A Methodological Framework for Event Signal Reconstruction in Net Condition/Event Systems

Zhenglong Han[1] Jiafeng Zhang [2] and Kamel Barkaoui [3]

Abstract— **Net Condition/Event Systems (NCES) offer a formal framework for modeling and verification of modular discrete-event systems, explicitly capturing condition constraints, event-triggering mechanisms, and inter-subsystem dependencies. Accurate inference of event signals is critical for monitoring, diagnosis, and optimization in applications such as industrial automation, unmanned aerial vehicle coordination, and intelligent traffic control. This paper presents a novel framework for reconstructing event signals from NCES reachability graphs, addressing an open problem in NCES-based modular discrete event system analysis. The proposed method can effectively deduce the transmitters and receivers of event signals based on the differences in the reachability graphs of NCES with and without event signals, and clarify the AND/OR processing modes for multiple event signals at the receiver end under both S-mode (maximal single spontaneous) and N-mode (arbitrary maximal) firing rules.**

I. Introduction

Petri nets provide a well-established formal framework for modeling and analyzing discrete-event systems (DES)[1], offering a clear representation of concurrency, synchronization, and causality. To explicitly capture logical conditions and event-driven interactions, Condition/Event nets (CE-nets)[2] extend classical Petri nets by introducing condition signals and event signals, which may exist both within and across modules.

Net Condition/Event Systems (NCES) further specialize CE-nets toward modular DES modeling. In an NCES, signals are restricted exclusively to inter-module connections; no condition or event signals appear inside any basic module. This structural constraint preserves the local autonomy of individual modules while enabling coordinated behavior through explicit signal interactions. Owing to their expressive power and structural clarity, NCES have been widely adopted for modeling, analysis, and verification in domains such as industrial automation, distributed control, and intelligent systems. Key application areas include industrial production lines and intelligent transportation systems [3], [4], [5], [6].

Event signals are defined as signals from sender transitions to receiver transitions in an NCES. Prior NCES-based research, including applications in controller synthesis, fault diagnosis, and verification of IEC 61499-based systems, typically assumes that event signals are known a priori [7], [8], [9], [10], [11]. However, the problem of reconstructing event signals from observed system behaviors remains open.

This paper investigates the reconstruction problem of event signals in NCES. Based on differences between the reachability graph of a system with event signals and that of a system without event signals, the possible senders and receivers of the event signals are deduced, and the multi-event signal processing mode at the receiver side is identified. The objective is to ensure that after adding the reconstructed event signals to the system without event signals, the resulting reachability graph is fully consistent with that of the original system containing event signals.

The remainder of this paper is organized as follows. Necessary preliminaries of NCES are reviewed in Section II. The event signal reconstruction problem in NCES is formally defined in Section III, where it is categorized into four distinct cases based on available information and firing rules. Corresponding reconstruction methods for each case are presented in Section IV. Four examples are provided in Section V to illustrate the proposed approach. Finally, conclusions along with limitations and future research directions are discussed in Section VI.

II. Preliminaries

A basic module of an NCES is a general Petri net, while inter-module interactions are realized exclusively through condition and event signals. An NCES is defined as

$$\Gamma = (\mathcal{D}, C, E, V, M_0), \tag{1}$$

where $\mathcal{D} = \{\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_n\}$ is a finite set of n basic modules of Γ, C is the set of condition signals among basic modules, E is the set of event signals among basic modules, V specifies multi-event signal processing modes of transitions, and M_0 denotes the initial marking (state) of Γ.

A basic module $\mathcal{D}_i$ $(i = 1, \ldots, n)$ of Γ is a general Petri net [12], [13] defined by a 4-tuple

$$\mathcal{D}_i = (P_i, T_i, F_i, W_i). \tag{2}$$

Let $\mathbf{P} = \bigcup_{i=1}^{n} P_i$ and $\mathbf{T} = \bigcup_{i=1}^{n} T_i$. The set of condition signals is defined as $C \subseteq \mathbf{P} \times \mathbf{T}$ where each condition signal represents a dependency from a place in a basic module to a transition in another basic module. Graphically, a condition signal is denoted by a line with a dark dot at the end from a place to a transition. The set of event signals is defined as $E \subseteq \mathbf{T} \times \mathbf{T}$, representing asymmetric event dependencies

