Basis Marking Representation of Petri Net Reachability Spaces and Its Application to the Reachability Problem

Ziyue Ma, Yin Tong, Zhiwu Li, and Alessandro Giua

June, 2017

Abstract

In this paper a compact representation of the reachability graph of a Petri net is proposed. The transition set of a Petri net is partitioned into the subsets of explicit and implicit transitions, in such a way that the subnet induced by implicit transitions does not contain directed cycles. The firing of implicit transitions can be abstracted so that the reachability set of the net can be completely characterized by a subset of reachable markings called *basis makings*. We show that to determine a max-cardinality-T_I basis partition is an NP-hard problem, but a max-set-T_I basis partition can be determined in polynomial time. The generalized version of the marking reachability problem in a Petri net can be solved by a practically efficient algorithm based on the basis reachability graph. Finally this approach is further extended to unbounded nets.

Published as:

[Z.Y. Ma, Y. Tong, Z.W. Li, A. Giua, "Basis Marking Representation of Petri Net Reachability Spaces and Its Application to the Reachability Problem," *IEEE Transactions on Automatic Control*, Vol. 62, No. 3, pp. 1078-1093, **2017**.] **DOI:** 10.1109/TAC.2016.2574120.

Ziyue Ma and Yin Tong are with School of Electro-Mechanical Engineering, Xidian University, Xi'an 710071, China, and also with Dipartimento di Ingegneria Elettrica ed Elettronica, Università degli Studi di Cagliari, Cagliari, Italy (email: maziyue@gmail.com, tongyin89@gmail.com).

Zhiwu Li is with Institute of Systems Engineering, Macau University of Science and Technology, Taipa, Macau (email: zhwli@xidian.edu.cn, systemscontrol@gmail.com).

Alessandro Giua is with Aix Marseille Université, CNRS, ENSAM, Université de Toulon, LSIS UMR 7296, Marseille 13397, France (email: alessandro.giua@lsis.org) and DIEE, University of Cagliari, Cagliari 09124, Italy (email: giua@diee.unica.it).

This work was supported in part by the National Natural Science Foundation of China under Grant Nos. 61374068, 61472295, the Recruitment Program of Global Experts, and the Science Technology Development Fund, MSAR, under Grant No. 078/2015/A3.

1 Introduction

Petri nets have been proposed as a fundamental model for *Discrete Event Systems* in a wide variety of applications and have been an asset to reduce the computational complexity involved in solving control problems. Among the methods that have been developed for the analysis of a Petri net, those based on reachability analysis are of particular importance. Typically they require solving the *marking reachability problem*, i.e., determining if a given marking is reachable from the initial one. It plays an important role in Petri net theory since many properties like liveness and deadlock-freeness require the reachability analysis of a system, and many other problems like supervisor design [1, 2], deadlock avoidance [3, 4], and controllability analysis [5, 6, 7] are equivalent or can be reduced to the marking reachability problem.

Although the marking reachability problem is decidable [8], it has been proved to be EXPSPACE-hard for arbitrary Petri nets [9]. In some restrictive subclasses of Petri nets, such as acyclic nets, the marking reachability problem can be determined by solving an *integer linear programming problem* (ILPP), and in state machines and marked graphs it can be solved in polynomial time [10]. As a result, people have turned to seeking non-polynomial but practically efficient methods to reduce the computational load.

The *binary decision diagrams* (BDD) method by Pastor et al. [11] encodes markings by a series of 0-1 boolean variables, and hence some property-checking problems can be solved by the *symbolic traverse* technique. However, it is not straightforward to reconstruct the behavior of the net, since the data structure of the BDD is based on P-invariants.

The *stubborn set* method by Valmari [12] works for Petri nets with identical concurrent structures, e.g., several identical workflows are triggered simultaneously and run in parallel such that markings can be abstracted by the submarkings of one workflow instead of many. However, the requirement that all workflows have identical structure is a very restrictive assumption that limits the applicability of this approach.

