Special Issue Article

Advances in Mechanical Engineering

Advances in Mechanical Engineering
2017, Vol. 9(2) 1–12

DOI: 10.1177/1687814017691210
journals.sagepub.com/home/ade
SAGE

Construct the closed-form solution of *A*-net of Petri nets by case study

Daniel Yuh Chao and Tsung Hsien Yu

Abstract

After our researches on the effect of a non-sharing resource in a *k*th order which is the concept of customization manufacturing, in this article we extend the research on the closed-form solution of control-related states to the so-called *A*-net which has one top non-sharing circle subnet connected to the idle place of left process in a *deficient* *k*th order system and is the fundamental model of different productions sharing the same common parts in manufacturing. The formulas just are depended on the parameter *k* and states' function of top non-sharing circle subnet for a subclass of nets with *k* sharing resources. By combining the concept of the partial deadlock avoidance/prevention policy, the moment to launch resource (controller) allocation based on the current state, and the construction of closed-form solution for *deficient* *k*th order system, it can realize the concept of dynamic non-sharing processes' allocation.

Keywords

Control systems, discrete event systems, flexible manufacturing systems, Petri nets

Date received: 8 March 2016; accepted: 5 January 2017

Academic Editor: ZhiWu Li

Introduction

Petri nets (PNs) have been used for modeling and analyzing concurrent systems such as flexible manufacturing system (FMS) or resource allocation system (RAS).[1–10] The net behavior depends not only on the graphical structure but also on the initial marking of the net. Therefore, they cannot be determined by static analysis such as dependency analysis; rather, they can be obtained with reachability analysis.[11–16] It has been shown that the complexity of the reachability analysis of the PNs is exponential.[15]

When using the Integrated Net Analyzer (INA)[17] as reachability analysis tools, there are some potential risks in large model systems: the first is the time-consuming risk as it may take 1 month to complete the reachability analysis; the second is validating the input net structure data by a human; it may take a long time to waiting an error result while the input net structure is wrong; the third is non-significant error due to the fact that presently INA cannot detect the so-called livelock states as shown in section "Computation of CRSs of an ordinary *A*-net"; the information of reachability analysis will become not valuable.

The deadlock avoidance/prevention policy for PNs presently is to find critic first-met bad markings (FBMs) for maximally permissive control purpose which is the policy based on the net structure.[18–23] The structural analysis based deadlock prevention has been extensively studied by researchers, in which siphon computation and control play an essential role.[24–30] However, current advanced approaches such as those in Chen et al.[31] produce maximally permissive supervisors while not being able to synthesize large controllers since reachability analysis of the PN must be employed;

Department of Management Information Systems, National Chengchi University, Taipei, Taiwan

Corresponding author:
Tsung Hsien Yu, Department of Management Information Systems, National Chengchi University, No. 64, Section 2, ZhiNan Road, Wenshan District, Taipei 11605, Taiwan.
Email: yutsunghsien@gmail.com

Chen et al.[20] applied the methodology of interval inhibitor arcs to optimal supervisory control under resources containing multi-token circumstance while suffering from the exponential increment state explosion problem. One solution is to apply mathematics to solve the reachability-related problem of PN in terms of closed-form solutions. As a result, an infinitely large system can be handled with ease.

To efficiently break the ever exponential time plight of getting control states' information into reasonable waiting time, Chao[32] applied the concept of complete reachability graph and graph theory to split the reachability graph of the control net into *reachable, live, forbidden, deadlock, non-reachable*, and *non-reachable + empty-siphon states* (below we call all different types of states as control-related states (CRSs), described in section "Methodology of closed-form solution of kth order system"). Enumerating the tokens' distribution and applying combinatorial mathematics, Chao[32] pioneered the very first closed-form solution of the number of CRSs for kth order system (defined in *Definition* 1). This is the first step that allows the exponential computation time for particular and very large PNs to be reduced into intra-seconds. We have also progressed one step further to analyze the effect of non-sharing resources of kth order and k-net systems based on the token number of idle place in each process being equal to the number of resource places in the associated process[33–37] and proposed the "proof by model" methodology to accelerate the construction of the closed-form formula for PNs.[36,37]

One of contributions of the closed-form solution of PN is that the solution can enhance the capability of dynamically modeling large real-time systems. Examples of application are the concept of moment to launch resource (MLR) (controller) allocation and the deadlock avoidance algorithm introduced below.

On the research of the deadlock prevention policy topic, Chao[18] showed that in a kth order system, it needs additional 10 controllers; Li et al.[28] showed that in a kth order–like system where $k = 3$ and initial marking = 4, it needs additional three controllers where the total token number in these additional controllers is 5. The derived problem is, "Should we need a deadlock prevention policy for very (infinitely) large nets?" Meanwhile in the structural analysis method, there is no indicator to show when to allocate the controllers for a partial deadlock prevention policy.

To solve this problem, based on the contributions of our closed-form solution researches listed above, we proposed a partial deadlock avoidance/prevention policy concept for a very large real-time dynamic RAS: the MLR[35,38] for such policy. Presently, the moment can be calibrated by the future deadlock ratio (the number of deadlock states/the number of reachable states) of the current state, which can be derived real-time by closed-form solution, as the indicator. Letting the deadlock thread-holder (DTH) be regarded as a dummy non-sharing waiting resource that can provide one process holding this DTH and wait for the other processes' work flow, a simple deadlock avoiding algorithm with the transitory maximum number of $\mathrm{int}(k/2) + 1$ DTH is proposed in Yu.[37] The decision-making of a DTH allocation is based on the maximum value of the reachable states that the DTH can be allocated at different location in a kth order system subnet, which will consume heavy computation time by applying mixed integer programming (MIP) method for a large system due to the non-deterministic polynomial-time hard (NP-hard) characteristic of the MIP problem[39] but can be derived by closed-form solution intra-seconds.

However, another symmetry system problem that we propose to solve is the effect of top non-sharing subnet connected to idle place (we call this kind of subnet as top non-sharing circle subnet below (TNCS)) in which structure the position of subnet will affect the token distribution for CRSs caused by some tokens flowing into TNCS but not going through the entire process of kth order system. In other words, with a TNCS connected to a kth order system, we have to construct the closed-form solution of CRSs of a modified kth order system, in which case the number of tokens of idle place is less than the number of places in the belonging process. Here, we call such a modified kth order system as *deficient* kth order system, defined in *Definition* 1, and an A-net as a net composed by TNCS and *deficient* kth order system, defined in *Definition* 4.

In this article, an example of A-net called *a-net* adopted from Liu et al.[40] will be deconstructed and analyzed, followed by the construction of the closed-form solution for the *deficient* kth order system to compute the number of CRSs of the *a-net*.

It is important to use A-net as a target net for research: first, it is most suitable and easy understanding net structure to enter the new research domain, especially the concept of states' classification, the methodology for closed-form solution of kth order system, integration analysis of different net structures, and the generalization of both *deficient* kth order system and TNCS adopted in this article. Second, the information obtained from the INA tool shows that A-net has no deadlock states, as explained in section "Computation of CRSs of an ordinary A-net." Hence, we focus on the characteristic of livelock states where the corresponding kth order system is in deadlock states, which can be analyzed by our methodology. Here, we improve the blind spot of INA tools. Third, by combining the concept of the partial deadlock avoidance/prevention policy, MLR, and the construction of closed-form solution for *deficient* kth order system, it can realize the concept of dynamic non-sharing processes' allocation.

Table 1. Control-related states of third-order system.