[1]Zhenglong Han is a Master's student with the School of Mechanical and Electrical Engineering, Xidian University, Xi'an 710071, China zhenglonghan1@gmail.com

[2]Jiafeng Zhang is with the School of Mechanical and Electrical Engineering, Xidian University, Xi'an 710071, China zhangjiafeng@xidian.edu.cn

[3]Kamel Barkaoui is with the Computer Science Department, Conservatoire National des Arts et Métiers, France kamel.barkaoui@cnam.fr

between transitions in different basic modules. An event signal is graphically denoted by an arrowed folding line from one transition to another. The mapping $V : \mathbf{T} \rightarrow \{\wedge, \vee\}$ assigns a multi-event signal processing mode to each transition, where $\wedge$ and $\vee$ denote AND-type and OR-type signal processing modes, respectively. $M : \mathbf{P} \rightarrow \mathbb{N}$ is called a marking (state) of Γ. For all $p \in P$, $M(p) \in \mathbb{N}$ denotes the number of tokens held by p at M. The net structure of an NCES $\Gamma = (\mathcal{D}, C, E, V, M_0)$ is defined by $N = (\mathcal{D}, C, E, V)$.

For an event signal $(t, t') \in E$, when the firing conditions of transition t are satisfied, an event signal is sent to transition t'. Transition t is called a *trigger transition* of t', and t' is called a *forced transition* of t. The set of trigger transitions of t is denoted by $^{\sim}t$, and the set of forced transitions of t is denoted by $t^{\sim}$. If $^{\sim}t = \emptyset$, transition t is called a *spontaneous transition*; otherwise, it is a *forced transition*. The sets of spontaneous and forced transitions are denoted by T_s and T_f, respectively.

In an NCES, system state changes are driven by *steps*, where a *step* $u \subseteq \mathbf{T}$ is a non-empty and signal-complete set of transitions [12], [14]. Two firing rules are considered in this work:

1) **N-mode:** Normal firing rule with arbitrary maximal steps. In this case, an executable step is constructed by first selecting a nonempty enabled set of spontaneous transitions, and then adding as many forced transitions as possible that are triggered by transitions within the step.

2) **S-mode:** Maximal single spontaneous transition steps. In this case, all steps containing more than one spontaneous transition are excluded.

The executable (firable) step set at state M in an NCES is denoted as $\mathrm{Ena}(M)$. A step u is executable (or firable) at a marking M, denoted as $M[u\rangle$. The notation $M[u\rangle M'$ indicates that the state M' is reachable from state M by executing the step u. The reachable state set of an NCES $\Gamma = (N, M_0)$ is denoted by $R(N, M_0)$. The computation of executable steps at a state and state transformation rules of an NCES are described in detail in [15].

Let U be the set of all executable steps of Γ. Taking M_0 as a root node, the reachability graph of an NCES is obtained, which is a directed graph and is denoted as $G = (\mathbb{V}, \mathbb{E})$, where $\mathbb{V} = R(N, M_0)$ is the set of nodes and $\mathbb{E} = \{u \subseteq U \mid M[u\rangle M',\ M, M' \in R(N, M_0)\}$ is the set of edges. This implies that a node in G corresponds to a reachable state in $R(N, M_0)$ and an edge in G corresponds to an executable step in U.

III. Event Signal Reconstruction Problem

This paper only considers NCES without condition signals in the system and assumes there is no cascading of event signals. Let $\Gamma_E = (D, C, E, V, M_0)$ be the NCES under investigation with $C = \emptyset$, and $\Gamma_0 = (D, M_0)$ be the corresponding signal-free system of Γ_E. Given the reachability graph G_E of Γ_E and the reachability graph G_0 of Γ_0, the goal of event signal reconstruction is to determine a set of event signals E' and a multi-signal processing mode V' such that $\Gamma = (D, E', V', M_0)$ is equivalent to Γ_E, i.e., the reachability graph G of Γ is equal to G_E.

The current work assumes that both Γ_0 and Γ_E are fair, meaning that every transition in each system fires at least once. The reachable state sets of Γ_E, Γ_0, and Γ are denoted by $R(N_E, M_0)$, $R(N_0, M_0)$, and $R(N, M_0)$ respectively. According to the available information and the firing rules of NCES, the event signal reconstruction problem can be classified into four cases.

Problem 1: Given that the firing rule of the system is **S-mode** and the multi-event signal processing mode V' of the transitions are known, find E'.