In [13, 14] a subnet can be reduced into a single transition while preserving properties such as the boundedness. In [15] by creating a "dependency graph" some substructures can be removed without changing the reachability property. However, these approaches also highly rely on some particular substructures of the net.

There are other methods including modular analysis [16] and multi-agent systems [17] that have shown good performance in particular cases. Moreover, some alternative methods have been developed to circumvent the marking reachability issue. For instance, in S^3PR nets *siphon analysis* can be used to determine if deadlock markings are reachable [18, 19, 20].

Recently a state compression approach [21, 22] proposed by Cabasino *et al.* has been used for state estimation and fault diagnosis in Petri nets with *observable* and *unobservable* transitions. The advantages of this technique is that only part of the reachability space, the namely *basis markings*, is enumerated; all

other markings reachable from them by firing only unobservable transitions can be characterized by a linear system. This method can be effectively used to solve the related controllability [6] and opacity problems [23, 24].

In this paper, we show that this approach can be generalized to provide a compact representation of the reachability set of a given Petri net. In particular, we present a practically efficient algorithm to solve the marking reachability problem, where only a subset of the reachable markings is enumerated. The proposed approach has wide applicability since it works for Petri nets with quite general structures. The main features of this approach are summarized as follows.

First, the transition set is partitioned into two disjoint sets called the *explicit transition set* and the *implicit transition set*, where the subnet induced by implicit transitions is acyclic, i.e., there exists no cycle of implicit transitions. A method to compute *minimal explanation vectors* for explicit transitions is proposed, and then a subset of reachable markings called *basis markings* is computed and the corresponding *basis reachability graph* (BRG) is constructed. We show that the BRG can be used to represent all reachable markings. Moreover, it also preserves the information of the firing sequences of the net, since the firing of all explicit transitions is directly recorded while the firing of implicit transitions can be easily reconstructed from the state equation.

Second, we prove that the number of basis markings decreases monotonically when the implicit transition set grows. Hence it is preferable to determine an implicit transition set as large as possible. However, we also prove that to find an implicit transition set with *maximum cardinality* is unfortunately *NP-hard* with respect to the scale of the plant net. Hence we propose an algorithm to find a *maximal implicit transition set* in polynomial time.

Third, we consider the problem to determine a firing sequence with a minimal cost to reach a target marking set defined by an OR-AND GMEC (its definition is given in Section V) in a given marked Petri net $\langle N, M_0 \rangle$. We show that if the net has a finite number of basis markings with respect to some basis partition, then the problem to determine a firing sequence with the minimal cost can be solved by first transforming the BRG into a *basis cost graph* and then solving the shortest path problem in it.

At the end of this paper we study the finiteness of the basis marking graph. If a net does not contain source transitions, then its BRG is finite if and only if the net is bounded. If the net contains source transitions, we introduce the *complete minimal explanation set* and propose a construct called the *potential explanation net*, which allows us to determine the finiteness of the BRG. This approach extends the applicability of BRG to unbounded systems by introducing a stopping criterion during its construction.

This paper is organized in seven sections. The basics of Petri nets are recalled in Section II. Section III gives the notions of minimal explanations, basis markings, and the construction of the basis reachability

graph. In Section IV the problem of finding a max-cardinality-T_I basis partition is proved to be NP-hard, and an algorithm to compute a max-set-T_I basis partition is proposed. In Section V an algorithm is developed to solve the generalized marking reachability problem. In Section VI the BRG approach is extended to unbounded systems. Conclusions are reached in Section VII.

2 Preliminaries

A Petri net is a four-tuple $N = (P, T, Pre, Post)$, where P is a set of m *places* represented by circles; T is a set of n *transitions* represented by bars; $Pre : P \times T \to \mathbb{N}$ and $Post : P \times T \to \mathbb{N}$ are the *pre-* and *post-incidence functions*, respectively, that specify the arcs in the net and are represented as matrices in $\mathbb{N}^{m \times n}$ (here $\mathbb{N} = \{0, 1, 2, \ldots\}$). The *incidence matrix* of a net is defined by $C = Post - Pre \in \mathbb{Z}^{m \times n}$ (here $\mathbb{Z} = \{0, \pm 1, \pm 2, \ldots\}$).