Types of states	Lists of states
Reachable	(0 0 0), (0 0 −1), (0 −1 0), (0 −1 −1), (−1 0 −1), (−1 −1 0), (−1 −1 −1), (−1 0 0), (1 0 0),(0 1 0), (1 1 0), (1 0 1), (0 1 1), (1 1 1), (0 0 1)
Forbidden	(1 0 −1), (1 −1 0), (0 1 −1)
Non-reachable	(0 −1 1), (−1 1 0), (−1 0 1), (−1 −1 1), (−1 1 1)
Deadlock	(1 −1 −1), (1 1 −1)
Non-reachable + empty-siphon	(1 −1 1), (−1 1 −1)

The rest of the article is organized as follows. Appendix 2 presents the preliminaries about PNs and S^3PR, respectively. Section "Methodology of closed-form solution of kth order system" defines *deficient* kth order system and lists the relative methodology of closed-form solution of kth order system. Section "Computation of CRSs of an ordinary A-net" analyzes and deconstructs an ordinary A-net structure to compute the number of CRSs. In sections "Computation of CRSs of *deficient* kth order system" and "Computation of CRSs of A-net," we construct the closed-form solution of CRSs for both *deficient* kth order system and A-net. Section "Conclusion" concludes the article. Appendix 3 shows how to apply our methodology to construct the closed-form solution for AR^+-net which is a net structure extended from A-net and contains multi-processes on the right-hand side.

Methodology of closed-form solution of *k*th order system

Let N be a PN and N^r be the reverse net of N. N^r is the net that all the input arcs in N reverse to output arcs; output arcs reverse to input arcs. Chao[32] defined the kth order system (as shown in parts of Definition 1); proposed the concept of complete reachability graph (Figure 5) that lists all states and all paths that any state can be reachable from all states in a kth order system; split the reachability graph of the control net into *reachable, live, forbidden, deadlock, non-reachable*, and *non-reachable + empty-siphon states*; and called all the different types of states as CRSs. Table 1 lists all the CRSs of a third-order system. Based on the concept of complete reachability graph, the relationship of the number of different types of CRSs in a kth order system is that the number of non-reachable states is the number of total states − R (R is the number of reachable states); $L = R - \vartheta$ where L (resp., ϑ) is the number of live (resp., forbidden) states.

According to graph theory, Chao found and proved *Lemma* 1, *Lemma* 2, and *Theorem* 1; *Lemma* 3 (resp., *Theorem* 2) enumerated the number of live (resp., reachable) states of a kth order system; based on the relationship between each type of CRSs, *Corollary* 2 derived the closed-form formulas of the number of forbidden, non-reachable, and non-reachable + empty-siphon states, respectively.

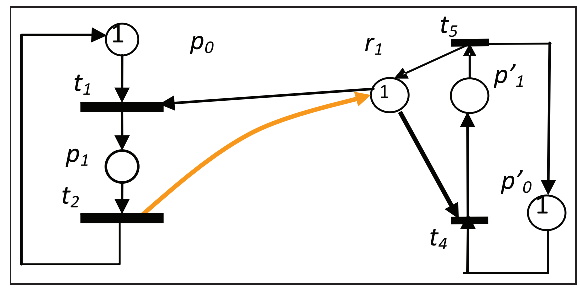

Figure 1. First-order system.

Here, we first define more general form of the kth order system, called *deficient* kth order system below. A kth order system is just one of its special cases.

Definition 1. A *deficient* kth order system is a subclass of S^3PR with k resource places $r_1, r_2, \ldots, r_k$ shared between two processes N_1 and N_2:[32]

1. For all $r \in P_R$, $M_0(r) = 1$.[32]
2. N_1 (resp., N_2) uses $r_1 - r_k$ (resp., $r_k - r_1$) in that order.[32]
3. $1 \le M_0(p_0) < k$, $M_0(p_0') = k$, where p_0 and p_0' are the idle places in processes N_1 and N_2, respectively. When $M_0(p_0) = M_0(p_0') = k$, the system is called a kth order system.
4. Holder places of r_j in N_1 and N_2 are denoted as p_j and p_j', respectively.[32]
5. The compound circuit containing $r_i, r_{i+1}, \ldots, r_{j-1}, r_j$ is called $(r_i - r_j)$ region.[32]
6. There are three possibilities for the token initially at r_i to sit at p_i (N_1), p_i', (N_2), and r_i. The corresponding token or r_i state is denoted by 1, −1, and 0, respectively.[32]

Examples are shown in Figures 1–4. Figure 4 is an example of *deficient* fourth-order system with $M_0(p_0) = 3$.

Definition 2.[32] $s = (x_1, x_2, \ldots, x_k)$, $x_i = 1, 0$, or -1, $i = 1$ to k, is a state for a kth order system N, x_i is the token

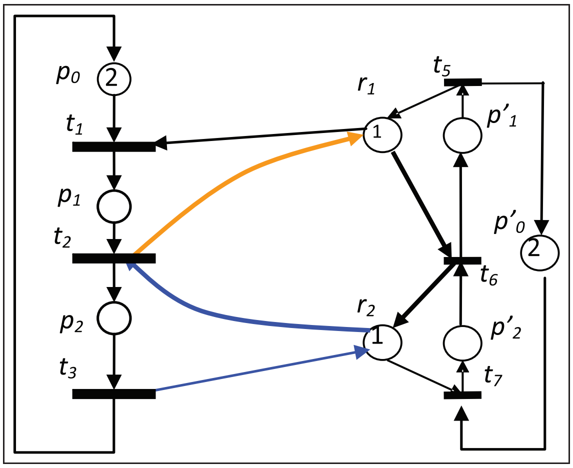

Figure 2. Second-order system.

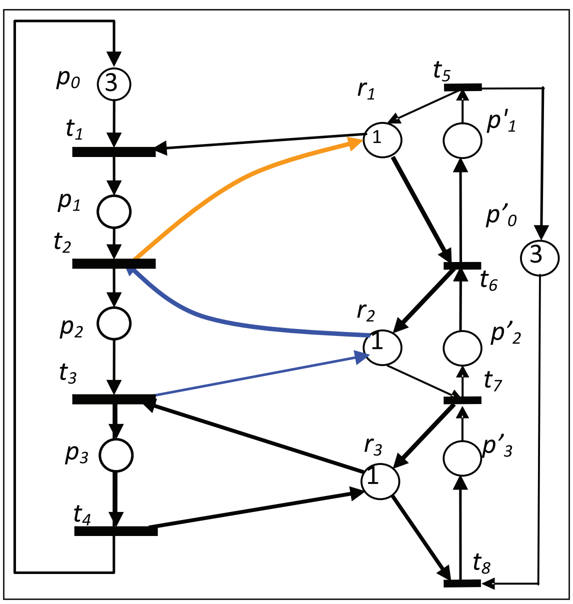

Figure 3. Third-order system.

at r_i to sit at p_i (N_1), r_i, or p_i' (N_2), respectively. (x_i, x_{i+1}, …, x_q, x_{q+1}), $k \geq i \geq 1$, $k \geq q \geq i \geq 1$ (embedded in s) is a sub-state of s.

For example, (0 0 0) is the state of third-order system that only resource places r_1, r_2, and r_3 carry tokens.

In Figures 3 and 4, reversing all the input arcs to output arcs, output arcs to input arcs, and inverting the net structures, we can find the reverse net of a (*deficient*) kth order system is also a (*deficient*) kth order system but the index of sharing resource is reversed. The reverse state of state (a b c) in third-order system N is (c b a) in its reverse net N^r. This implies that a (*deficient*) kth order system and its reverse net have the same number of each type of CRSs. In Table 1, the reverse state of the forbidden state (1 0 −1) in N is (−1 0 1) in N^r

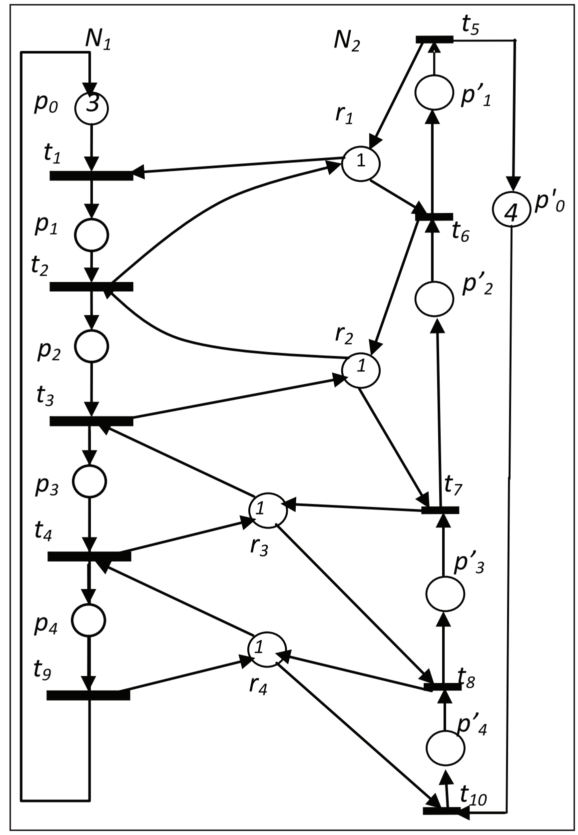

Figure 4. *Deficient* fourth-order system.

which is a non-reachable state as shown in *Lemma* 1. For the third-order system, there are three kinds of unmarked (resp., non-reachable) siphon states: (1 −1 x), (x 1 −1), and (1 0 −1) (resp., (−1 1 x), (x−1 1), and (−1 0 1)), where $x = -1$, 0, 1. Extending Lemmas 1 and 2, Chao et al.[36] found that the reverse state of a live state in a PN N is a live state in its reverse net N^r, and the number of live states in N is equal to the number of live states in N^r which is the main theory of the concept of proof by model.

