Verification of the Boundedness Property in a Petri Net-Based Specification of the Control Part of Cyber-Physical Systems

Marcin Wojnakowski(✉) and Remigiusz Wiśniewski

Institute of Control and Computation Engineering, University of Zielona Góra,
65-516 Zielona Góra, Poland
{M.Wojnakowski,R.Wisniewski}@issi.uz.zgora.pl

Abstract. A method of analysis of a control part of the cyber-physical system described by a Petri net is presented in the paper. In particular, boundedness of the system is examined. Contrary to other well-known techniques, the proposed idea does not require obtaining of all place invariants, nor computation of all reachable states in the net. Therefore, it is possible to check the boundedness of a net in a more effective and efficient way, compared to the traditional, well-known methods. Furthermore, the proposed algorithm has been examined experimentally with a set of 243 benchmarks (Petri nets). The research results show the high efficiency of the proposed method, since a solution was found even for such nets where popular techniques were not able to analyse boundedness of the system. Finally, the presented idea is illustrated by a case-study real-life example.

Keywords: Boundedness · Petri nets · Control part of cyber-physical systems · Invariants · Linear algebra

1 Introduction and Problem Formulation

Petri nets are popular state-transition systems that allow for comfortable and easy specification of concurrent systems [1–4]. They offer the possibility of graphical modelling, as well, as a wide opportunity of analysis techniques [3, 5, 6]. Recently, Petri nets have become particularly popular in the modelling of the control part of the cyber-physical systems [7–9]. A cyber-physical system (CPS) [7, 10] combines computation with physical routes. The behaviour of a CPS is defined by two parts: the cyber and the physical parts [11]. Such systems join physical processes, networks and the computational modules of the system. A CPS finds applications in various fields of human life, for instance medical systems [12], vehicular systems [13], power electronic converters [14] and smart homes [15].

This work focuses on the analysis of the control part of the cyber-physical systems, which plays the computational part of the system. A Petri net-based approach benefits the verification of the design even at the specification phase, allowing a reduction in time and costs of the design of CPS [3, 8]. A very important property of a Petri net-based

L. M. Camarinha-Matos et al. (Eds.): DoCEIS 2021, IFIP AICT 626, pp. 83–91, 2021.
https://doi.org/10.1007/978-3-030-78288-7_8

system is boundedness [1, 2, 5]. Various design methods require this property in their inputs (cf. [6, 8, 9, 16, 17]). Furthermore, boundedness is an essential in case of systems that are oriented on the implementation in the hardware (for example within the field programmable gate arrays, FPGAs).

The two most popular methodologies for boundedness examination of a Petri net are one which applies linear algebra (place invariants computation [5, 18]), and another which involves reachability tree exploration [19]. However, they are seriously limited, since the number of invariants (or reachable states, respectively) can be exponential [2, 4, 5]. So-called *state explosion problem* may be a real challenge to the designer during the analysis of the system. Usually, in such a case the solution is not found within the assumed time due to the exponential computational complexity.

In the paper a technique for the boundedness verification of the Petri net-based CPS is proposed. The method does not involve computation of all invariants in the system, thus it is more efficient and effective compared to the most popular techniques. The idea of the presented solution is based on the computation of the reduced set of the place invariants. The main contributions are as follows:

- A method that allows for the boundedness verification of the control part of a Petri net-based CPS is proposed.
- The presented technique allows for the efficient boundedness verification of the system, which means that the solution is found in the assumed time.
- The idea has been validated and verified experimentally in order to confirm its efficiency and effectiveness.
- The algorithm is explained by a case-study real-life example of a CPS.

2 Petri Nets in Applied Artificial Intelligence Systems

Application of various Petri net-based aspects can be found in the artificial intelligence systems [23]. In particular, boundedness property may play important role in analysis and verification of such systems. Let us briefly present the possible applications that show relations between Petri nets and Applied AI.