For a transition $t \in T$ we define the *set of its input places* as ${}^\bullet t = \{p \in P \mid Pre(p, t) > 0\}$ and the *set of its output places* as $t^\bullet = \{p \in P \mid Post(p, t) > 0\}$. The notions for ${}^\bullet p$ and $p^\bullet$ are analogously defined.

A *marking* is a vector $M : P \to \mathbb{N}$ that assigns to each place of a Petri net a non-negative integer number of tokens, represented by black dots and can also be represented as an m-component vector. We denote by $M(p)$ the marking of place p. A *marked net* $\langle N, M_0 \rangle$ is a net N with an initial marking M_0. We denote by $R(N, M_0)$ the set of all markings reachable from the initial one. We also use $x_1 p_1 + \cdots + x_n p_n$ to denote the marking $[x_1, \ldots, x_n]^T$ for simplicity.

A transition t is *enabled* at M if $M \geq Pre(\cdot, t)$ and may fire reaching a new marking $M' = M + C(\cdot, t)$. We write $M[\sigma\rangle$ to denote that the sequence of transitions σ is enabled at M, and we write $M[\sigma\rangle M'$ to denote that the firing of σ yields M'. The vector $\mathbf{y}_\sigma$ is the Parikh vector of $\sigma \in T^*$, i.e., $y_\sigma(t) = k$ if transition t appears k times in σ.

Given a net $N = (P, T, Pre, Post)$ we say that $\hat{N} = (\hat{P}, \hat{T}, \hat{Pre}, \hat{Post})$ is a subnet of N if $\hat{P} \subseteq P$, $\hat{T} \subseteq T$ and $\hat{Pre}$ (resp., $\hat{Post}$) is the restriction of Pre (resp., $Post$) to $\hat{P} \times \hat{T}$. In particular, a T_I-induced subnet is the net $(P, T_I, \hat{Pre}, \hat{Post})$ where $T_I \subset T$, and we use C_I to denote the incidence matrix C restricted to $P \times T_I$.

A Petri net $\langle N, M_0 \rangle$ is said to be bounded if there exists an integer $K \in \mathbb{N}$ such that $\forall M \in R(N, M_0)$ and $\forall p \in P$, $M(p) \leq K$ holds. A net N is structurally bounded if for any $M_0 \in \mathbb{N}^m$, the marked net $\langle N, M_0 \rangle$ is bounded.

A *graph* is denoted as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ and $\mathcal{E}$ are the set of *vertices* and *edges*, respectively.

3 Basis Markings and Basis Reachability Graphs

In the work of Cabasino *et al.* [21, 22], a compact way to represent the reachability set of a Petri net is proposed to solve diagnosis problems. In their approach the transition set T is partitioned into the *observable* transition set T_o and the *unobservable* transition set T_{uo}, the latter of which also includes fault transitions. In [21, 22] only a subset of the reachable markings, called *basis markings*, is computed, and a non-deterministic finite state automaton called *basis reachability graph* (BRG) is constructed, in which each state corresponds to a basis marking. All non-basis markings and the unobservable firing sequences can be characterized by the solution of a set of linear equalities depending on the basis markings.

The original BRG approach has only been applied to the issues related to the observability, e.g., diagnosability problems and related opacity problems [21, 22, 23, 24, 25, 26]. In this paper we show that a similar approach based on the construction of a BRG provides a practically efficient approach to solve the general reachability problem. Although the term *basis reachability graph* was used in [21] by one of us, it would be appropriate to rename the structure proposed in [21] as "Diagnosis BRG" since it contains additional information tailored for the analysis of diagnosis and diagnosability properties. On the contrary, the BRG we use in this paper is a more fundamental structure that only contains basis markings.