Lemma 1.[32]. Any forbidden state in N is non-reachable in N^r.

Lemma 2.[32]. Any non-reachable state s in N is a forbidden one or a non-reachable one in N^r.

Theorem 1.[32]. $\vartheta(k) = ¥(k) - B(k)$, where $\vartheta(k)$, $¥(k)$, and $B(k)$ are the number of forbidden, non-reachable, and non-reachable + empty-siphon states in a kth order system, respectively.

Lemma 3.[32]. (1) s is a live state if and only if (iff) $s = \{y_1, \ldots, y_k)|y_i = -1 \text{ or } 0\}$, or $s = \{(x_1, \ldots, x_k)|x_i = 1 \text{ or } 0\}$. (2) The set of live states $L_k = \{(x_1, \ldots, x_k)|x_i = 1 \text{ or } 0\} \cup \{(y_1, \ldots, y_k)|y_i = -1 \text{ or } 0\} = L_a \cup L_b$. (3) The total number of live states is $2^{k+1} - 1$.

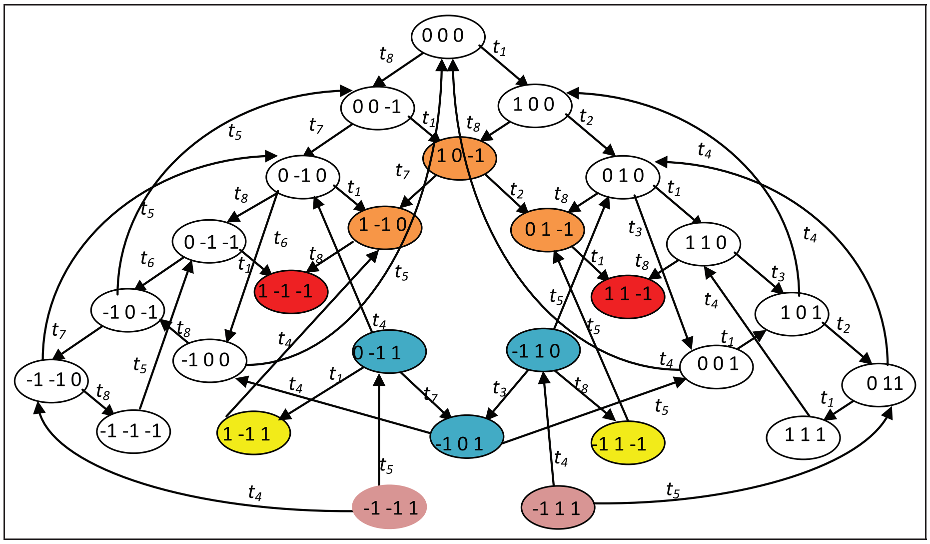

Figure 5. Complete reachability graph of a third-order system (Figure 3).[32]

Theorem 2[32]

1. The possible reachable states are $s = \{(x_1, x_2, \ldots, x_j, y_{j+1}, \ldots, y_k) | 0 \leq j \leq k\} = \{(x_1, \ldots, x_j\ 1\ y_{j+2}, \ldots, y_k) | 1 \leq j \leq \mathrm{k}\} \cup \{(y_1, \ldots, y_k)\}$, where $x_i = 1$ or 0 $(i = 1$ to $j)$ and $y_p = 0$ or -1 $(p = j + 2$ to $k) = R_c \cup R_d$.
2. The total number of reachable states is $(k + 2)2^{(k-1)}$.

Corollary 2.[32]. (1) The number of forbidden states $\vartheta(k) = (k-2)2^{(k-1)} + 1$. (2) The number of non-reachable states $¥(k) = 3^k - (k+2)2^{(k-1)}$. (3) The number of non-reachable + empty-siphon states $B(k) = 3^k - k2^k - 1$.

Theorem 3.[32] In kth order system, a deadlock state has the pattern: $(1_1\ 1_2, \ldots, 1_m -1_{m+1} -1_{m+2}, \ldots, -1_k)$, $1 \leq m < k$. The total number of deadlock states $D(k) = k - 1$.

To sum up, shown below are the total number of each type of CRSs in a kth order system that Chao[32] proved.

The total number of states is 3^k.

The total number of live states $L(k) = 2^{k+1} - 1$.

The total number of reachable states $R(k) = (k+2)2^{(k-1)}$.

The number of forbidden states $\vartheta(k) = R(k) - L(k) = (k-2)2^{(k-1)} + 1$.

The number of non-reachable states $¥(k) = 3^k - R(k) = 3^k - (k+2)2^{(k-1)}$.

The number of non-reachable + empty-siphon states $B(k) = ¥(k) - \vartheta(k) = 3^k - k2^k - 1$.

The total number of deadlock states $D(k) = k - 1$.

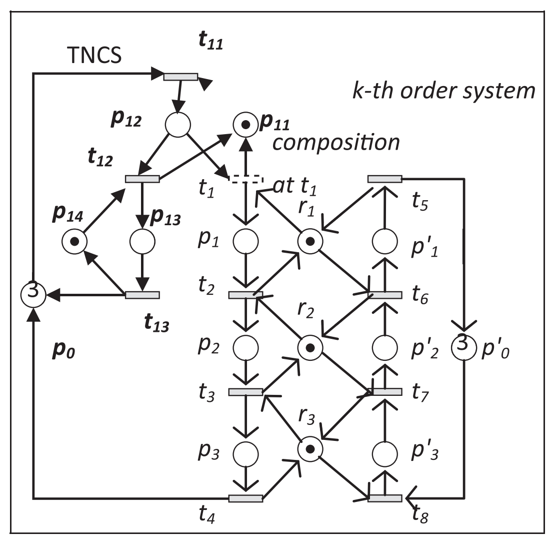

Figure 6. *a*-net, an ordinary *A*-net: composed by deficient *k*th order system and TNCS.[40]

Based on the concept of complete reachability graph (Figure 5) and letting a livelock state be the state that has a directed path to itself state but has no path to initial state, here we extend the definition of forbidden states as the states that have no directed path to the initial state but have a directed path to a deadlock or livelock states. Due to that, the sets of live and forbidden states are two independent sets in a static complete reachability graph, and the number of forbidden states still is $\vartheta = R - L$.

Computation of CRSs of an ordinary *A*-net

Compared with Figure 3, Figure 6 is a kth order–like system connecting with the TNCS on the left-side process formed by (t_1, t_{11}, p_{12}, p_{11}, t_{12}, p_{13}, t_{13}, p_{14}, p_0), where the link point is t_1.

Since there is no empty siphon in TNCS, it will have no deadlock states in this subnet. The tokens in p_{12} can either fire t_1 or t_{12}, which shows that TNCS and the left process can be concurrently processed. However, there are so-called livelock states where tokens can only flow in the left process TNCS when the kth order–like system contains the deadlock states' pattern as shown in Theorem 3. When tokens flow into left process TNCS, it will affect the A-net's state.