Problem 2: Given that the firing rule of the system is **N-mode** and the multi-event signal processing mode V' of the transitions are known, find E'.

Problem 3: Given that the firing rule of the system is **S-mode**, determine both the event signal set E' and the multi-event signal processing V'.

Problem 4: Given that the firing rule of the system is **N-mode**, determine both the event signal set E' and the multi-event signal processing V'.

IV. Event Signal Reconstruction Methods

Based on the above assumptions, it is straightforward that a transition $t_i \in \mathbf{T}$ is a forced transition if it cannot fire alone in Γ_E, i.e., the following condition holds:

$$\begin{cases} \exists M \in R(N_0, M_0) : \{t_j\} \in \mathrm{Ena}(M), \\ \nexists M \in R(N_E, M_0) : \{t_j\} \in \mathrm{Ena}(M). \end{cases} \tag{3}$$

In an NCES, if a transition is not a forced transition, it is a spontaneous transition. Spontaneous transitions might be senders of event signals and forced transitions will be receivers of event signals.

A. Problem 1

Due to the firing mode **S-mode**, an executable step will include either only one spontaneous transition, or a spontaneous transition together with one of its forced transitions.

a) Case 1: $V(t) = \wedge$. If the multi-event signal processing mode of a forced transition t is “AND”, it can fire only when its unique trigger transition fires.

b) Case 2: $V(t) = \vee$. If the multi-event-signal processing mode of a forced transition t is “OR”, receiving any one of event signals from $^{\sim}t$ is sufficient for its firing after it gets enabled.

In both the two cases, an event signal $t_i \rightarrow t_j$ exists, i.e., $(t_i, t_j) \in E'$, if the following condition holds:

$$\exists M \in R(N_E, M_0) : \{t_i, t_j\} \in \mathrm{Ena}(M). \tag{4}$$

B. Problem 2

a) Case 1: $V(t) = \wedge$. Let $u \subseteq T_s$ be a set of spontaneous transitions and t_j be a forced transition. If the following condition holds:

$$\begin{cases} \exists M \in R(N_E, M_0) : u \cup \{t_j\} \in \mathrm{Ena}(M), \\ \nexists M \in R(N_E, M_0), \forall u' \subset u : u' \cup \{t_j\} \in \mathrm{Ena}(M), \end{cases} \tag{5}$$

we have that $\forall t_i \in u$, an event signal $t_i \to t_j$ exists, i.e., $(t_i, t_j) \in E'$. This condition implies that in Γ_E, t_j cannot fire alone, and t_j must always fire simultaneously with all transitions in u.

b) Case 2: $V(t) = \vee$. Let t_i be a spontaneous transition and t_j be a forced transition. An event signal $t_i \to t_j$ exists, i.e., $(t_i, t_j) \in E'$, if the following condition holds:

$$\exists M \in R(N_E, M_0), \{t_i, t_j\} \in \mathrm{Ena}(M). \tag{6}$$

C. Problem 3

a) AND-mode: If a forced transition t_j always fires with a special transition t_i in Γ_E, i.e., there exists no marking $M \in R(N_E, M_0)$ and no transition $t'_i \in T \setminus \{t_i\}$ such that $\{t'_i, t_j\} \in Ena(M)$, then we have $(t_i, t_j) \in E'$ and $V'(t_j) = \wedge$. Note that if $|{}^{\sim}t_j| \leq 1$, the default multi-event signal processing mode of t_j is set to "AND".

b) OR-mode: If a forced transition t_j could fire with different transitions in Γ_E, i.e., there exist markings $M, M' \in R(N_E, M_0)$ and spontaneous transitions t_i, t'_i such that $\{t_i, t_j\} \in Ena(M)$ and $\{t'_i, t\} \in Ena(M')$, then we have $(t_i, t_j) \in E'$, $(t'_i, t_j) \in E'$, and $V'(t) = \vee$.