3.1 Basis Partitions and Minimal Explanations

Definition 1 *Given a Petri net $N = (P, T, Pre, Post)$, a pair $\pi = (T_E, T_I)$ is called a* basis partition *of T if (1) $T_I \subseteq T$, $T_E = T \setminus T_I$; and (2) the T_I-induced subnet is acyclic. In a basis partition (T_E, T_I), the sets T_E and T_I are called the* explicit transition set *and the* implicit transition set*, respectively.* □

In plain words, a basis partition is a partition of T into T_E and T_I such that the T_I-induced subnet is acyclic. Note that the terms "explicit" and "implicit" are not related to the physical meaning of the transitions. We denote $|T_E| = n_E$ and $|T_I| = n_I$.

Definition 2 *Given a Petri net $N = (P, T, Pre, Post)$, a basis partition $\pi = (T_E, T_I)$, a marking M, and a transition $t \in T_E$, we define*

$$\Sigma(M, t) = \{\sigma \in T_I^* \mid M[\sigma\rangle M', M' \geq Pre(\cdot, t)\}$$

the set of explanations *of t at M, and we define*

$$Y(M, t) = \{\boldsymbol{y}_\sigma \in \mathbb{N}^{n_I} \mid \sigma \in \Sigma(M, t)\}$$

the set of explanation vectors. □

The physical meaning of $\Sigma(M,t)$ is the following: from M if we want to enable the explicit transition t by firing only implicit transitions, then some sequence $\sigma \in \Sigma(M,t)$ must fire. The set $Y(M,t)$ is composed of the firing vectors associated to the firing sequences in $\Sigma(M,t)$.

Definition 3 *Given a Petri net $N = (P, T, Pre, Post)$, a basis partition $\pi = (T_E, T_I)$, a marking M, and a transition $t \in T_E$, we define*

$$\Sigma_{min}(M,t) = \{\sigma \in \Sigma(M,t) \mid \nexists \sigma' \in \Sigma(M,t) : \boldsymbol{y}_{\sigma'} \lneq \boldsymbol{y}_{\sigma}\}$$

the set of minimal explanations *of t at M, and we define*

$$Y_{min}(M,t) = \{\boldsymbol{y}_{\sigma} \in \mathbb{N}^{n_I} \mid \sigma \in \Sigma_{min}(M,t)\}$$

the corresponding set of minimal explanation vectors. □

In plain words, $\Sigma_{min}(M,t)$ is the set of sequences in $\Sigma(M,t)$ with minimal firing sequences and $Y_{min}(M,t)$ is the set of these minimal firing vectors. The two notions will be used to define basis markings. Typically $\Sigma_{min}(M,t)$ is not a singleton, since there are possibly multiple minimal sequences $\sigma \in T_I^*$ that can enable the explicit transition t, which implies that $Y_{min}(M,t)$ is neither a singleton in general. By [27], if the T_I-induced subnet is acyclic and backward-conflict-free (i.e., each place has at most one input transition), then $Y_{min}(M,t)$ is a singleton. If $\Sigma(M,t) = \Sigma_{min}(M,t) = \emptyset$ (which implies that $Y(M,t) = Y_{min}(M,t) = \emptyset$), then from M one cannot enable t by firing only implicit transitions.

The following algorithm can be used to compute $Y_{min}(M,t)$ for a given marking M and an explicit transition t [21, 28].

For economy of space, we do not present an example to illustrate Algorithm 1, but examples can be found in [21] and [28]. In plain words, Algorithm 1 iteratively searches and enumerates all possible firing vectors $\mathbf{y}_{\sigma} \in \mathbb{N}^{n_I}$ in a *breadth-first* way such that σ is an explanation of t, i.e., $M[\sigma\rangle M'[t\rangle$, since for any index i^* the following equality holds[1]:

$$A(i^*, \cdot) = M + C_I \cdot B(i^*, \cdot) - Pre(\cdot, t).$$

The condition $A(i^*, j^*) < 0$ and $\mathcal{X}^+ = \emptyset$ means that the number of tokens in place p_{j^*} is not sufficient and cannot be increased by firing any implicit transition, which implies that there does not exist any explanation whose firing vector $\mathbf{y} \geq B(i^*, \cdot)$ so that row i^* can be discarded.