Analysis and modelling aspects of multi-agent systems are considered in [20]. The paper studies several important properties of Petri net-based systems, including boundedness and liveness. As stated by the Authors, those features are applicable in the modelling of multi-agent systems. In particular, the system is verified against the deadlocks by analysis of boundedness and liveness properties.

Application of Petri nets for intelligent control and supervision is shown in [21]. The Authors propose a modelling tool called Continuous Fuzzy Petri Net (CFPN). Such a net can be used for the improvement of the performance and optimization of the system. Moreover, CFPNs are applicable in fault tolerance and diagnostics (e.g., to help the operator in the controlling and monitoring of thousands of actuators and sensors). The idea is explained by an example of a water treatment plant.

The overview of scheduling, planning and control of manufacturing systems with the application of AI-based search methods and Petri nets is presented in [22]. The paper focuses on the various aspects, starting with Petri nets and their utilization in the modelling of manufacturing systems. Then, scheduling techniques are presented, including combination of Petri nets and AI-based heuristic search methods.

3 Definitions and Notations

The presented definitions correspond to the notations shown in [2, 4, 5, 24, 25].

Definition 1. *A Petri net N* is a 4-tuple: $N = (P, T, F, M_0)$, where P is a set of places, T is a set of transitions, $F \subseteq (P \times T) \cup (T \times P)$ is a set of arcs, $M_0 : P \to \mathbb{N}$ is an initial marking.

Definition 2. *An incidence matrix* $A_{|T|\times|P|}$ of a Petri net $N = (P, T, F, M_0)$ is given by:

$$\mathrm{a_{ij}} = \begin{cases} -1, \left(\mathrm{p_j}, \mathrm{t_i}\right) \in F \\ 1, \left(\mathrm{t_i}, \mathrm{p_j}\right) \in F \\ 0, \textit{otherwise} \end{cases},$$

where cell a_{ij} of matrix A refers to transition t_i and place p_j.

Definition 3. *A place invariant (p-invariant)* of a Petri net $N = (P, T, F, M_0)$ is an integer vector such that $A\vec{x} = 0$.

Definition 4. A Petri net $N = (P, T, F, M_0)$ is *covered* by place invariants if every place $p \in P$ belongs to at least one p-invariant.

Definition 5. A Petri net $N = (P, T, F, M_0)$ is said to be *bounded* if there is no marking (state) M_n such that any place $p \in P$ contains more than a finite number of tokens. A Petri net N bounded for any finite initial marking M_0 is said to be *structurally bounded*.

Theorem 1. A Petri net $N = (P, T, F, M_0)$ is structurally bounded if it is covered by p-invariants [19].

4 The Idea of the Proposed Method

This section presents the idea of the proposed technique. Firstly, we will show the main steps of the proposed method, supplemented by an adequate description. Next, the case study example of the boundedness verification of the real-life cyber-physical system is presented.

The proposed method includes the following steps:

1. Initialization:
   a) Read incidence matrix $A_{|T|\times|P|}$ of Petri net $N = (P, T, F, M_0)$ that describes the control part of the cyber-physical system.

   b) Form the unit matrix $Q = [D_{|P|\times|P|}|A^{\mathrm{T}}_{|T|\times|P|}]$, where D is an identity matrix, and $A^{\mathrm{T}}_{|T|\times|P|}$ is the transposed incidence matrix of N.
   c) Initialize the place invariants cover: $C = \emptyset$.

2. Searching for the place-invariants cover: for each column $t \in T$ in $A^{\mathrm{T}}_{|T|\times|P|}$:

   a) Find all pair of rows that annul the j-th transition (column) of A^{T} and add them to the matrix Q.
   b) Find all rows which the intersection with the j-th transition (column) is not equal to 0 and delete them from Q.
   c) Find all rows that cover binary the other ones and delete them from Q.
   d) Boundedness verification: for each row r of A^{T} such that all entries of r are equal to 0:

      - add place invariant I that refer to the row r in D to the set C: $C = C \cup \{I\}$ (i.e., values of I refer directly to the row r in the matrix D),
      - examine, whether C covers all places in the net:

        - break if C covers all places, the system is structurally bounded;
        - otherwise, execute the algorithm from step 2(a).