We follow the kth order system to compute the number of states. Since p_{11} and p_{14} are resources, and p_{12} is a holder of p_{11}; p_{13} is a holder of p_{14}. We can focus on tokens' distribution of resource set of $\{p_{11}, r_1, r_2, r_3, p_{14}\}$ to simplify presentation and computation of CRSs, where $\{r_1, r_2, r_3\}$ is a resources' set of a third-order system; $\{p_{11}, p_{14}\}$ is a resources' set of TNCS.

Let z be the token number that has flowed into TNCS. We can use $({}^{g(f(z),i)}x_1, x_2, x_3)$ to denote the state of an A-net, where x_1, x_2, x_3 are r_1, r_2, r_3 states defined in step (6) of *Definition* 1; $f(z)$ is a sequence function mapping to two-dimensional vector set which is token distribution of $\{p_{11}, p_{14}\}$. In this case, $f(0) = \{(0_{11}, 0_{14})\}$; $f(1) = \{(1_{11}, 0_{14})), (0_{11}, 1_{14})\}$; $f(2) = \{(1_{11}, 1_{14})\}$ where 0_{11} (resp., 0_{14}) is token in p_{11} (resp., p_{14}); 1_{11} (resp., 1_{14}) is token in p_{12} (resp., p_{13}). $|f(0)| = 1$; $|f(1)| = 2$; $|f(2)| = 1$. $g(f(z), i)$ is a function mapping to a TNCS state according to given $f(z)$ and i, $1 < = i < = |f(z)|$, i is the sequence of the set of $f(z)$. For example $g(f(1), 1) = (1_{11}, 0_{14})$; $g(f(1), 2) = (0_{11}, 1_{14})$; $g(f(2), 1) = (1_{11}, 1_{14})$.

Up to now, we have constructed the presentation of state of an a-net that not only can show the sequence of state transformation but also provide the important variable, $f(z)$, to compute the number of CRSs.

Computation of CRSs of a-net

Let $R(k)$ (resp., $L(k)$ and $D(k)$) be the number of reachable (resp., live and deadlock) states of the kth order system; $R_D(k, q)$ (resp., $L_D(k, q)$ and $D_D(k, q)$) be the number of reachable (resp., live and deadlock) states of *deficient* kth order system with the token number of left idle place being q; $R'(3)$ (resp., $L'(3)$ and $D'(3)$) be the number of reachable (resp., forbidden and livelock) states of *a-net*. We have the below:

The number of reachable states of *a-net* is $R'(3) = |f(0)|R(3) + |f(1)|R_D(3, 2) + |f(2)|R_D(3, 1) = R(3) + 2R_D(3, 2) + R_D(3, 1)$.

The number of live states of *a-net* is $L'(3) = |f(0)|L(3) + |f(1)|L_D(3, 2) + |f(2)|L_D(3, 1) = L(3) + 2L_D(3, 2) + L_D(3, 1)$.

The number of livelock states of *a-net* is $D'(3) = |f(0)|D(3) + |f(1)|D_D(3, 2) + |f(2)|D_D(3, 1) = D(3) + 2D_D(3, 2) + D_D(3, 1)$.

When the token number in TNCS is 0, the number of reachable states in the right third-order system is $|f(0)|R(3)$ since there is only one state $(0_{11}, 0_{14})$; hence, the number of reachable states is $R(3)$ for the third-order system. The case of $|f(1)|R_D(3, 2)$ is when the token number in TNCS is 1; there being two states $(1_{11}, 0_{14})$ and $(0_{11}, 1_{14})$, the number of reachable states is $2R_D(3, 2)$, in which case the token number of the left idle place is 2. The case of $|f(2)|R_D(3, 1)$ is the token number in TNCS being 2, the token number of left idle place being 1, and the state of TNCS being $(1_{11}, 1_{14})$ only. The analysis of $L'(3)$ and $D'(3)$ is similar to that for $R'(3)$.

By Theorem 2, $R(k) = (k + 2)2^{(k-1)} => R(3) = (3 + 2)2^{(3-1)} = 20$.

Extending Theorem 2, $R_D(3, 2) = |R_c| + |R_d| = (2^{(3-1)} + 2 \times 2^{(3-2)} + 2^{(3-1)} - 1) + (2^3) = 19$, where $R_c = \{(1\ y_2\ y_3)\} \cup \{(x_1\ 1\ y_3)\} \cup \{(0_1\ 0_2\ 1_3), (0_1\ 1_2\ 1_3), (1_1\ 0_2\ 1_3)\}$, where $x_1 = 0$ or 1, y_2 and y_3 can be 0 or -1. $R_d = \{(y_1, \ldots, y_3)\}$ $y_p = 0$ or -1, $p = 1$ to 3.

$R_D(3, 1) = |R_c| + |R_d| = (2^{(3-1)} + 2^{(2-1)} + 2^{(1-1)}) + (2^3) = 15$, where $R_c = \{(1\ y_2\ y_3)\} \cup \{(0_1\ 1\ y_3)\} \cup \{(0_1\ 0_2\ 1_3)\}$, where y_2 and y_3 can be 0 or -1. $R_d = \{(y_1, \ldots, y_3)\}$ $y_p = 0$ or -1, $p = 1$ to 3.

Hence, $R'(3) = R(3) + 2R_D(3, 2) + R_D(3, 1) = 20 + 2 \times 19 + 15 = 73$.

By Lemma 3, $L(k) = 2^{k+1} - 1 => L(3) = 2^{3+1} - 1 = 15$.

Extending Lemma 3, $L_D(3, 2) = |L_a| + |L_b| = (2^{(3-3)} + 2^{(3-2)} + 2^{(3-1)} - 1) + (2^3) = 14$, where $L_a = \{(1_1\ 0_2\ 0_3)\} \cup \{(x_1\ 1_2\ 0_3)\} \cup \{(0_1\ 0_2\ 1_3), (0_1\ 1_2\ 1_3), (1_1\ 0_2\ 1_3)\}$ where $x_1 = 1$ or $0\}$, $L_b = \{(y_1\ y_2\ y_3)|y_i = -1$ or 0, $i = 1$ to $3\}$.

$L_D(3, 1) = |L_a| + |L_b| = (2^{(3-3)} + 2^{(3-2)} - 1 + 1) + (2^3) = 11$.

Hence, $L'(3) = L(3) + 2L_D(3, 2) + L_D(3, 1) = 15 + 2 \times 14 + 11 = 54$.

By *Theorem* 3, $D(k) = k - 1$. Since the total number of deadlock states corresponds to the position m, hence $D(3) = D_D(3, 2) = D_D(3, 1) = 2$. $D'(3) = D(3) + 2D_D(3, 2) + D_D(3, 1) = 8$.

The number of forbidden states of *a-net* is $R'(3) - L'(3) = 73 - 54 = 19$.

To sum up, shown below are the total number of reachable, live, forbidden and livelock states of *a-net*.

The total number of reachable states $R'(3) = 73$.

The total number of live states $L'(3) = 54$.

The number of forbidden states $\vartheta'(3) = 19$.

The total number of livelock states $D'(3) = 8$.

Computation of CRSs of *deficient k*th order system

To extend to general A-net, we have to complete the general formula of $R_D(k, q)$ and $L_D(k, q)$ of the *deficient*

Table 2. Counts of impossible states of $R_D(k, q)$ where $k = 8$, $q = 2$.

l	i	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	Total
1	–	1	0: there is no impossible state							0
2	–		1	0: there is no impossible state						0
3	0–0	$C(2, 2) = 1$		1	2^5: $x_j = 0$ or -1, $j = 4$ to 8 ($k - l = 8 - 3 = 5$)					32
4	0–1	$C(3, 3) + C(3, 2) = 4$			1	2^4: $x_j = 0$ or -1, $j = 5$ to 8 ($k - l = 4$)				64
5	0–2	$C(4, 4) + C(4, 3) + C(4, 2) = 11$				1	2^3: ($k - 1 = 8 - 5 = 3$)			88
6	0–3	$C(5, 5) + C(5, 4) + C(5, 3) + C(5, 2) = 26$					1	2^2: ($k - 6 = 2$)		104
7	0–4	$C(6, 6) + C(6, 5) + C(6, 4) + C(6, 3) + C(6, 2) = 57$						1	2^1	114
8	0–5	$C(7, 7) + C(7, 6) + C(7, 5) + C(7, 4) + C(7, 3) + C(7, 2) = 120$							1	120
Total impossible state										522
$R_D(8, 2) = R(8) - 522 = 1280 - 522 = 758$										

kth order system described in section "Computation of CRSs of an ordinary A-net" first.