[1] We use $A(x, \cdot)$ and $B(x, \cdot)$ to denote the x-th row of the matrix $\mathbf{A}$ and $\mathbf{B}$, respectively.

Algorithm 1 Calculation of $Y_{min}(M,t)$ [Algorithm 3.5 in [28]]

Input: A Petri net N, a basis partition $\pi = (T_E, T_I)$, a marking M, and $t \in T_E$
Output: $Y_{min}(M,t)$

1: Let $\Gamma = \begin{bmatrix} C_I^T & I_{n_I \times n_I} \\ \mathbf{A} & \mathbf{B} \end{bmatrix}$ where $\mathbf{A} = (M - Pre(\cdot,t))^T$, $\mathbf{B} = \mathbf{0}_{n_I}^T$;
2: **while** $\mathbf{A} \ngeq \mathbf{0}$ **do**
3: Choose an element $A(i^*, j^*) < 0$;
4: Let $\mathcal{X}^+ = \{i \mid C_I^T(i, j^*) > 0\}$;
5: **if** $\mathcal{X}^+ = \emptyset$, **then**
6: delete $[A(i^*,\cdot) \mid B(i^*,\cdot)]$ from $[\mathbf{A} \mid \mathbf{B}]$, goto Step 2;
7: **end if**
8: **for all** $i \in \mathcal{X}^+$, **do**
9: add to $[\mathbf{A} \mid \mathbf{B}]$ a new row $[A(i^*,\cdot) \mid B(i^*,\cdot) + \Gamma(i,\cdot)]$;
10: **end for**
11: Delete row $[A(i^*,\cdot) \mid B(i^*,\cdot)]$ from $[\mathbf{A} \mid \mathbf{B}]$;
12: **end while**
13: Let Y be the set of row vectors in $\mathbf{B}$;
14: Let $Y_{min}(M,t)$ be the set of minimal elements in Y. Output $Y_{min}(M,t)$.

In the next subsection we define the BRG which can be constructed by these methods if the net is bounded. Moreover, in Section VI we present a more general approach to compute Y_{min} which can be applied to construct the BRG of unbounded nets.

3.2 Basis Markings and Basis Reachability Graph

The definition of basis markings can be given in an iterative way as follows.

Definition 4 *Given a Petri net $N = (P, T, Pre, Post)$ with an initial marking M_0 and a basis partition $\pi = (T_E, T_I)$, its* basis marking set $\mathcal{M}(N, M_0, \pi)$ *is defined as follows:*

- $M_0 \in \mathcal{M}(N, M_0, \pi)$*;*
- *If $M \in \mathcal{M}(N, M_0, \pi)$, then $\forall t \in T_E$, $\forall \boldsymbol{y} \in Y_{min}(M,t)$,*

$$(M' = M + C_I \cdot \boldsymbol{y} + C(\cdot, t)) \Rightarrow (M' \in \mathcal{M}(N, M_0, \pi)).$$

A marking M in $\mathcal{M}(N, M_0, \pi)$ is called a basis marking *of $\langle N, M_0 \rangle$ with respect to $\pi = (T_E, T_I)$.* □

In other words, the set of basis markings contains the initial marking. All other markings in the set are reachable from another basis marking by firing a sequence $\sigma_I t$ where $t \in T_E$ is an explicit transition and the sequence $\sigma_I \in T_I^*$ is its minimal explanation.