D. Problem 4

a) AND-mode: Let $u \subseteq T_s$ be a set of spontaneous transitions and t_j be a forced transition. We have that $\forall t_i \in u$, an event signal $t_i \to t_j$ exists, i.e., $(t_i, t_j) \in E'$ and $V'(t_j) = \wedge$, if the following condition holds:

$$\begin{cases} \exists M \in R(N_E, M_0) : u \cup \{t_j\} \in \mathrm{Ena}(M), \\ \nexists M \in R(N_E, M_0), \forall u' \subset u : u' \cup \{t_j\} \in \mathrm{Ena}(M). \end{cases} \tag{7}$$

This condition implies that in Γ_E, t_j cannot fire alone, and t_j must always fire simultaneously with all transitions in u.

b) OR-mode: Let t_i and t'_i be two different spontaneous transitions and t_j be a forced transition. We have $(t_i, t_j) \in E'$, $(t'_i, t_j) \in E'$, and $V'(t_j) = \vee$, if the following condition holds:

$$\begin{cases} \exists M \in R(N_E, M_0), \{t_i, t_j\} \in \mathrm{Ena}(M), \\ \exists M \in R(N_E, M_0), \{t'_i, t_j\} \in \mathrm{Ena}(M). \end{cases} \tag{8}$$

In an NCES, any step that is firable under the **S-mode** firing rule is necessarily firable under the **N-mode** firing rule. That is, the reachability graph obtained under the **S-mode** firing rule is necessarily a subgraph of the reachability graph obtained under the **N-mode** firing rule. Therefore, Problem 1 is a special case of Problem 2 and Problem 3 is a special case of Problem 4.

V. ILLUSTRATING EXAMPLE

This chapter presents four simple examples to illustrate the event signal reconstruction method proposed in this paper. The net structure of the signal-free system remains identical across all four examples, as shown in Fig. 1, and the differences lie either in their initial markings or in their firing modes. The correctness of the results is confirmed by using the software tool SESA.

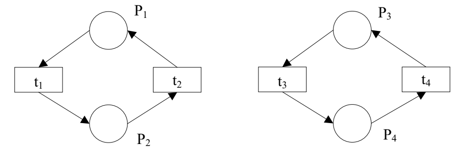

Fig. 1. Net structure of the no-signal system

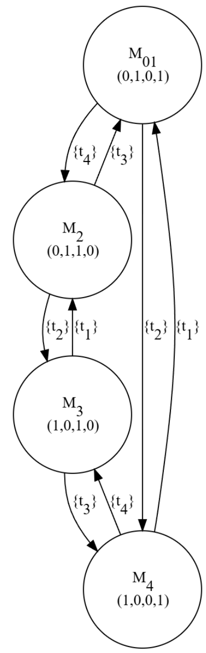

Fig. 2. Reachability graph of Γ_{01}

A. Example 1

The NCES without event signals is $\Gamma_{01} = (\mathcal{D}, M_{01})$ with $M_{01} = [0, 1, 0, 1]$. The reachability graph of Γ_{01} under **S-mode** is shown in Fig. 2. Suppose that the reachability graph of the actual system $\Gamma_{E1} = (\mathcal{D}, E_1, V_1, M_{01})$ has been obtained, as shown in Fig. 3. In this example, both E_1 and V_1 are unknown.

By comparison of the two reachability graphs, we can observe that t_2 and t_3 can fire concurrently in Γ_{E1}, which allows us to infer the existence of an event signal between t_2 and t_3. Noting that t_3 cannot fire alone in Γ_{E1}, we conclude that $E_1 = \{(t_2, t_3)\}$. In this case, since there is only one trigger transition of t_3, its multi-event signal processing mode is "AND", i.e., $V_1(t_3) = \wedge$. The system Γ_{E1} is shown in Fig. 4.

B. Example 2

In this example, the NCES without event signals is $\Gamma_{02} = (N, M_{02})$ with $M_{02} = [1, 1, 1, 0]$. The NCES with event signals is $\Gamma_{E2} = (N, E_2, V_2, M_{02})$. The reachability graphs of Γ_{02} and Γ_{E2} under **N-mode** are shown in Fig. 5 and Fig. 6, respectively.