The phenomenon of the *deficient* kth order system in an A-net is when the token number flowing into TNCS is greater than 1 as described in section "Computation of CRSs of an ordinary A-net."

Construct general formula of $R_D(k, q)$ and $L_D(k, q)$

Let k_q be a *deficient* kth order system with q ($q \leq k$) tokens in the left idle place; that is, $M(p_0) = q$ and $M(p_0') = k$.

Definition 3. An impossible state in a *deficient* kth order system is a reachable state in kth order system, but not a legal state due to the token number in left process being greater than that of the left idle place p_0.

For example, state $s = (1_1\ 1_2\ 1_3\ 1_4\ 0_5)$ is a reachable state in fifth-order system but an impossible state in *deficient* 5_3-th order system, since the token number in left process is 4, greater than 3 which is the token number of the left idle place in *deficient* 5_3-th order system.

Theorem 4. Let $R(k)$ be the reachable states of a kth order system and $R_D(k, q)$ be the number of reachable states of k_q, a *deficient* kth order system, then

$$R_D(k, q) = R(k) - \sum_{l=q+1}^{k} \left(\sum_{i=0}^{l-q-1} C(l-1, l-1-i) \right) (2^{k-l})$$

where $C(l-1,\ l-1-i) = (l-1)!/[(l-1-i)!i!]$, is a binomial coefficient.

Proof. The line of thinking is $R_D(k,\ q) = R(k) -$ (the impossible states in k_q but are all reachable states in kth order system). According to Theorem 2, comparing with $R_D(k, q)$ and $R(k)$, the same value both in $R_D(k, q)$ and $R(k)$ is $|R_d|$, where $R_d = \{(y_1, \ldots, y_k)|y_i = 0$ or -1 ($p = 1$ to k)$\}$, which is the token distribution on the right S^2PR (Definition 7 in Appendix 2). In R_c, the same number of reachable states' subsets both in $R_D(k, q)$ and $R(k)$ are from subset $\{(1_1, y_2, \ldots, y_k)|1 \leq q \leq k$, $y_p = 0$ or -1 ($p = 2$ to k)$\}$ till subset $\{(x_1, \ldots, x_{q-1}\ 1_q y_{q+1}, \ldots, y_k)|1 \leq q \leq k$, where $x_i = 1$ or 0 ($i = 1$ to $q - 1$) and $y_p = 0$ or -1 ($p = q + 1$ to k)$\}$. By induction, the impossibility of reachable states in subset $R_{q+1} = \{(x_1, \ldots, x_q\ 1_{q+1} y_{q+2}, \ldots, y_k)|\ldots\}$ of $R_D(k, q)$ is subset $I_q = \{(x_1, \ldots, x_q\ 1_{q+1} y_{q+2}, \ldots, y_k)|$where $x_i = 1$, $i = 1$ to q, $y_p = 0$ or -1 ($p = q + 2$ to k)$\}$, and $|I_q| = C(q,\ q)2^{(k-q-1)}$. When $R_{q+2} = \{(x_1, \ldots, x_{q+1}\ 1_{q+2} y_{q+3}, \ldots, y_k)|\ldots\}$, the set of impossibility is $I_{q+1} = \{(x_1, \ldots, x_{q+1}\ 1_{q+2} y_{q+3}, \ldots, y_k)\}$ where $x_i = 1$, $i = 1$ to $q + 1$ (the number is $2^{(k-q-2)}C(q + 1, q + 1)) \cup \{(x_1, \ldots, x_{q+1}\ 1_{q+2} y_{q+3}, \ldots, y_k)|x_i = 0$ or 1, $i = 1$ to $q + 1$, $x_1 + \cdots + x_{q+1} = q$, $y_p = 0$ or -1 ($p = q + 3$ to k)$\}$ (the number is $2^{(k-q-2)}C(q + 1, q)$); hence $|I_{q+1}| = [C(q + 1, q + 1)) + C(q + 1, q)]2^{(k-q-2)}$, till $R_k = \{(x_1, \ldots, x_{k-1}\ 1_k)|\ldots\}$ the subset of impossibility is $I_{k-1} = \{(x_1, \ldots, x_{q+1} x_{q+2} x_{q+3} \ldots 1_k)|x_i = 0$ or 1, $i = 1$ to $k - 1$, $x_1 + \cdots + x_{k-1} \geq q\}$, and $|I_{k-1}| = [C(k - 1,\ k - 1)) + \cdots + C(k - 1,\ q)]2^{(k-k)}$. We have all subsets of impossibility occurring from R_{q+1} to R_k, and the number of impossibility counts from $C(q,\ q)2^{(k-q-1)}$ to $[C(k - 1,\ k - 1)) + \cdots + C(k - 1,\ q)]2^{(k-k)} = \sum_{l=q+1}^{k} (\sum_{i=0}^{l-q-1} C(l - 1,\ l - 1 - i))\ (2^{k-l})$.

Hence, $R_D(k, q) = R(k) - \sum_{l=q+1}^{k} (\sum_{i=0}^{l-q-1} C(l - 1, l - 1 - i))(2^{k-l})$. ■

Let $k = 8$, $q = 2$, impossible states of $R_D(8, 2)$ are shown below. l is the first position of $x_i = 1$ (Table 2).

Theorem 5. Let $L(k)$ be the live states of a kth order system, $L_D(k, q)$ be the number of live states of kth order system with token number of left idle place being q

$L_D(k, q) = L(k) - \sum_{l=q+1}^{k} (\sum_{i=0}^{l-q-1} C(l - 1, l - 1 - i))$, where $C(l - 1,\ l - 1 - i) = (l - 1)!/[(l - 1 - i)!i!]$ is a binomial coefficient.

Table 3. The value of $R_D(k, q)$, $L_D(k, q)$, and $L_D(k, q)$, where $k = 8$, $q = 1$ to 7.

q	8	7	6	5	4	3	2	1
$R_D(8, q)$	1280	1279	1270	1233	1140	977	758	511
$L_D(8, q)$	511	510	502	474	418	348	292	264
$\vartheta_D(8, q)$	769	769	768	759	722	629	466	247

Table 4. The value of $R'_A(k, m)$,$L'_A(k, m)$, and $\vartheta'_A(k, m)$, where $k = 4$ to 8; $m = 2$.

k	4	5	6	7	8
$R'_A(k, m)$	184	439	1014	2293	5108
$L'_A(k, m)$	117	244	499	1010	2033
$\vartheta'_A(k, m)$	67	195	515	1283	3075

Proof. The proof is similar to that of $R_D(k, q)$. The difference is that by *Lemma* 3, s is a live state if and only if $s = \{(y_1, \ldots, y_k)|y_i = -1 \text{ or } 0\}$, or $s = \{(x_1, \ldots, x_k)|x_i = 1 \text{ or } 0\}$. Hence, comparing to $R_D(k, q)$, the impossibility of live states in subset $L_{q+1} = \{(x_1, \ldots, x_q\ 1_{q+1} y_{q+2}, \ldots, y_k)|\ldots\}$ of $L_D(k, q)$ is subset $I_q = \{(x_1, \ldots, x_q\ 1_{q+1}\ 0_{q+2} \ldots 0_k)|$ where $x_i = 1$, $i = 1$ to q, $y_p = 0$ ($p = q + 2$ to k)$\}$, $|I_q| = C(q, q)$. $I_{q+1} = \{(x_1, \ldots, x_{q+1}\ 1_{q+2}\ 0_{q+3} \ldots 0_k)\}$ where $x_i = 1$, $i = 1$ to $q + 1$ (the number is $C(q + 1, q + 1)) \cup \{(x_1, \ldots, x_{q+1}\ 1_{q+2}\ 0_{q+3} \ldots 0_k)|x_i = 0$ or 1, $i = 1$ to $q + 1$, $x_1 + \cdots + x_{q+1} = q\}$ (the number is $C(q + 1, q)$), hence $|I_{q+1}| = C(q + 1, q + 1)) + C(q + 1, q)$. Similar to $R_D(k, q)$, we have all the subsets of impossibility occurring from L_{q+1} to L_k; the number of impossibility counts from $C(q, q)$ to $[C(k-1, k-1)) + \cdots + C(k-1, q) = \sum_{l=q+1}^{k} (\sum_{i=0}^{l-q-1} C(l-1, l-1-i))$.