For different basis partitions the corresponding basis markings are different. However, to simplify the notations in the sequel we denote $\mathcal{M}(N, M_0, \pi)$ as $\mathcal{M}$ without specifying its net and the basis partition, in case that there is no confusion. Moreover, from the definition of $\mathcal{M}$ it is trivial that $\mathcal{M} \subseteq R(N, M_0)$. Now we present an algorithm to construct the *Basis Reachability Graph* (BRG) of a given bounded Petri net and a basis partition.

Definition 5 *Given a bounded net $N = (P, T, Pre, Post)$ with an initial marking M_0 and a basis partition $\pi = (T_E, T_I)$, its* basis reachability graph *(BRG) is a non-deterministic finite state automaton $\mathcal{B}$ output by Algorithm 2. The BRG $\mathcal{B}$ is a quadruple $(\mathcal{M}, Tr, \Delta, M_0)$, where:*

- *the state set $\mathcal{M}$ is the set of* basis markings*;*
- *the event set Tr is the set of pairs $(t, \mathbf{y}) \in T_E \times \mathbb{N}^{n_I}$;*
- *the transition relation Δ is:*

$$\Delta = \{(M_1, (t, \mathbf{y}), M_2) \mid t \in T_E, \mathbf{y} \in Y_{min}(M_1, t), \\ M_2 = M_1 + C_I \cdot \mathbf{y} + C(\cdot, t)\}$$

- *the initial state is the initial marking M_0.*

□

Algorithm 2 Construction of a basis reachability graph

Input: A marked Petri net $\langle N, M_0 \rangle$ and a basis partition $\pi = (T_E, T_I)$
Output: A BRG $\mathcal{B} = (\mathcal{M}, Tr, \Delta, M_0)$

1: Let $\mathcal{M} = \emptyset$ and $\mathcal{M}_{new} = \{M_0\}$;
2: **while** $\mathcal{M}_{new} \neq \emptyset$ **do**
3: Select a state $M \in \mathcal{M}_{new}$;
4: **for all** $t \in T_E$, **do**
5: Compute $Y_{min}(M, t)$;
6: **for all** $\mathbf{y} \in Y_{min}(M, t)$, **do**
7: Let $\hat{M} = M + C_I \cdot \mathbf{y} + C(\cdot, t)$;
8: **if** $\nexists \hat{M} \in \mathcal{M} \cup \mathcal{M}_{new}$ **then**
9: Let $\mathcal{M}_{new} = \mathcal{M}_{new} \cup \{\hat{M}\}$;
10: **end if**
11: Let $\Delta = \Delta \cup (M, (t, \mathbf{y}), \hat{M})$;
12: **end for**
13: **end for**
14: Let $\mathcal{M} = \mathcal{M} \cup \{M\}$;
15: Let $\mathcal{M}_{new} = \mathcal{M}_{new} \setminus \{M\}$;
16: **end while**
17: Output $\mathcal{B} = (\mathcal{M}, Tr, \Delta, M_0)$.

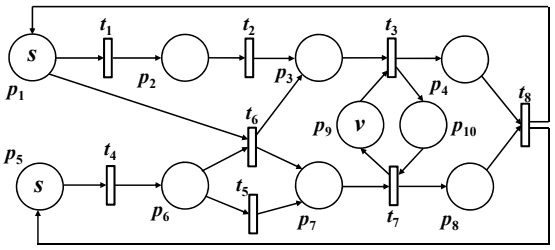

Figure 1: A parameterized Petri net plant.