It can be observed that t_3 cannot fire alone in Γ_{E2}. The only transition that can fire together with t_3 is t_2. Therefore,

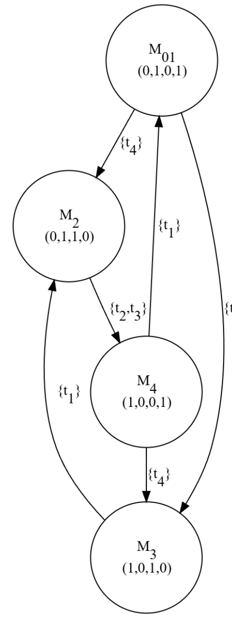

Fig. 3. Reachability graph of Γ_{E1}

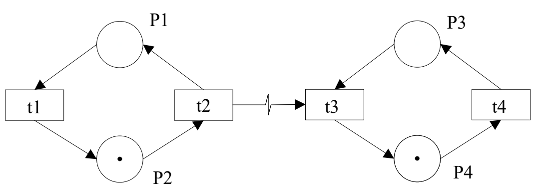

Fig. 4. Obtained Γ_{E1}

there exists an event signal from t_2 to t_3. We conclude that $E_2 = \{(t_2, t_3)\}$ and $V_2(t_3) = \wedge$. The system Γ_{E2} is shown in Fig. 7.

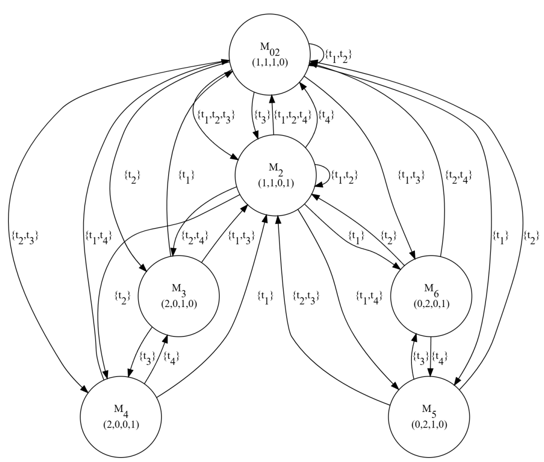

Fig. 5. Reachability graph of Γ_{02}

C. Example 3

Let $\Gamma_{E3} = (N, E_3, V_3, M_{03})$ denote the NCES with event signals. Its counterpart without event signal is $\Gamma_{03} = (N, M_{03})$ with $M_{03} = M_{02} = [1, 1, 1, 0]$. The reachability

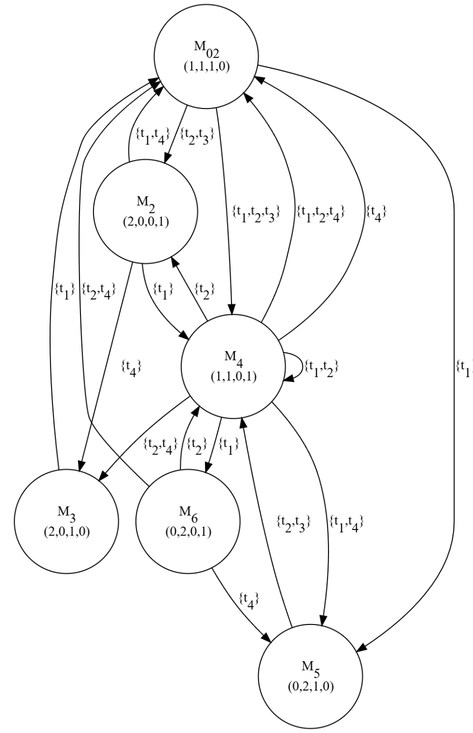

Fig. 6. Reachability graph of Γ_{E2}

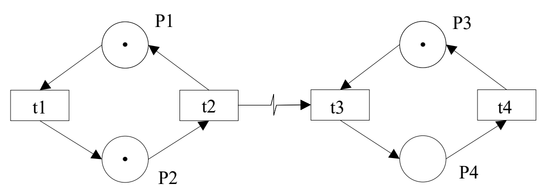

Fig. 7. Obtained Γ_{E2}

graph of Γ_{03} under **N-mode** is identical to that of Γ_{02}, as shown in Fig. 5. The reachability graph of Γ_{E3} under **N-mode** is shown in Fig. 8.

It can be observed that t_3 cannot fire alone in Γ_{E3}. Moreover, t_3 can occur only when both t_1 and t_2 fire simultaneously. Therefore, t_3 is identified as a **forced transition**, while t_1 and t_2 are identified as **trigger transitions** of t_3. Based on the observed firing behavior, it is inferred that the enabling of t_3 requires the simultaneous presence of event signals generated by both t_1 and t_2. Hence, we conclude that $E_3 = \{(t_1, t_3), (t_2, t_3)\}$ and $V_3(t_3) = \wedge$. The system Γ_{E3} is shown in Fig. 9.