Hence, $L_D(k, q) = L(k) - \sum_{l=q+1}^{k} (\sum_{i=0}^{l-q-1} C(l-1, l-1-i))$. ■

The forbidden states of k_q is $\vartheta_D(k, q) = R_D(k, q) - L_D(k, q)$. Table 3 shows the number of reachable, live and forbidden states of k_q where $k = 8$ and $q = 1$ to 7

Computation of CRSs of *A*-net

According to $R_D(k, q)$ and $L_D(k, q)$ of *deficient* kth order system derived in section "Computation of CRSs of *deficient* kth order system," and a given state function of TNCS, $f(z)$, in this section we construct closed-form solution to compute CRSs of A-net.

Definition 4. An A-net system is a subclass of S^3PR composed by a kth order system and a deadlock-free TNCS connected to idle process of left process:

Table 5. The value of $R'_A(k, m)$, where $m = 3$ to 5; $k = 8$.

m	3	4	5
$R'_A(k, m)$	10,160	20,088	39,382

1. The link point of kth order system and TNCS is the top-left transition of kth order system.
2. None of transition of TNCS is firable by the resources and places of kth order system.
3. Let m be the maximum token number that can flow into TNCS; the limitation of m is $m < k$.
4. Under the limitation of $m < k$, both kth order system and TNCS can expand its net structure independently.
5. There is a measurable state function of TNCS $f(z)$, where z is the number of tokens flowing into TNCS, such that $|f(z)|$ can map to a non-ambiguous number of different states of combination under current information of z in TNCS.

Let $R_D(k, k) = R(k)$. Hence, we have the following:

The reachable state of A-net is $R'_A(k, m) = \sum_{l=0}^{m} (|f(l)|) R_D(k, k-l))$.

The live state of A-net is $L'_A(k, m) = \sum_{l=0}^{m} (|f(l)|) L_D(k, k-l))$.

The forbidden state of A-net is $\vartheta'_A(k, m) = R'_A(k, m) - L'_A(k, m) = \sum_{l=0}^{m} (|f(l)|)(R_D(k, k-l) - L_D(k, k-l))$.

The livelock state of A-net is $D'_A(k, m) = \sum_{l=0}^{m} (|f(l)|) D_D(k, k-l)) = \sum_{l=0}^{m} (|f(l)|) D(k)$.

The important characteristic of A-net is that m and k can be expanded independently under the condition $m < k$. Extending k of A-net from 4 to 8, we have the value of $R'_A(k, m)$,$L'_A(k, m)$, and $\vartheta'_A(k, m)$ as listed in Table 4.

These results of $R'_A(k, m)$ have been validated experimentally by INA.

Extending m of A-net from 3 to 5 with $k = 8$, we have the value of $R'_A(k, m)$ as listed in Table 5.

Application. Based on the closed-form solution and the concept of proof by model, we can accelerate the construction of the closed-form formula of the reverse net of A-net, rev(A-net), which is the fundamental model of merging two different manufacturing processes, due to that the live states' pattern of rev(A-net) will be the reverse states of A-net's live states and they have the same number of live states.

In Appendix 3, we will show how to extend our methodology to derivate the closed-form solution of AR^+-net which is a net structure containing a non-sharing subnet and multi-processes on the right-hand side extended from A-net.

Conclusion

Here, we not only report the very first method to compute in closed form the number of CRSs of A-net without constructing a reachability graph but also demonstrate the procedure of how to construct *deficient* kth order system by particular case step by step. The most important line of thinking is $R_D(k, q) = R(k) -$ (the impossible states in k_q but are all reachable states in kth order system), as shown in the proof procedure of *Theorem* 4. According to the procedure listed above, readers can derive the closed-form solution of more complicated A-net or even of which case non-sharing circle subnet can be any position in left or right process. Besides, by our closed-form solution, we can provide live, forbidden, and livelock states' information that INA tool cannot offer, which show non-significant reachable states' information only, due to no deadlock states in A-net and that INA tool cannot detect livelock states. This helps estimate the percentage of livelock (deadlock) and legal state losses due to the addition of a monitor and avoid the dire situation of mid-run abortion of reachability analysis due to exhausted memory. Furthermore, here we extend the MLR concept to the variant kth order system with non-sharing subnet.

Future work should extend the analysis method in this article to A-net with one non-sharing resource (e.g. used by only one process) located in any position of the system.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the National Science Council (Republic of China) under Grant NSC 103-2221-E-004-001.

References

1. Chao DY. Reachability of nonsynchronized choice Petri nets and its applications. *IEEE T Syst Man Cyb A* 2005; 35: 1203–1213.
2. Chao DY. Computation of elementary siphons in Petri nets for deadlock control. *Comput J* 2006; 49: 470–479.
3. Chao DY. Improvement of suboptimal siphon- and FBM-based control model of a well-known S^3PR. *IEEE T Autom Sci Eng* 2011; 8: 404–411.
4. Chao DY. Enumeration of lost states of a suboptimal control model of a well-known S^3PR. *IET Control Theory Appl* 2011b; 5: 1277–1286.
5. Ezpeleta JJ, Colom M and Martinez J. A Petri net based deadlock prevention policy for flexible manufacturing systems. *IEEE T Robot Autom* 1995; 11: 173–184.
6. Lee JS, Zhou MC and Hsu PL. An application of Petri nets to supervisory control for human-computer interactive systems. *IEEE T Ind Electron* 2005; 52: 1220–1226.
7. Sun P, Jiang CJ and Zhou MC. Interactive web service composition based on Petri net. *T Inst Measure Control* 2009; 33: 116–132.
8. Uzam M and Zhou MC. An improved iterative synthesis approach for liveness enforcing supervisors of flexible manufacturing systems. *Int. J Prod Res* 2006; 44: 1987–2030.
9. Wang JC, Zhou XZ and Ding JH. Software architectural modelling and verification: a Petri net and temporal logic approach. *T Inst Measure Control* 2010; 33: 168–181.
10. Zimmermann A. Technische Universitat Ilmenau, System and Software Engineering, D-98684 Ilmenau, Germany, 2012, http://www.tu-ilmenau.de/TimeNeT
11. Ferrarini L. On the reachability and reversibility problems in a class of Petri nets. *IEEE T Syst Man Cyb* 1994; 24: 1474–1482.
12. Hiraishi K and Ichikawa A. A class of Petri nets that a necessary and sufficient condition for reachability is obtainable. *T Soc Instrum Control Eng* 1988; 24: 635–640.
13. Ichikawa A, Yokoyama K and Kurogi S. Control of event-driven systems: reachability and control of conflict-free Petri nets. *T Soc Instrum Control Eng* 1985; 21: 324–330.
14. Kostin AE. Reachability analysis in T-invariant-less Petri nets. *IEEE T Autom Control* 2003; 48: 1019–1024.
15. Lee DI, Kumagai S and Kodama S. Reachability of LSFC nets. *IEEE Int Symp Circuit Syst* 1990; 4: 2666–2669.
16. Liang H and Chao DY. Enumeration of reachable states for arbitrary marked graphs. *IET Control Theory Appl* 2012; 6: 1536–1543.
17. Starke PH. *INA: Integrated Net Analyzer handbook*. Germany: Humboldt University, 1992.
18. Chao DY. Closed-form solution of controller synthesis for infinitely large systems of resource sharing systems of

a subclass of Petri nets. *T Inst Measure Control* 2015; 38: 83–92.