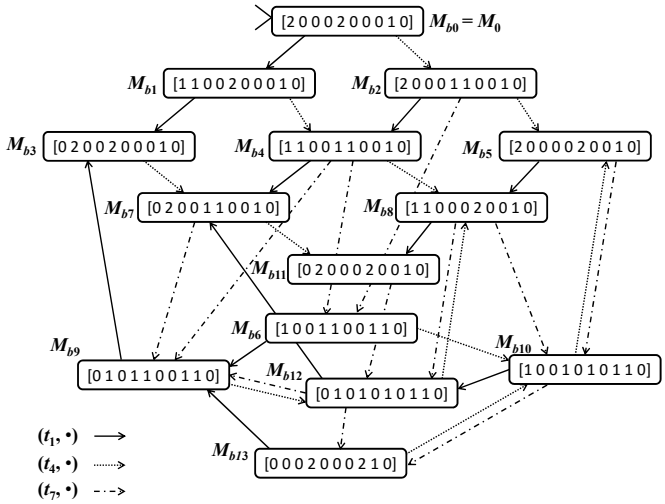

Figure 2: The BRG $\mathcal{B}$ of the net in Figure 1 for $T_E = \{t_1, t_4, t_7\}$ and $T_I = \{t_2, t_3, t_5, t_6, t_8\}$.

Algorithm 2 works in the following way. Initially, the initial marking M_0 is added to $\mathcal{M}_{new}$ to denote that it is not checked yet. In the iteration cycle, if $\mathcal{M}_{new}$ is not empty, then a marking M in $\mathcal{M}_{new}$ is selected. For each explicit transition $t \in T_E$ the set $Y_{min}(M, t)$ is calculated, and for each $\mathbf{y} \in Y_{min}(M, t)$, a new marking $\hat{M} = M + C_I \cdot \mathbf{y} + C(\cdot, t)$ is added to $\mathcal{M}_{new}$ if $\hat{M}$ is neither in $\mathcal{M}$ nor in $\mathcal{M}_{new}$. Then the transition relation $\Delta(M, (t, \mathbf{y}), M')$ is defined. Finally M is moved from $\mathcal{M}_{new}$ to $\mathcal{M}$ to denote that M has been checked. This procedure runs iteratively until there is no unchecked marking in $\mathcal{M}_{new}$. The BRG of a net $\langle N, M_0 \rangle$ with basis partition π is denoted by $\mathcal{B}(N, M_0, \pi)$, and when no confusion can arises we simply denote it as $\mathcal{B}$.

By $\mathcal{M} \subseteq R(N, M_0)$, if $\langle N, M_0 \rangle$ is bounded, then Algorithm 2 terminates in a finite time. For an unbounded net, the basis marking set $\mathcal{M}$ can be either finite or infinite, as discussed in Section VI.

Example 1 *Consider the Petri net in Figure 1 with a parameterized initial marking M_0, i.e., $M_0(p_1) = M_0(p_5) = s$, $M_0(p_9) = v$, and $M_0(p) = 0$ for other places. This net models a system that contains two workflows ($p_1 t_1 p_2 t_2 p_3 t_3 p_4$ and $p_5 t_4 p_6 t_5 p_7 t_7 p_8$) that produce two types of parts to be assembled (transition t_8). Both workflows have capacity s. A special machine, represented by t_6, can process raw parts from the buffers p_1 and p_6 simultaneously and output the machined parts into p_3 and p_7, respectively. Two monitors*

represented by places p_9 and p_{10} enforce a control policy that the numbers of parts in buffers p_4 and p_8 should not differ too much, i.e., $M(p_4) - M(p_8) \leq v$ and $M(p_8) - M(p_0) \leq 0$.

Now consider this net with the instance $s = 2$ and $v = 1$. The reachability graph of the net instance has 67 reachable markings, which is too complex to be graphically presented here. However, under a basis partition $\pi = (T_E, T_I)$ with $T_E = \{t_1, t_4, t_7\}$ and $T_I = \{t_2, t_3, t_5, t_6, t_8\}$, the resulting BRG has only 14 basis markings, as shown in Figure 2. □

3.3 Properties of BRG

In the following we show that the BRG preserves the reachability and the information about the firing sequences of the net.

Definition 6 *Given a net $N = (P, T, Pre, Post)$, a basis partition $\pi = (T_E, T_I)$, and a basis marking $M_b \in \mathcal{M}$, we define $R_I(M_b)$ the* implicit reach *of M_b as:*

$$R_I(M_b) = \{M \in \mathbb{N}^m \mid \exists \sigma \in T_I^*, M = M_b + C \cdot \boldsymbol{y}_\sigma\}.$$

□

The implicit reach of a basis marking M_b consists of all markings that can be reached from M_b by firing only implicit transitions.