3. Boundedness verification:

   - if the net is covered by place invariants, **the system is structurally bounded**,
   - otherwise, its **boundedness of the system is not determined**.

The presented method involves linear algebra. It is based on the technique initially proposed in [18], however it does not require computation of all place invariants in the Petri net. The algorithm works as follows. Initially, the unit matrix Q of matrices D and A^{T} is formed. Matrix D is initially equal to the identity matrix, while A^{T} denotes the transposed incidence matrix of the Petri net. Next, the method searches for the invariants by transformations of the matrix Q. In particular, subsequent transitions are examined in order to zeros matrix A^{T}. Meanwhile, matrix D holds partially obtained invariants. If any row of A^{T} is completely zeroed (that is, all its entries are equal to 0), the proper invariant can be obtained from matrix D. The algorithm verifies existence of new invariants at each stage, add adds them to the set C. The method finishes, once the set C covers all the places.

Let us now explain the proposed algorithm with a real-life example. Figure 1 shows a Petri net-based control system responsible for managing a multi-robot, initially presented in [26]. There are nine places and six transitions in the net, denoted by $p_1, \ldots, p_9$ and $t_1, \ldots, t_6$, respectively. The system involves pick-and-place operations in order to transfer or obtain parts by two robot arms. Places $p_1, \ldots, p_3$, and transitions $t_1, \ldots, t_3$ refers to the activities of robot first arm. Similarly, places $p_4, \ldots, p_6$, and transitions $t_4, \ldots, t_6$ are related to the second arm. The presented example focuses on the collision-free movements. Therefore, only one robot arm is able to access the workspace at a time.

The activities in common workspace are represented by places p_3 and p_6. The collision-free movements are secured by the mutual exclusion technique, and involvement of place p_7 (places p_8 and p_9 are used as additional buffers).

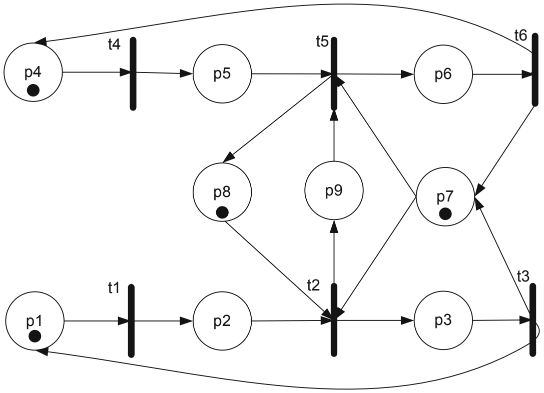

Fig. 1. A multi-robot controller specified by a Petri net.

Let us now examine the boundedness of the system with the proposed method. According to the algorithm, initially the unit matrix $Q = [D|A^T]$ is formed (Table 1, left). Next, the matrix is transformed, while the subsequent transitions are examined. In the presented example, only three transitions (out of six) are required to be processed to obtain the solution. Table 1 (right) shows the unit matrix after examination of the third transition. Note that four rows of A^T are already zeroed. They refer to the place invariants formed in the matrix D (marked by blue boxes). Clearly, those invariants cover all places in the net. The algorithm terminates its execution with the result that the system is bounded.