19. Li ZW and Zhao M. On controllability of dependent siphons for deadlock prevention in generalized Petri nets. *IEEE T Syst Man Cyb A* 2008; 38: 369–384.
20. Chen YF, Li ZW and Barkaoui K. New Petri net structure and its application to optimal supervisory control: interval inhibitor arcs. *IEEE T Syst Man Cyb Syst* 2014; 44: 1384–1400.
21. Uzam M, Li ZW, Gelen G, et al. A divide-and-conquer-method for the synthesis of liveness enforcing supervisors for flexible manufacturing systems. *J Intell Manuf* 2016; 27: 1111–1129.
22. Liu GY and Barkaoui K. Necessary and sufficient liveness condition of GS3PR Petri nets. *Int J Syst Sci* 2015; 46: 1147–1160.
23. Li SY, An AM, Wu HM, et al. Policy to cope with deadlocks and livelocks for flexible manufacturing systems using the max-controlled new smart siphon. *IET Control Theory Appl* 2014; 8: 1607.
24. Chao DY. A simple modification of deadlock prevention policy of S3PR based on elementary siphons. *T Inst Measure Control* 2010; 33: 93–115.
25. Chao DY, Chen J-T and Yu F. A novel liveness condition for S3PGR2. *T Inst Measure Control* 2012; 35: 131–137.
26. Chao DY and Wu K-C. An integrated approach for supervisory control of a subclass of Petri nets. *T Inst Measure Control* 2012; 35:117–130.
27. Chen YF, Li ZW, Khalgui M, et al. Design of a maximally permissive liveness: enforcing Petri net supervisor for flexible manufacturing systems. *IEEE T Autom Sci Eng* 2011; 8: 374–393.
28. Li ZW, Liu GY, Hanisch HM, et al. Deadlock prevention based on structure reuse of Petri net supervisors for flexible manufacturing systems. *IEEE T Syst Man Cyb A* 2012; 42:178–191.
29. Li ZW, Wu NQ and Zhou MC. Deadlock control of automated manufacturing systems based on Petri nets—a literature review. *IEEE T Syst Man Cybt C* 2012; 42: 437–462.
30. Li S and Li ZW. Solving siphons with the minimal cardinality in Petri nets and its applications to deadlock control. *Int J Prod Res* 2012; 50: 6203–6218.
31. Chen YF, Li ZW and Zhou MC. Optimal supervisory control of flexible manufacturing systems by Petri Nets: a set classification approach. *IEEE T Autom Sci Eng* 2014; 11: 549–563.
32. Chao DY. Enumeration of reachable (forbidden, live, and deadlock) states of K-th order system of Petri nets. *IMA J Math Control Inf* 2014; 32: 823–837.
33. Chao DY and Yu TH. Computation of control related states of bottom *k*-th order system (with a non-sharing resource place) of Petri nets. *T Inst Measure Control* 2014; 37: 382–395.
34. Chao DY and Yu TH. Enumeration of reachable, forbidden, live states of gen-left k-net system (with a non-sharing resource place) of Petri nets. In: *Proceedings of the 2014 IEEE symposium on computational intelligence in control and automation (CICA)*, Orlando, FL, 9–12 December 2014, pp.1–7. New York: IEEE.
35. Chao DY and Yu TH. The fundamental closed-form solution of control related states of k-th order S3PR system with left-side non-sharing resource places of Petri nets. *Int J Control* 2015; 89: 169–178.
36. Chao DY, Chi YP, Yu TH, et al. Proof by model: a new knowledge-based reachability analysis methodology for Petri net. *IMA J Math Control Inf*. Epub ahead of print 11 July 2016. DOI: 10.1093/imamci/dnw025.
37. Yu TH. Parameterized of control related states of Gen-Right k-th order system of Petri nets based on proof by model of Gen-Left. In: *Proceedings of the 42nd annual conference of the IEEE industrial electronics society, IECON*, Florence, Italy, 23–26 October 2016, pp.276–281. New York: IEEE.
38. Chao DY and Yu TH. MLR: a new concept to launch a partial deadlock avoidance policy for k-th order system of Petri nets. In: *Proceedings of the 41st annual conference of the IEEE industrial electronics society, IECON*, Yokohama, Japan, 9–12 November 2015, pp.003148–003152. New York: IEEE.
39. Schrijver A. *Theory of linear and integer programming*. Hoboken, NJ: John Wiley & Sons, 1998.
40. Liu M, Wang S and Li ZW. Supervisor reconfiguration for deadlock prevention by resources reallocation. *J Appl Math* 2013; 2013: 315894.

Appendix 1

Notation

a-net	ordinary A-net
A-net	net composed by *deficient* *k*th order system and TNCS
$B(k)$	total number of non-reachable + empty-siphon states in *k*th order system
$D(k)$	total number of deadlock states in *k*th order system
$D'_A(k,m)$	total number of livelock states in *A*-net
$D_D(k)$	total number of deadlock states in *deficient* *k*th order system
INA	tool package supporting the analysis of place/transition nets (Petri nets) and colored Petri nets (http://www2.informatik.hu-berlin.de/˜starke/ina.html)
$L(k)$	number of live states in *k*th order system
$L'_A(k,m)$	number of live states in *A*-net
$L_D(k)$	number of live states in *deficient* *k*th order system
M	marking
M_0	initial marking
$M_0(p)$	all tokens initially in p
MG	marked graph
N	Petri net (P, T, F, W)
N_i	S^2PR in N, $i = 1, 2, \ldots, n$
N^r	reverse net of N
(N, M_0)	marked net or a net system
P	set of places

p_i	operation place, $i = 1, 2, \ldots, n$
P_i	set of operation places of N_i, $i = 1, 2, \ldots, n$
r	resource place
$R(k)$	number of reachable states in kth order system
$R'_A(k,m)$	number of reachable states in A-net
$R_D(k)$	number of reachable states in *deficient* kth order system
$R(N, M_0)$	set of reachable marking in the net (N, M_0)
S	siphon or strict minimal siphon
SMS	strict minimal siphon, that is, a minimal siphon that does not contain a siphon as a proper subset
S^2P	simple sequential process
S^2PR	simple sequential process with resources
S^3PR	systems of simple sequential process with resources
¥(k)	total number of non-reachable states in kth order system
$\vartheta(k)$	number of forbidden states in kth order system
$\vartheta'_A(k,m)$	number of forbidden states in A-net
$\vartheta_D(k)$	number of forbidden states in *deficient* kth order system

Appendix 2

Preliminaries

A Petri net (PN) is a four-tuple $N = (P, T, F, W)$, where P is the set of *places;* T is the set of *transitions;* $F \subseteq (P \times T) \cup (T \times P)$ is called *flow relation* of the net, represented by arcs with arrows from places to transitions or vice versa; and $W\colon F \to \mathrm{Z}$ (the set of non-negative integers) is a mapping that assigns a weight to an arc. M_0: $P \to Z$ is the *initial marking* assigned to each place $p \in P$, $M_0(p)$ tokens. (N, M_0) is called a marked net or a net system. In the special case that W maps onto $\{0, 1\}$, the PN is said to be *ordinary* (otherwise, *general*). $N' = (P', T', F', W')$ is called a subnet of N where $P' \subseteq P$, $T' \subseteq T$, $F' = F \cap ((P' \times T') \cup (T' \times P')$ and $W'\colon F' \to \mathrm{Z}'$.

The set of input (resp., output) transitions of a place p is denoted by ${}^\bullet p$ (resp., $p^\bullet$). Similarly, the set of input (resp., output) places of a transition t is denoted by ${}^\bullet t$ (resp., $t^\bullet$). Finally, an ordinary PN such that (s.t.)$\forall t \in T$, $|t^\bullet| = |{}^\bullet t| = 1$ is called a *state machine*. It is called a *Marked Graph* if $\forall p \in {}^\bullet P$, $|p^\bullet| = |{}^\bullet p| = 1$. A PN is strongly connected (SC) if $\forall x, x' \in (P \cup T)$, such that $x \neq x'$, there is a directed path from x to x'. A node x in $N = (P, T, F, W)$ is either a $p \in P$ or a $t \in T$. N^r is the reverse net of N obtained by reversing the direction of all arcs in N with the initial marking unchanged.

