Supplementary File for "A Petri Net-based Deadlock Avoidance Policy for Flexible Manufacturing Systems with Assembly Operations and Multiple Resource Acquisition"

To test the efficiency in terms of permissiveness of MBA, a large example is given, as shown in Fig. 5. The system contains six types of robots, i.e., r_1-r_6 and eight types of machines, i.e., m_1-m_8. The capacities of m_1, m_7, m_8, r_1, and r_6 are all 1, those of m_2, m_3, m_4, m_5, m_6, r_2, r_3, and r_4 are all 2, and that of r_5 is 4. There are five types of raw parts, which can be manufactured and assembled into three types of products. The part type set is $J = \{J_i \mid i \in Z_5\}$. A raw J_1 part is taken from I_1 by r_1. After being processed by m_1 and m_3 simultaneously, it is moved to m_2 by r_2. After being processed by m_2, a product is completed and is moved to O_1 by r_1. A raw J_2 part is taken from I_2 by r_1. After being processed by m_2 and m_4 simultaneously, it is moved to m_1 by r_2. A raw J_3 part is taken from I_3 by r_5. After being processed by m_4 and m_5 simultaneously, it is moved to m_6 by r_4. After a J_2 part being processed by m_1 and a J_3 part being processed by m_6, they are moved to m_3 by r_4 and r_3, respectively. After being assembled in m_3, a product is completed and is moved to O_2 by r_3. A raw J_4 part is taken from I_4 by r_6. After being processed by m_7, it is moved either to m_6 by r_3 or to m_8 by r_6. A raw J_5 part is taken from I_5 by r_6. Then, it is processed by m_8. After a J_4 part being processed by m_6 or m_8 and two J_5 parts being processed by m_8, they are moved to m_5 and m_7 by r_2 and r_5, respectively. After being assembled by m_5 and m_7 simultaneously, a product is completed and is moved to O_3 by r_6. In Fig. 5, the labeled notation, namely, t_j ($j \in Z_{33}$) beside each arc and leading to a resource indicates that the resource is required in order to trigger the event represented by t_j. The number of required resources is represented by the weight of the outgoing arc. The occurrence of an event stands for moving one or more parts from the current resource to a required one. The number of moved parts is represented by the integer before t_j. For example, $2t_{25}$ in Fig. 5 represents that the occurrence of t_{25} moves 2 parts from r_5. For simplicity, the integer 1 is omitted.

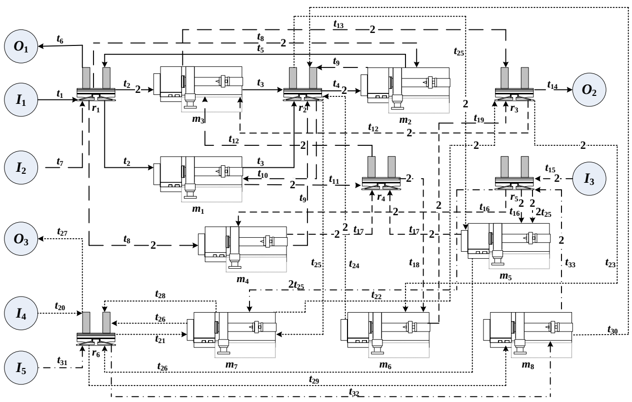

Fig. 5. The block diagram of an AMS.

The system's PN model is denoted as (N_3, M_{30}) as shown in Fig. 6. As it allows flexible routes, H_{15} cannot be applied to it. Since $\exists W(p_{37}, t_{25}) > 1$, (N_3, M_{30}) is not an AEMG. Thus, H_{13} cannot be applied to it. The first two conditions in W_{08} can be changed to twelve constraints as follows:

1) $M(p_2) + M(p_3) + M(p_5) + M(p_6) + M(p_9) + M(p_{10}) + M(p_{12}) + M(p_{14}) \leq 1$;

2) $M(p_2) + M(p_3) + M(p_4) + M(p_5) + M(p_6) + M(p_9) + M(p_{10}) + M(p_{11}) + M(p_{12}) + M(p_{13}) + M(p_{14}) + M(p_{15}) + M(p_{19}) + M(p_{20}) + M(p_{22}) + M(p_{26}) + M(p_{28}) \leq 2$;

3) $M(p_2) + M(p_3) + M(p_5) + M(p_6) + M(p_9) + M(p_{10}) + M(p_{12}) + M(p_{13}) + M(p_{14}) + M(p_{15}) + M(p_{19}) +$

$M(p_{20}) + M(p_{22}) + M(p_{26}) \leq 1$;