D. Example 4

Let $\Gamma_{E4} = (N, E_4, V_4, M_{04})$ denote the NCES with event signals. Its counterpart without event signal is $\Gamma_{04} = (N, M_{04})$ that has the same net structure as that of Γ_{01} in Example 1 and is assigned the same initial state $M_{04} = M_{02} = [1, 1, 1, 0]$ as that of Γ_{02} in Example 2. The reachability graph of Γ_{04} under **N-mode** is identical to that of Γ_{02}, as shown in Fig. 5. The reachability graph of Γ_{E4} under **N-mode** is shown in Fig. 10.

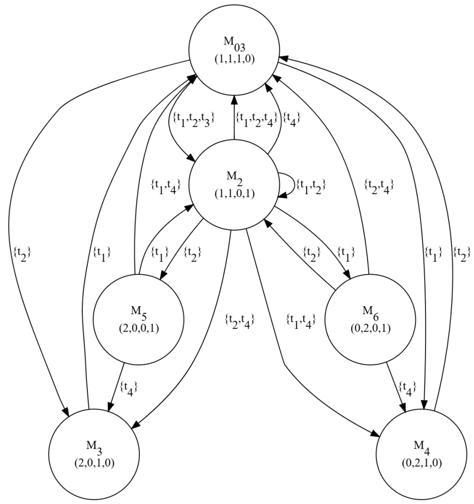

Fig. 8. Reachability graph of Γ_{E3}

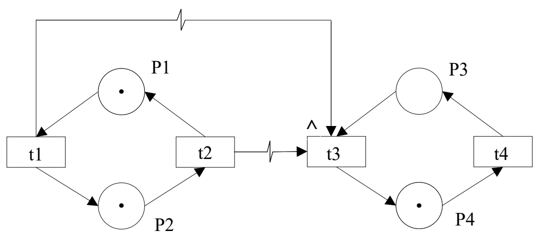

Fig. 9. Obtained Γ_{E3}

It can be observed that t_3 cannot fire alone in Γ_{E4}. Moreover, t_3 can be triggered by t_1, t_2, or the combination of both. Therefore, t_3 is identified as a **forced transition**, while t_1 and t_2 are identified as **trigger transitions** of t_3. Consequently, Hence, we conclude that $E_4 = \{(t_1, t_3), (t_2, t_3)\}$ and $V_4(t_3) = \vee$. The system Γ_{E4} is shown in Fig. 11.

VI. Conclusions

In this paper, a first methodological attempt toward event signal reconstruction in Net Condition/Event Systems (NCES) is presented by comparing reachability graphs with and without event signals. The proposed framework provides deterministic conditions for inferring event signal existence and distinguishing AND/OR processing modes at the receiver side, under **S-mode** and **N-mode** firing rules. However, this work is merely an initial exploration of the event signal reconstruction problem. The current method is subject to several strong assumptions that significantly limit its applicability. Specifically, it assumes the absence of condition signals, no cascading of event signals, fairness of transitions, and complete knowledge of the reachability graphs. As a result, the proposed approach is applicable only to a narrow class of NCES.

In future work, more general event signal reconstruction methods that relax these constraints are targeted. Key directions include: (1) incorporating condition signals into the reconstruction process, (2) supporting event signal cascading, (3) handling partial or sampled state information, and (4) mitigating the state explosion problem using symbolic or compositional techniques.

