Supplementary material to the paper "Predictability Verification of Fault Patterns in Labeled Petri Nets"

Xuya Cong, Zhenhua Yu, Maria Pia Fanti, Agostino Marcello Mangini, and Zhiwu Li

In this note, some experimental results are provided about the application of the proposed methodology in [1] for predictability verification of fault patterns in labeled Petri nets (LPNs). All the algorithms are coded in Matlab R2022a with some implementation details in [2], and the results are obtained by a laptop under Windows 10 operating system with Intel i7-10875H CPU Core 2.3 GHz and 16 GB RAM.

We consider two groups of unbounded LPNs with fixed number of places and transitions. For each group of the nets, we consider different initial markings and different fault patterns. To make the experiments meaningful, the considered LPNs and fault pattern nets should satisfy Assumptions A1–A3 in [1].

The net structure of the first group of LPNs with parameter k (the number of tokens in place p_1) is shown in Fig. 1, which has 11 places and nine transitions (four observable transitions and five unobservable transitions). There are four labels in this net structure. A fault pattern net for LPN 1 is shown in Fig. 2(a) and a fault pattern net for LPNs 2 and 3 is shown in Fig. 2(b). In particular, the fault pattern net in Fig. 2(a) recognizes the occurrence of either one of two possible transitions t_3 and t_9 in LPN 1, and the fault pattern net in Fig. 2(b) recognizes the occurrence of two consecutive t_3 without t_9 in LPNs 2 and 3.

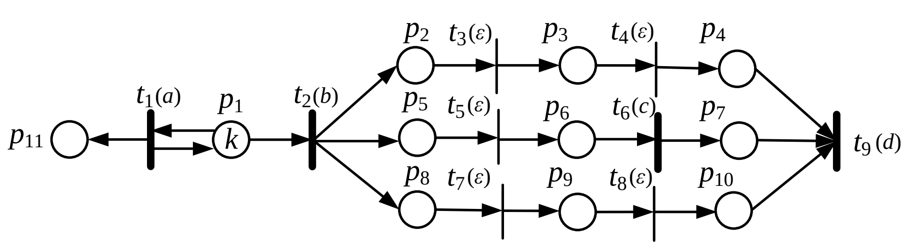

Fig. 1: Net structure of LPNs 1–3.

The net structure of the second group of LPNs with parameter k (the number of tokens in place p_7) is shown in Fig. 3, which has 16 places and 16 transitions (six observable transitions and 10 unobservable transitions). There are six labels in this net structure. The fault pattern nets for LPNs 4, 5, and 6 are shown in Figs. 4(a), 4(b) and 4(c), respectively. In particular, the fault

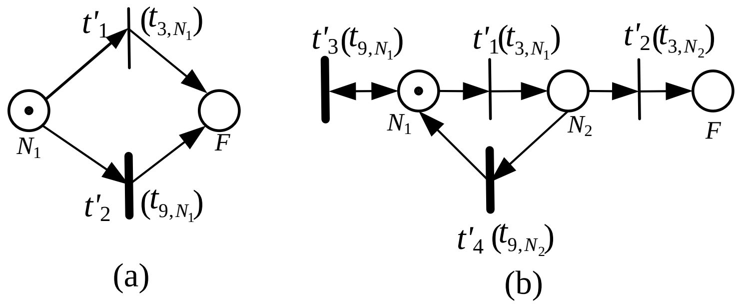

Fig. 2: (a) A fault pattern net for LPN 1 and (b) a fault pattern net for LPNs 2 and 3.

pattern net in Fig. 4(a) recognizes the occurrence of two consecutive t_9 without t_{13} in LPN 4, the fault pattern net in Fig. 4(b) recognizes the occurrence of three consecutive t_9 without t_{13} in LPN 5, and the fault pattern net in Fig. 4(c) recognizes the occurrence of four consecutive t_9 without t_{13} in LPN 6.