4) $M(p_{38}) + M(p_{39}) + M(p_{40}) + M(p_{46}) + M(p_{47}) \geq 7$;

5) $M(p_3) + M(p_{10}) + M(p_{13}) + M(p_{14}) + M(p_{15}) + M(p_{19}) + M(p_{20}) + M(p_{21}) + M(p_{22}) + M(p_{26}) + M(p_{27}) + M(p_{29}) \leq 1$;

6) $M(p_3) + M(p_{10}) + M(p_{13}) + M(p_{14}) + M(p_{15}) + M(p_{18}) + M(p_{19}) + M(p_{20}) + M(p_{21}) + M(p_{22}) + M(p_{26}) + M(p_{27}) + M(p_{29}) + M(p_{37}) \leq 2$;

7) $M(p_{42}) + M(p_{45}) + M(p_{51}) \geq 3$;

8) $M(p_{18}) + M(p_{19}) + M(p_{24}) + M(p_{25}) + M(p_{29}) + M(p_{30}) + M(p_{32}) + M(p_{33}) + M(p_{35}) + M(p_{36}) + M(p_{37}) \leq 2$;

9) $M(p_{15}) + M(p_{19}) + M(p_{21}) + M(p_{22}) + M(p_{24}) + M(p_{25}) + M(p_{26}) + M(p_{27}) + M(p_{29}) + M(p_{30}) + M(p_{32}) + M(p_{35}) \leq 1$;

10) $M(p_4) + M(p_{11}) + M(p_{15}) + M(p_{19}) + M(p_{21}) + M(p_{22}) + M(p_{24}) + M(p_{25}) + M(p_{26}) + M(p_{27}) + M(p_{28}) + M(p_{29}) + M(p_{30}) + M(p_{32}) + M(p_{35}) \leq 2$;

11) $M(p_{19}) + M(p_{24}) + M(p_{25}) + M(p_{29}) + M(p_{30}) + M(p_{32}) + M(p_{33}) + M(p_{35}) + M(p_{36}) \leq 1$;

12) $M(p_4) + M(p_{11}) + M(p_{19}) + M(p_{24}) + M(p_{25}) + M(p_{28}) + M(p_{29}) + M(p_{30}) + M(p_{32}) + M(p_{33}) + M(p_{35}) + M(p_{36}) \leq 2$.

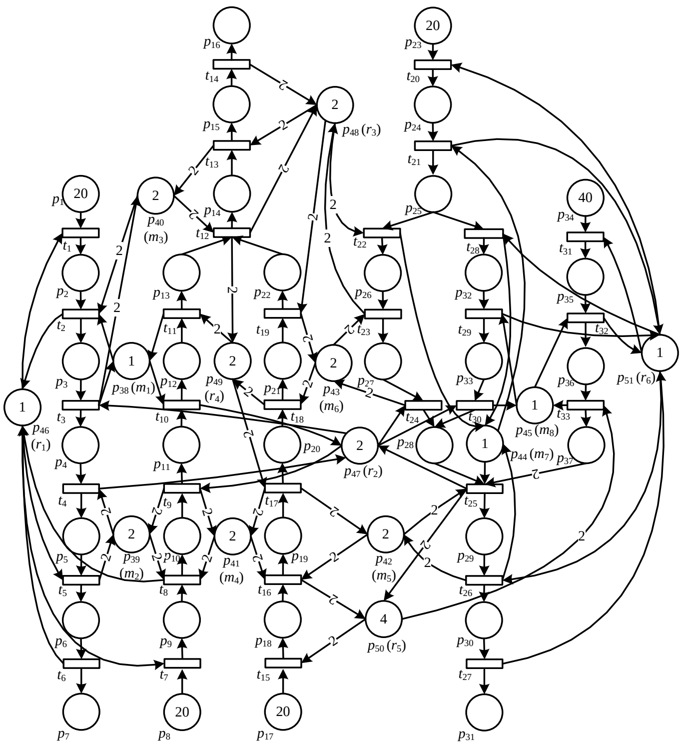

Fig. 6. Petri net model of the AMS in Fig. 5.

For (N_3, M_{30}), the number of reachable markings under these twelve constraints is 114, which means that the number of reachable markings under W_{08} is no more than 114. The number of reachable markings under MBA is 2999. Obviously, MBA admits much more markings than W_{08}. Since W_{08} admits much more markings than [24], MBA owns the highest permissiveness among these policies.

