Current-state opacity verification in discrete event systems using an observer net

Abdeldjalil Labed[1]**, Ikram Saadaoui**[1]**, Naiqi Wu**[1]**, Jiaxin Yu**[2]**, and Zhiwu Li**[1,*]

[1]Macau University of Science and Technology, Institute of Systems Engineering, Taipa, 999078, Macau SAR, China
[2]Hitachi Building Technology (Guangzhou) Co.,Ltd., GuangZhou, 510670, China
[*]Corresponding author

ABSTRACT

1 Experimental study data

The effectiveness of the approach developed is investigated by comparing it with the opacity verification methods recently proposed in the literature to demonstrate their advantages and limitations. It is based on the CPU time in seconds of a desktop computer running under the operating system Windows 10 with I7.4 CPU 3.40 GHz, 32 GB memory.

To do so, we apply the proposed on-line algorithm to a larger version of the LPN (presented in the paper), where the initial marking is $M_0 = 15p_1$. Let $S = \{(4p_3 + 11p_5), (5p_1 + 10p_5), (8p_1 + 3p_5 + 4p_6), (11p_1 + 2p_2 + 2p_3), (13p_1 + 2p_5), (14p_1 + p_5), (15p_1)\}$ be a secret. We use the standard opacity verification approach, which consists of computing the RG and converting the obtained RG into its equivalent DFA. After that, for each state of the observer (i.e., $\mathcal{C}(w)$), we check whether it is fully included in the secret or not.

Table 1. Performance of Algorithm 2

Event occurrence	time (s)	CSO
observable event a occurs	7.4×10^{-2}	Y
observable event a occurs	6.7×10^{-1}	Y
observable event b occurs	1.1×10^{1}	Y
observable event a occurs	2.6×10^{1}	Y
observable event a occurs	3.9×10^{1}	Y
observable event b occurs	5.7×10^{1}	N

Now, let us implement the on-line algorithm to the same example. Table 1 shows the performance of Algorithm 2. The first column represents the occurrence of an event. The second indicates the time (CPU seconds) required to run an observer net when an event occurs. The third column shows the algorithm's outputs when an event occurs: "Y" if the observation is CSO and "N" otherwise. From Table 1, it is known that the observed event sequence $w = aabaab$ is non-current-state opaque wrt S.

Examine the LPN system G (presented in the paper) with initial marking $M_0 = kp_1$, where $k \in \mathbb{N}$. Accordingly, we consider a family of nets rather than a single LPN, which is parameterized by the initial marking. Table 2 compares the colored estimator construction using the observer net, i.e., EST_Φ, as shown in Algorithm 3, with the standard approach for observer construction, i.e., DFA construction using the RG of an LPN. The first column shows the value of k. The number of reachable markings is represented in Column 2. Columns 3 and 4 give the number of states of the standard observer and its construction time, respectively. Finally, Columns 5 and 6, respectively, expose the number of states in the observer net and the time of its construction. We use the notation "o.t." (out of time) to indicate that the computation takes more than three hours to complete.

Table 2. Performance of Algorithm 3

k	$\lvert RG\rvert$	$\lvert Obs_{RG}\rvert$	time (s)	$\lvert EST_{\Phi}\rvert$	time (s)
2	21	6	0.8×10^1	6	0.2×10^1
4	126	15	1.4×10^2	15	0.4×10^2
8	1287	39	4.7×10^3	39	1.4×10^3
10	3003	-	o.t.	54	5.2×10^3
20	53130	-	o.t.	159	1.0×10^4
40	-	-	o.t.	-	o.t.

Acknowledgments

This work is partially supported by the Guangzhou Innovation and Entrepreneurship Leading Team Project Funding under Grant No. 202009020008.

Abdeldjalil labed, Ikram Saadaoui, Naiqi Wu and Zhiwu Li are with the Institute of Systems Engineering, Macau University of Science and Technology, Taipa 999078, Macau SAR, China (e-mails: labed994@gmail.com, saadaoui.ikrame@gmail.com, nqwu@must.edu.mo, zwli@must.edu.mo). Jiaxin Yu is with the Hitachi Building Technology (Guangzhou) Co.,Ltd., GuangZhou 510670, China (e-mail: yujiaxin@hitachi-helc.com).