Table 1. Matrix $Q = [D|A^T]$ before the transformation (left) and after the transformation (right).

p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9	t_1	t_2	t_3	t_4	t_5	t_6
1	0	0	0	0	0	0	0	0	-1	0	0	0	1	0
0	1	0	0	0	0	0	0	0	1	-1	0	0	0	0
0	0	1	0	0	0	0	0	0	0	1	0	0	-1	0
0	0	0	1	0	0	0	0	0	0	0	-1	0	0	1
0	0	0	0	1	0	0	0	0	0	0	1	-1	0	0
0	0	0	0	0	1	0	0	0	0	0	0	1	0	-1
0	0	0	0	0	0	1	0	0	0	-1	0	-1	1	1
0	0	0	0	0	0	0	1	0	0	-1	0	1	0	0
0	0	0	0	0	0	0	0	1	0	1	0	-1	0	0

p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9	t_1	t_2	t_3	t_4	t_5	t_6
0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	1	1	0	0	0	0	0	0	0	0
0	0	2	0	0	0	1	1	0	0	0	0	0	-1	1
1	1	0	0	0	1	0	0	1	0	0	0	0	1	-1
0	0	0	1	1	1	0	0	0	0	0	0	0	0	0
0	0	1	1	1	0	0	1	0	0	0	0	0	-1	1

5 Experiments

The proposed method was verified experimentally. Its effectiveness and efficiency was compared to the popular technique of place invariants computation (denoted as an *exact algorithm*), initially shown in [18]. Both methods were examined in terms of their runtime and obtained results (covering of the net by p-invariants). The set of benchmarks includes 243 Petri nets, modelling real and hypothetical cyber-physical systems and control systems. Their description can be found on the websites: http://gres.uninova.pt and http://hippo.iee.uz.zgora.pl. The experiments were perfor-med on the dedicated computational server: Intel® Xeon® Platinum 8160 @2.1 GHz processor, 16 GB of RAM. The results for selected benchmarks are shown in Table 2. The particular columns contain the following values: *Name of the system* – the name of the system described by a Petri net, |*P*| - the number of places, |*T*| - the number of transitions, *covered* – whether the system is covered by place invariants according to the algorithm, *runtime* – the execution time of the algorithm in milliseconds.

Table 2. Exemplary results of the experiments.

Name of the system (Petri net)	|*P*|	|*T*|	Exact method		Proposed method	
			Covered	Runtime [ms]	Covered	Runtime [ms]
traffic_light_v2	4	3	Yes	0.424	Yes	0.415
pn_silva_05e	4	4	No	0.439	No	0.635
esparza2	15	13	Yes	3.366	Yes	1.061
2pusher	15	18	No	10.743	No	6.536
silva5	16	8	Yes	2.660	Yes	1.496
hulgaard1	19	12	Yes	27.805	Yes	3.275
ConsistentExample	29	26	No	42986.800	No	5752.120
zuberek1	30	22	Yes	22.324	Yes	3.273
crossroadSM_FPGA	32	12	Yes	807879.000	Yes	183.985
zuberek5	41	31	n/a	Timeout	Yes	45.045
cn_crr7	56	15	n/a	Timeout	Yes	155.278
cn_crr25	200	51	n/a	Timeout	Yes	78954.828

It can be observed that for small systems, containing a few places and transitions, (such as *traffic_light_v2* or *pn_silva_05e*) both methods are efficient. It can be even noticed that the exact method computes the result faster than the proposed algorithm. However, in case of more complicated systems (such as *ConsistentExample*, *crossroadSM_FPGA*), the difference is notable. The proposed method was able to compute the solution within a few seconds, while the runtime of the exact method is much longer (even more than 13 min in case of *crossroadSM_FPGA*). Finally, a huge difference can be noticed in case of complex systems (*zuberek5*, *cn_crr7*, *cn_crr25*). For such systems, the exact method was not able to compute results due to the state explosion problem (the method was stopped after one hour, and denoted as “timeout” in the table). In contrast, the proposed method found the result for the worst case (*cn_crr25*) in less than one and a half minute. Finally, let us note that module *crosroadSM_FPGA* describes the real-life cyber-physical system (collision free crossroad for cars and pedestrians). It was implemented and partially reconfigured within the programmable device, thus the boundedness property was essential.

The performed experiments proved the effectiveness and efficiency of the proposed technique. The results obtained by both methods were the same for all examined benchmarks (for which the result for the exact method was obtained). This provides experimental validation of the correctness of the method. Furthermore, the runtime of the proposed method confirms its very high efficiency, since the result for the worst-case example was obtained in less than one and half a minute.