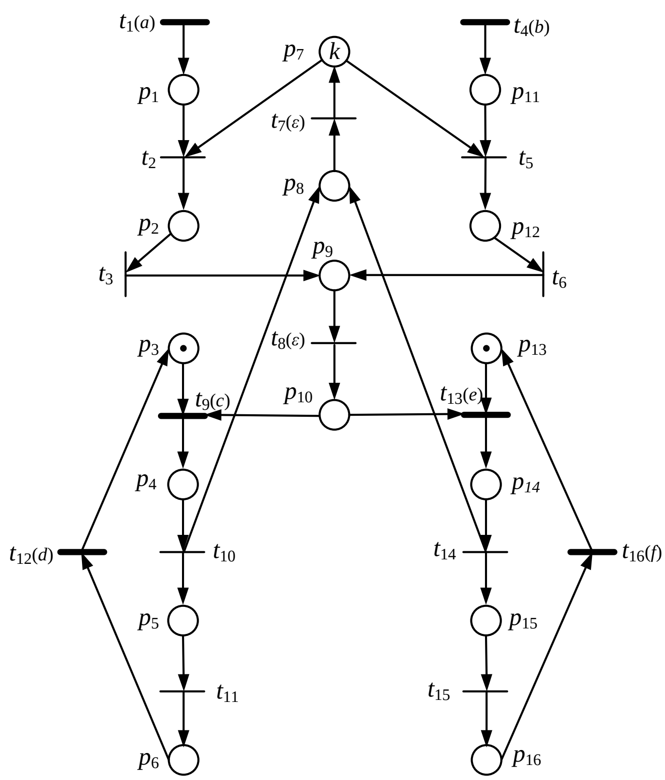

Fig. 3: Net structure of LPNs 4–6.

Table I shows the performance of the proposed method applied to LPNs 1–6: column two

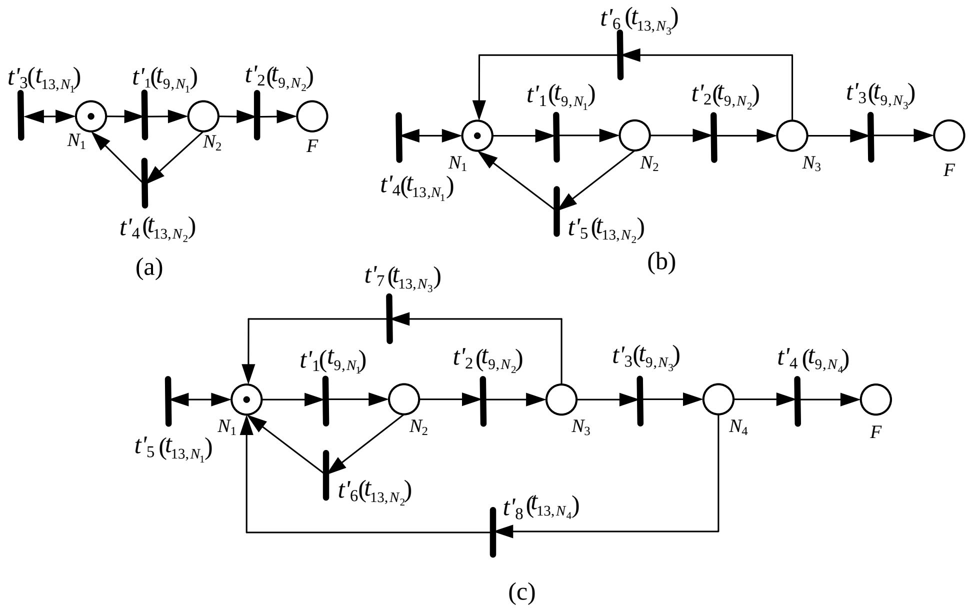

Fig. 4: (a) A fault pattern net for LPN 4, (b) a fault pattern net for LPN 5, and (c) a fault pattern net for LPN 6.

TABLE I: Experimental results for the LPNs 1–6

	k	$\lvert\tilde{P}\rvert$	$\lvert\tilde{T}\rvert$	$\lvert\tilde{\mathcal{M}}_B\rvert$	$\lvert\tilde{\mathcal{M}}\rvert$	Predictability
LPN1	1	26	22	6	606	Yes
LPN2	2	28	30	37	82047	No
LPN3	3	28	30	100	1463478	No
LPN4	1	38	49	180	81883	No
LPN5	2	40	59	334	2176315	No
LPN6	3	42	73	589	o.t.	No

represents the number of tokens in p_1 for LPNs 1–3 and that of tokens in p_7 for LPNs 4–6, columns three and four represent the number of places and transitions of the fault pattern predictor nets (FP^{2}Ns) that associate with LPNs 1–6 and their corresponding fault pattern nets, respectively. Column five is the number of basis markings in the basis fault pattern predictor graph of the corresponding FP^{2}N by using Algorithm 1 in [1], column six is the number of markings in the coverability graph (CG) of the corresponding FP^{2}N by using the PN analysis software TINA [3], and the last column shows whether the LPN is predictable w.r.t. the given fault pattern. Note that o.t. means that the number of markings in the CG cannot be computed by using TINA due to the problem of memory overflow. From the experimental results shown in Table I, we can see that the proposed method is more efficient than the one generating the full state space.

REFERENCES

[1] X. Y. Cong, Z. H. Yu, M. P. Fanti, A. M. Mangini, and Z. W. Li, "Predictability verification of fault patterns in labeled Petri nets," submitted to *IEEE Trans. Autom. Control*.

[2] Matlab toolbox for fault pattern predictability verification. [Online]. Available: https://github.com/congxuya/Fault-pattern-predictability

[3] TINA (Version 3.5.0): Time Petri Net Analyzer 2019. [Online]. Available: http://projects.laas.fr/tina//download.php