Now, we use AMS in Figs. 3, 4, and 6 to test the runtime of MBA. Note that H_{13} and H_{15} are both offline control policies that are established in advance. Comparing MBA with them in terms of runtime is meaningless. Thus, we only compare the runtime of MBA with that of W_{08}. All algorithms are implemented in C++. They are compiled by MSBuild 4.0 and run on a 3.4 GHz desktop computer with 16G RAM. Its operating system is Windows 7 Professional. Simulation results are shown in Table I. From it, we know that the average runtime of W_{08} is much shorter than that of MBA. Although MBA is slower than W_{08}, it can detect the safety of markings of tested AMSs in 4.409 μs averagely. Besides, from Table I, we find that AR $\approx 0.00013 \times \text{AN}^2 \times |T|^2$ μs. It means that MBA can detect the safety of markings of a system with $|T| = 600$ and AN = 140 in 0.917 s averagely. Thus, MBA is capable of handling large scale AMSs.

TABLE I

SIMULATION RESULTS OF MBA AND W_{08}.

	(N_1, M_{10}) $\lvert T\rvert = 24$, AN = 5.33		(N_2, M_{20}) $\lvert T\rvert = 16$, AN = 6.62		(N_3, M_{30}) $\lvert T\rvert = 33$, AN = 5.68	
methods	W_{08}	MBA	W_{08}	MBA	W_{08}	MBA
NTM	5134	5134	19401	19401	20049	20049
TR (μs)	64	10692	316	28209	353	88396
AR (μs)	0.012	2.083	0.016	1.454	0.018	4.409

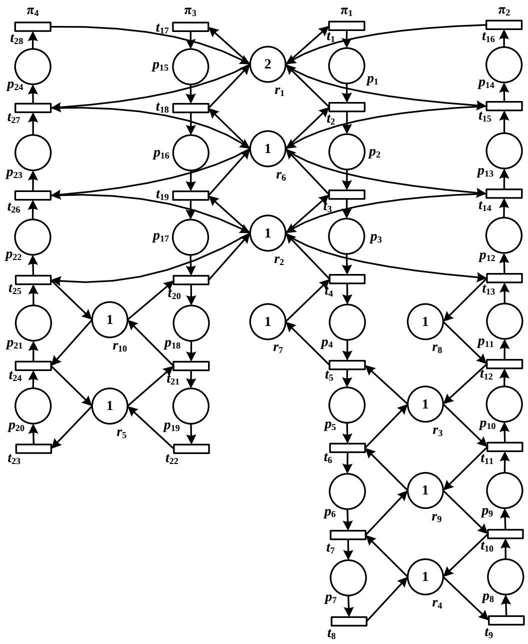

Fig. 7. Petri net model of a railway network system.

Now, we compare MBA with an existing method using the railway network system studied in [9]. It contains ten types of resources (tracks and stations), i.e., r_1-r_{10} and four types of trains, i.e., v_1-v_4. Their paths are $\pi_1 = r_1r_6r_2r_7r_3r_9r_4$, $\pi_2 = r_4r_9r_3r_8r_2r_6r_1$, $\pi_3 = r_1r_6r_2r_{10}r_5$, and $\pi_4 = r_5r_{10}r_2r_6r_1$, respectively. Suppose that the capacities of r_1-r_{10} are 2, 1, 1, 1, 1, 1, 1, 1, 1, and 1, respectively. Note that their capacities are smaller than those given in [9]. This is because the number of reachable markings under those capacities given in [9] is too large to be enumerated. The PN model of the system is given in Fig. 7. The deadlock prevention policy in [9] for this system contains nine constraints, denoted as follows:

1) $M(p_1) + M(p_{13}) + M(p_{15}) + M(p_{23}) \leq 2$;

2) $M(p_2) + M(p_{12}) + M(p_{16}) + M(p_{22}) \leq 1$;

3) $M(p_5) + M(p_9) \leq 1$;

4) $M(p_6) + M(p_8) \leq 1$;

5) $M(p_{17}) + M(p_{21}) \leq 1$;

6) $M(p_{18}) + M(p_{20}) \leq 1$;

7) $M(p_2) + M(p_{12}) + M(p_{16}) + M(p_{17}) + M(p_{21}) + M(p_{22}) \leq 2$;

8) $M(p_5) + M(p_6) + M(p_7) + M(p_8) + M(p_9) + M(p_{10}) \leq 1$;

9) $M(p_3) + M(p_{12}) + M(p_{17}) + M(p_{18}) + M(p_{19}) + M(p_{20}) + M(p_{21}) + M(p_{22}) \leq 1$.

The number of reachable markings under these nine constraints is 11169, while that under MBA is 48622. Obviously, MBA admits much more markings than the deadlock prevention method in [9] does.