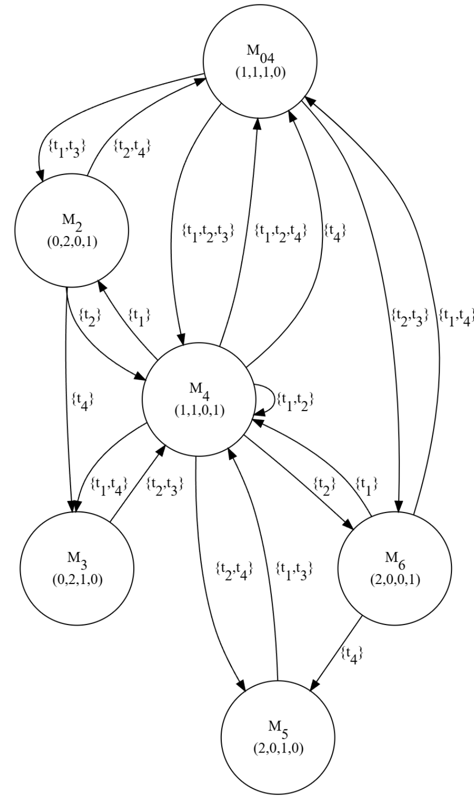

Fig. 10. Reachability graph of Γ_{E4}

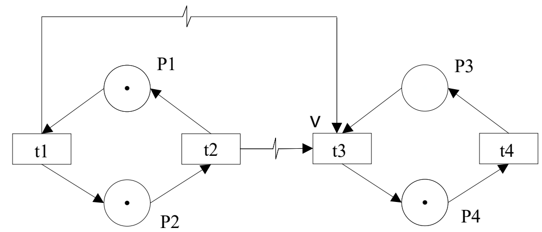

Fig. 11. Obtained Γ_{E4}

References

[1] R. S. Sreenivas and B. H. Krogh, "Petri net based models for condition/event systems," in *Proceedings of the 1991 American Control Conference*. IEEE, 1991, pp. 2899–2904.

[2] M. Rausch and H. M. Hanisch, "Net condition/event systems with multiple condition outputs," in *Proceedings of the 1995 INRIA/IEEE Symposium on Emerging Technologies and Factory Automation (ETFA'95)*, vol. 1. IEEE, 1995, pp. 592–600.

[3] G. Zhu, L. Feng, Z. Li *et al.*, "An efficient fault diagnosis approach based on integer linear programming for labeled Petri nets," *IEEE Transactions on Automatic Control*, vol. 66, no. 5, pp. 2393–2398, 2020.

[4] A. Gavric, D. Bork, and H. A. Proper, "Petri net of thoughts: A structure-enhanced prompting approach for process-aware artificial

intelligence," in *EMISA 2025*. Gesellschaft für Informatik, 2025, pp. 105–110.

[5] A. Khodadadi, A. Zare, M. Jungmann *et al.*, "A tutorial on data-driven Petri net model extraction and simulation for digital twins in smart manufacturing," in *Proceedings of the 2025 Winter Simulation Conference (WSC)*. IEEE, 2025, pp. 1212–1226.

[6] M. Montali, "Automated reasoning for data-aware Petri nets," in *Proceedings of the International Conference on Applications and Theory of Petri Nets and Concurrency*. Cham: Springer, 2025, pp. 1–17.

[7] H. Chen and H. M. Hanisch, "Analysis of hybrid systems based on hybrid net condition/event system model," *Discrete Event Dynamic Systems*, vol. 11, no. 1–2, pp. 163–185, 2001.

[8] A. Lüder, U. Anderssohn, and H. M. Hanisch, "Logic controller synthesis for net condition/event systems with forbidden paths," in *Proceedings of the European Control Conference (ECC)*. IEEE, 1997, pp. 2045–2050.

[9] T. Both, H. M. Hanisch, and J. Jorn, "Fault treatment with net condition/event systems: a first approach," in *Proceedings of the 8th International Conference on Emerging Technologies and Factory Automation (ETFA 2001)*, vol. 2. IEEE, 2001, pp. 429–432.

[10] A. Lobov, J. L. M. Lastra, R. Tuokko *et al.*, "Methodology for modeling visual flowchart control programs using net condition/event systems formalism in distributed environments," in *Proceedings of the IEEE Conference on Emerging Technologies and Factory Automation (ETFA 2003)*, vol. 2. IEEE, 2003, pp. 329–336.

[11] M. Xavier, S. Patil, V. Dubinin, and V. Vyatkin, "Formal modelling, analysis, and synthesis of modular industrial systems inspired by net condition/event systems," in *Application and Theory of Petri Nets and Concurrency*, L. Gomes and R. Lorenz, Eds. Cham: Springer Nature Switzerland, 2023, pp. 16–33.

[12] T. Murata, "Petri nets: Properties, analysis and applications," *Proceedings of the IEEE*, vol. 77, no. 4, pp. 541–580, 1989.

[13] W. Reisig, *Understanding Petri Nets*. Springer, 2013.

[14] J. Zhang, H. Li, G. Frey *et al.*, "Reconfiguration control of dynamic reconfigurable discrete event systems based on NCESs," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 3, pp. 857–868, 2019.

[15] P. Starke and S. Roch, "Analysing signal-net systems," Humboldt-Universität zu Berlin, Tech. Rep., 2002.