6 Conclusions

The design process of the control part of cyber-physical systems involves several aspects. One of them refers to the proper specification and further formal verification. In the paper an analysis technique of the boundedness property is proposed. The presented solution is based on the existing solutions and applies transformations of the incidence matrix of the system. Contrary to the other, most popular analysis techniques, the introduced method does not require computation of all place invariants in the system. The performed experiments proved its efficiency and effectiveness.

On the other hand, there is a limitation of the presented method. The boundedness of the Petri net is guaranteed only for those systems that are covered by place invariants. Otherwise, the system might be unbounded, but the final result remains unsolved. However, it should be underlined that such a situation (the net being bounded but not covered by place invariants) is rather rare (about 5.5% of the examined benchmarks). Nevertheless, this aspect is planned for further enhancement of the algorithm. Moreover, plans for future research include analysis of well-formed nets (safeness, liveness), which are the key properties of Petri net-based description of the control part of cyber-physical systems.

Acknowledgments. This work is supported by the National Science Centre, Poland, under Grant number 2019/35/B/ST6/01683.

References

1. Girault, C., Valk, R.: Petri Nets for Systems Engineering: A Guide to Modeling, Verification, and Applications. Springer, Heidelberg (2003). https://doi.org/10.1007/978-3-662-05324-9
2. Karatkevich, A.: Dynamic Analysis of Petri Net-Based Discrete Systems. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-71560-3
3. Zhou, M., Wu, N.: System Modeling and Control with Resource-Oriented Petri Nets, 1st edn. CRC Press, Boca Raton (2009)

4. Murata, T.: Petri nets: properties, analysis and applications. Proc. IEEE **77**(4), 541–580 (1989). https://doi.org/10.1109/5.24143
5. Wiśniewski, R.: Prototyping of Concurrent Control Systems Implemented in FPGA Devices. Springer, Switzerland (2017). https://doi.org/10.1007/978-3-319-45811-3
6. Wisniewski, R., Wisniewska, M., Jarnut, M.: C-exact hypergraphs in concurrency and sequentiality analyses of cyber-physical systems specified by safe Petri nets. IEEE Access **7**, 13510–13522 (2019). https://doi.org/10.1109/access.2019.2893284
7. Yu, Z., Zhou, L., Ma, Z., El-Meligy, M.A.: Trustworthiness modeling and analysis of cyber-physical manufacturing systems. IEEE Access **5**, 26076–26085 (2017)
8. Grobelna, I., Wiśniewski, R., Wojnakowski, M.: Specification of cyber-physical systems with the application of interpreted nets. In: Proceedings of the IECON' 19 - 45th Annual Conference of the IEEE Industrial Electronics Society, Lisbon, Portugal, pp. 5887–5891 (2019)
9. Wiśniewski, R., Bazydło, G., Szcześniak, P., Wojnakowski, M.: Petri net-based specification of cyber-physical systems oriented to control direct matrix converters with space vector modulation. IEEE Access **7**, 23407–23420 (2019). https://doi.org/10.1109/access.2019.2899316
10. Lee, E.A.: Cyber physical systems: design challenges. In: 2008 11th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC), May 2008, pp. 363–369 (2008). https://doi.org/10.1109/isorc.2008.25
11. Lee, E.A., Seshia, S.A.: Introduction to Embedded Systems: A Cyber-Physical Systems Approach, 2nd edn. The MIT Press, Cambridge (2016)
12. Dey, N., Ashour, A.S., Shi, F., Fong, S.J., Tavares, J.M.R.S.: Medical cyber-physical systems: a survey. J. Med. Syst. **42**(4), 74 (2018). https://doi.org/10.1007/s10916-018-0921-x
13. Jia, D., Lu, K., Wang, J., Zhang, X., Shen, X.: A Survey on platoon-based vehicular cyber-physical systems. IEEE Commun. Surv. Tutor. **18**(1), 263–284 (2016). https://doi.org/10.1109/comst.2015.2410831
14. Wiśniewski, R., Bazydło, G., Szcześniak, P.: Low-cost FPGA hardware implementation of matrix converter switch control. IEEE Trans. Circuits Syst. II Express Briefs **66**(7), 1177–1181 (2019). https://doi.org/10.1109/tcsii.2018.2875589
15. Wiśniewski, R., Karatkevich, A., Adamski, M., Costa, A., Gomes, L.: Prototyping of concurrent control systems with application of Petri nets and comparability graphs. IEEE Trans. Control Syst. Technol. **26**(2), 575–586 (2018). https://doi.org/10.1109/tcst.2017.2692204
16. Clempner, J.: An analytical method for well-formed workflow/Petri net verification of classical soundness. Int. J. Appl. Math. Comput. Sci. **24**(4), 931–939 (2014). https://doi.org/10.2478/amcs-2014-0068
17. Li, B., Khlif-Bouassida, M., Toguyéni, A.: On–the–fly diagnosability analysis of bounded and unbounded labeled Petri nets using verifier nets. Int. J. Appl. Math. Comput. Sci. **28**(2), 269–281 (2018). https://doi.org/10.2478/amcs-2018-0019
18. Martínez, J., Silva, M.: A simple and fast algorithm to obtain all invariants of a generalised Petri net. In: Girault, C., Reisig, W. (eds.) Application and Theory of Petri Nets. Informatik-Fachberichte, vol. 52, pp. 301–310. Springer, Heidelberg (1982). https://doi.org/10.1007/978-3-642-68353-4_47
19. Reisig, W.: Nets consisting of places and transistions. In: Reisig, W. (ed.) Petri Nets. EATCS Monographs on Theoretical Computer Science, vol 4, pp. 62–76. Springer, Heidelberg (1985). https://doi.org/10.1007/978-3-642-69968-9_6
20. Celaya, J.R., Desrochers, A.A., Graves, R.J.: Modeling and analysis of multi-agent systems using Petri nets. In: 2007 IEEE International Conference on Systems, Man and Cybernetics, pp. 1439–1444, October 2007. https://doi.org/10.1109/icsmc.2007.4413960