$R(N, M_0)$ is the set of markings reachable from M_0. A forbidden (resp., live) marking or state is one that (resp., not) is a—or necessarily evolves into—deadlock marking.

A transition $t \in T$ is live at M_0 if $\forall M \in R(N, M_0)$, $\exists M' \in R(N, M)$, and t is enabled at M'. A PN is *live* at M_0 if $\forall t \in T$, and t is live at M_0. A PN is said to be *deadlock-free*, if at least one transition is enabled at every reachable marking.

For a Petri net (N, M_0), a non-empty subset S (resp., τ) of places is called a *siphon* (resp., *trap*) if ${}^\bullet S \subseteq S^\bullet$ (resp., $\tau^\bullet \subseteq {}^\bullet\tau$), that is, every transition having an output (resp., input) place in S has an input (resp., output) place in S (resp., τ). A siphon is a set of places where tokens can continuously flow out so that $M_0(S) = \sum_{p \in S} M_0(p) = 0$; S is called an *empty siphon* or *unmarked siphon* at M_0; all output transitions of S are permanently dead. A *minimal siphon* does not contain a siphon as a proper subset. It is called a *strict minimal siphon* (*SMS*), denoted by S, if it does not contain a trap.

Definition 5.[5] A simple sequential process (S^2P) is a net $N = (P \cup \{p^0\}, T, F)$ where (1) $P \neq \varnothing$, $p^0 \notin P$ (p^0 is called the process idle or initial or final operation place), (2) N is strongly connected state machine, and (3) every circuit of N contains the place p^0.

Transitions in $p^{0\bullet}$ and ${}^\bullet p^0$ are called *source and sink transitions*, respectively.

Definition 6.[5] A simple sequential process with resources (S^2PR), also called a working process (WP), is a net $N = (P \cup \{p^0\} \cup P_R, T, F)$ so that (1) the subnet generated by $X = P \cup \{p^0\} \cup T$ is an S^2P; (2) $P_R \neq \varnothing$ and $P \cup \{p^0\} \cap \mathrm{P}_R = \varnothing$; (3)$\forall p \forall P$, $\forall t \in {}^\bullet p$, $\forall t' \in p^\bullet$, $\exists r_p \in P_R$, ${}^\bullet t \cap P_R = t'^\bullet \cap P_R = \{r_p\}$; (4) the two following statements are verified: $\forall r \in P_R$, (a) ${}^{\bullet\bullet}r \cap P = r^{\bullet\bullet} \cap P \neq \varnothing$; b) ${}^\bullet r \cap r^\bullet = \varnothing$; (5) ${}^{\bullet\bullet}(p^0) \cap P_R = (p^0)^{\bullet\bullet} \cap P_R = \varnothing$. $\forall p \in P$, p is called an operation place. $\forall r \in P_R$, r is called a resource place. $H(r) = {}^{\bullet\bullet}r \cap P$ denotes the set of holders of r (operation places that use r). Any resource r is associated with a minimal P-invariant whose support is denoted by $\rho(r) = \{r\} \cup H(r)$.

Definition 7.[5] A system of S^2PR (S^3PR) is defined recursively as follows: (1) an S^2PR is defined as an S^3PR and (2) let $N_i = (P_i \cup P_i^{\,0} \cup P_{Ri}, T_i, F_i)$, $i \in \{1, 2\}$ be two S^3PR so that $(P_1 \cup P_1^{\,0}) \cap (P_2 \cup P_2^{\,0}) = \varnothing$. $P_{R1} \cap P_{R2} = P_C(\neq \varnothing)$ and $T_1 \cap T_2 = \varnothing$. The net $N = (P \cup P^0 \cup P_R, T, F)$ resulting from the composition of N_1 and N_2 via P_C (denoted by N_1 o N_2) defined as follows: (1)

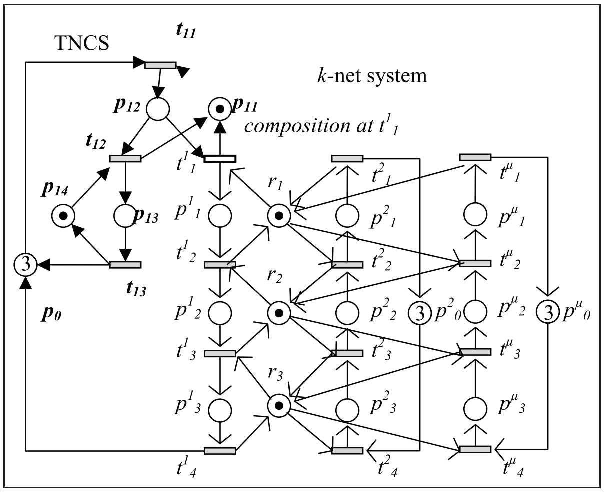

Figure 7. AR^+-net: composed by *deficient* k-net system and TNCS.

$P = P_1 \cup P_2$; (2) $P^0 = P_1{}^0 \cup P_2{}^0$; (3) $P_R = P_{R1} \cup P_{R2}$; (4) $T = T_1 \cup T_2$, and (5) $F = F_1 \cup F_2$ is also an S^3PR.

Appendix 3

Applying to AR$^+$-net

An AR^+-net is a k-net[32]-like system connecting with the TNCS in left side process as shown in Figure 7.

Let z_i^j denote the ith token state at Process j (>1). $z_i^j = -1$ means the ith token is at operation place p_i of Process j and not at operation place p_i of other processes. Hence, $z_i^2 + z_i^3 + \cdots + z_i^\mu = z_i = -1$ with ($\mu - 1$) possibilities; that is, exactly one of $z_i^2, z_i^3, \ldots, z_i^\mu$ equals -1; the rest are 0. $z_i^j = 0$ means that the ith token is at resource place r_i. Thus, $z_i \leq 0$.

Chao[32] constructed the formulas of live ($L(k, \mu)$) and reachable ($R(k, \mu)$) states for the k-net as shown in *Theorems* 6 and 7:

Theorem 6.[32] For a k-net with μ processes, the total number of live states is $L(k, \mu) = 2^k + (\mu)^k - 1$.

Theorem 7.[32] For a k-net with μ processes, the total number of reachable states is $R(k, \mu) = 2^k + (\mu - 1)y(1 - x^k)/(1 - x)$, where $x = \mu/2$ and $y = 2^{(k-1)}$.

Here, we extend to construct the formulas of live ($L_{AR}(k, \mu, m)$) and reachable ($R_{AR}(k, \mu, m)$) states for the AR^+-net based on these results.

Theorem 8. For an AR^+-net with μ processes, the total number of reachable markings is $R_{AR}(k, \mu, m) = \sum_{l=0}^{m} (|f(l)|) \quad R_D(k, k-l, \mu))$, where $R_D(k, q, \mu) = R(k, \mu) - \sum_{l=q+1}^{k} (\sum_{i=0}^{l-q-1} C(l-1, l-1-i))(\mu^{k-l})$ and $f(l)$ is a measurable state function of TNCS, where l is the number of tokens flowing into TNCS.

Proof. In the proof procedure of *Theorem* 4 replacing the token distribution number of the sharing resources of the impossible states 2 ($y_p = 0$ or -1) by μ ($z_p = 0, z_p^2, z_p^3, \ldots,$ or z_p^μ), we have the formula. ■

Theorem 9. For an AR^+-net with μ processes, the total number of live markings is $L_{AR}(k,\mu,m) = \sum_{l=0}^{m} (|f(l)|) L_D(k, k-l, \mu))$, where $L_D(k,q,\mu) = L(k,\mu) - \sum_{l=q+1}^{k} (\sum_{i=0}^{l-q-1} C(l-1, l-1-i))$ and $f(l)$ is a measurable state function of TNCS, where l is the number of tokens flowing into TNCS.

Proof. In the proof procedure of *Theorem* 5, the token distribution numbers of the sharing resources of impossible state are the same both in kth order system and k-net system, and we have the formula. ■