacturing systems. *IEEE Trans. Syst. Man Cybern. Part A Syst. Hum.* **2012**, *42*, 178–191. [CrossRef]
28. Uzam, M. An Optimal Deadlock Prevention Policy for Flexible Manufacturing Systems Using Petri Net Models with Resources and the Theory of Regions. *Int. J. Adv. Manuf. Technol.* **2002**, *19*, 192–208. [CrossRef]
29. Uzam, M.; Zhou, M. Iterative synthesis of Petri net based deadlock prevention policy for flexible manufacturing systems. In Proceedings of the 2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No.04CH37583), The Hague, The Netherlands, 10–13 October 2004.
30. Chen, Y.; Li, Z.; Barkaoui, K. Optimal Petri net supervisor with lowest implemental cost for flexible manufacturing systems. In Proceedings of the Etfa2011, Toulouse, France, 5–9 September 2011.
31. Uzam, M. The use of the Petri net reduction approach for an optimal deadlock prevention policy for flexible manufacturing systems. *Int. J. Adv. Manuf. Technol.* **2004**, *23*, 204–219. [CrossRef]
32. Uzam, M. Synthesis of feedback control elements for discrete event systems using Petri net models and theory of regions. *Int. J. Adv. Manuf. Technol.* **2004**, *24*, 48–69. [CrossRef]
33. Ghaffari, A.; Rezg, N.; Xie, X. Design of a live and maximally permissive petri net controller using the theory of regions. *IEEE Trans. Robot. Autom.* **2003**, *19*, 137–142. [CrossRef]
34. Huang, Y.S.; Pan, Y.L. Enhancement of an Efficient Liveness-Enforcing Supervisor for Flexible Manufacture Systems. *Int. J. Adv. Manuf. Technol.* **2010**, *48*, 725–737. [CrossRef]
35. Huang, Y.S.; Pan, Y.L.; Zhou, M. Computationally Improved Optimal Deadlock Control Policy for Flexible Manufacturing Systems. *IEEE Trans. Syst. Man Cybern. Part A Syst. Hum.* **2012**, *42*, 404–415. [CrossRef]
36. Pan, Y.L. An Efficient Deadlock Prevention Policy for Flexible Manufacturing Systems Using the Theory of Regions and Selective Siphon Method. *Adv. Sci. Lett.* **2012**, *15*, 342–345. [CrossRef]
37. Huang, Y.S.; Pan, Y.L. An improved maximally permissive deadlock prevention policy based on the theory of regions and reduction approach. *IET Control Theory Appl.* **2011**, *5*, 1069–1078. [CrossRef]
38. Li, Z.W.; Zhou, M.C.; Wu, N.Q. A Survey and Comparison of Petri Net-Based Deadlock Prevention Policies for Flexible Manufacturing Systems. *IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.)* **2008**, *38*, 173–188.
39. Piroddi, L.; Cordone, R.; Fumagalli, I. Selective Siphon Control for Deadlock Prevention in Petri Nets. *IEEE Trans. Syst. Man Cybern. Part A Syst. Hum.* **2008**, *38*, 1337–1348. [CrossRef]
40. Pan, Y.L.; Yang, C.F.; Jeng, M.D. Enhancement of Selective Siphon Control Method for Deadlock Prevention in FMSs. *Math. Probl. Eng.* **2015**, *2015*, 1–6.
41. Moody, J.; Yamalidou, K.; Lemmon, M.; Antsaklis, P. Feedback control of Petri nets based on place invariants. In Proceedings of the 1994 33rd IEEE Conference on Decision and Control, Lake Buena Vista, FL, USA, 14–16 December 1994.
42. Pan, Y.L.; Tseng, C.Y.; Row, T.C. Design of Improved Optimal or Suboptimal Deadlock Prevention for FMSs Based on Place Invariant and Reachability Graph Analysis Methods. *J. Algorithms Comput. Technol.* **2017**, *11*, 261–270. [CrossRef]

43. Pan, Y.L.; Tseng, C.Y.; Huang, Y.S. Design of one computationally improved deadlock prevention based on MFFP technology in flexible manufacturing system. In Proceedings of the 2015 International Automatic Control Conference (CACS), Yilan, Taiwan, 18–20 Novembe 2015.
44. Huang, Y.S.; Pan, Y.L.; Su, P.J. Transition-based Deadlock Detection and Recovery Policy for FMSs Using Graph Technique. *ACM Trans. Embed. Comput. Syst.* **2013**, *12*. [CrossRef]
45. Xiuyan, Z.; Murat, U. Transition-based Deadlock Control Policy Using Reachability Graph for Flexible Manufacturing Systems. *Adv. Mech. Eng.* **2016**, *8*, 1–9.
46. Chen, Y.F.; Li, Z.W.; Al-Ahmari, A.; Wu, N.; Qu, T. Deadlock recovery for flexible manufacturing systems modeled with Petri nets. *Inf. Sci.* **2017**, *381*, 290–303. [CrossRef]
47. Row, C.; Pan, Y.L. Maximally Permissive Deadlock Prevention Policies for Flexible Manufacturing Systems Using Control Transition. *Adv. Mech. Eng.* **2018**, *10*, 1–10. [CrossRef]
48. Row, T.C.; Hsu, W.M.; Pan, Y.L.; Wang, C.C. One Novel and Optimal Deadlock Recovery Policy for Flexible Manufacturing Systems Using Iterative Control Transitions Strategy. *Math. Probl. Eng.* **2019**, *2019*, 1–12. [CrossRef]
49. Pan, Y.L. One Computational Innovation Transition-Based Recovery Policy for Flexible Manufacturing Systems Using Petri nets. *Appl. Sci.* **2020**, *10*, 2332. [CrossRef]
50. Chen, Y.F.; Li, Z.W.; Barkaoui, K.; Wu, N.Q.; Zhou, M.C. Compact supervisory control of discrete event systems by Petri nets with data inhibitor arcs. *IEEE Trans. Syst. Man Cybern. Syst.* **2017**, *47*, 364–379. [CrossRef]
51. Uzam, M.; Li, Z.W.; Gelen, G.; Zakariyya, R.S. A divide-and-conquer-method for the synthesis of liveness enforcing supervisors for flexible manufacturing systems. *J. Intell. Manuf.* **2016**, *27*, 1111–1129. [CrossRef]
52. Murata, T. Petri nets: Properties, analysis and applications. *Proc. IEEE* **1989**, *77*, 541–580. [CrossRef]
53. PN-Tools, A. *Petri Net Analysis Tool, Ver. 1.0*; Pedagogical University Rzesów: Warsaw, Poland, 1987.
54. INA (Integrated Net Analyzer). A Software Tool for Analysis of Petri Nets. Version 2.2, 31.07. 2003. Available online: http://www.informatik.hu-berlin.de/~{}starke/ina.html (accessed on 31 July 2003).
55. Sipser, M. *Introduction to the Theory of Computation*, 2nd ed.; Cengage Learning: Boston, MA, USA, 2013.
56. "Time Complexity". Wikipedia, 28 December 2017. Available online: https://en.wikipedia.org/wiki/Time_complexity (accessed on 11 January 2018).
57. Pan, Y.L.; Huang, Y.S.; Weng, Y.S.; Wu, W.; Jeng, M.D. Computationally Improved Optimal Control Methodology for Linear Programming Problems of Flexible Manufacturing Systems. *J. Appl. Math.* **2013**, *2013*, 1–11. [CrossRef]