21. Pang, G.K.H., Tang, R., Woo, S.S.: Intelligent control and supervision based on fuzzy Petri nets. In: Tzafestas, S.G. (ed.) Methods and Applications of Intelligent Control. Microprocessor-Based and Intelligent Systems Engineering, vol. 16, pp. 217–243. Springer, Dordrecht (1997). https://doi.org/10.1007/978-94-011-5498-7_8
22. Cang, S., Yu, H.: Overview of modelling, scheduling, planning, and control using Petri net representation and AI search. In: Advanced Design and Manufacture to Gain a Competitive Edge, London, pp. 397–406 (2008). https://doi.org/10.1007/978-1-84800-241-8_41
23. Edelkamp, S., Jabbar, S.: Action planning for directed model checking of Petri nets. Electron. Notes Theor. Comput. Sci. **149**(2), 3–18 (2006). https://doi.org/10.1016/j.entcs.2005.07.023
24. Wisniewski, R., Wojnakowski, M., Stefanowicz, Ł.: Safety analysis of Petri nets based on the SM-cover computed with the linear algebra technique. In: AIP Conference Proceedings, vol. 2040, no. 1, p. 080008, November 2018. https://doi.org/10.1063/1.5079142
25. Wiśniewski, R., Karatkevich, A., Stefanowicz, Ł., Wojnakowski, M.: Decomposition of distributed edge systems based on the Petri nets and linear algebra technique. J. Syst. Archit. **96**, 20–31 (2019). https://doi.org/10.1016/j.sysarc.2019.01.015
26. Zurawski, R., Zhou, M.: Petri nets and industrial applications: a tutorial. IEEE Trans. Ind. Electron. **41**(6), 567–583 (1994). https://doi.org/10.1109/41.334574