Theorem 1 *Given a Petri net $\langle N, M_0 \rangle$, a basis partition $\pi = (T_E, T_I)$, and a marking $M \in \mathbb{N}^m$, it holds:*

$$\begin{aligned} &(\exists \sigma \in T^*, \sigma_{\uparrow T_E} = \sigma_E) M_0[\sigma\rangle M \Leftrightarrow \\ &(\exists M_b)(M_0, \sigma_E, M_b) \in \Delta^*, M \in R_I(M_b) \end{aligned} \tag{1}$$

where $w_{\uparrow X}$ denotes the natural projection of a word w onto the alphabet X, and $(M_0, \sigma_E, M_b) \in \Delta^*$ denotes that in the BRG there exists a sequence $(t_1, \boldsymbol{y}_1), \ldots, (t_k, \boldsymbol{y}_k)$ where $t_1 \cdots t_k = \sigma_E$ and M_b is reachable from M_0 by firing the event sequence $(t_1, \boldsymbol{y}_1) \cdots (t_k, \boldsymbol{y}_k)$.

Proof: This result follows from Theorem 3.8 in [21] by considering observable transitions as T_E and unobservable transitions as T_I. ■

Corollary 1 *Given a Petri net $\langle N, M_0 \rangle$ and its BRG $\mathcal{B}$, we have:*

$$R(N, M_0) = \bigcup_{M_b \in \mathcal{M}} R_I(M_b).$$

Table 1: The performance of Algorithm 2

Run	s	v	$\|R(N, M_0)\|$	[1]Time [s]	$\|\mathcal{M}\|$	Time [s]	$\|\mathcal{M}\|/\|R(N, M_0)\|$
1	2	1	67	< 1	14	<0.1	20.9%
2	4	3	783	7	55	0.4	7.0%
3	6	5	4298	273	140	1.1	3.3%
4	8	7	16026	3816	285	2.5	1.7%
5	10	9	46981	15667	506	5.2	1.1%
6	15	14	-	o.t.	1496	25	-
7	20	19	-	o.t.	3311	87	-
8	30	29	-	o.t.	10416	674	-

*In this benchmark $T_E = \{t_1, t_4, t_7\}$ and $T_I = \{t_2, t_3, t_5, t_6, t_8\}$.
[1]We denote *overtime* (o.t.) if the Matlab program does not terminate within 8 hours (28,800 seconds).

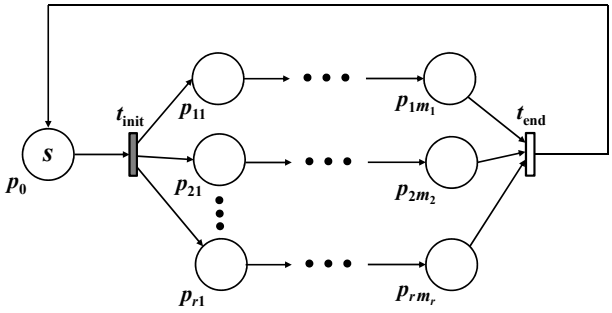

Figure 3: The Petri net for Example 2.

By Theorem 1, for an arbitrary net $\langle N, M_0\rangle$ and an arbitrary basis partition $\pi = (T_E, T_I)$, a marking M is reachable from M_0 by firing $\sigma \in T^*$ if and only if there exists some basis marking M_b such that $(M_0, \sigma_{\uparrow E}, M_b) \in \Delta^*$ and M belongs to the implicit reach of M_b.

3.4 Computational Efficiency of BRG

Let us first consider the complexity of a BRG, i.e., the number of basis markings in it. Consider a basis partition (T_E, T_I) in which $T_E = T$ and $T_I = \emptyset$ (which is the only valid basis partition in the case that all transitions have self-loops). Then the set of basis markings is identical to the reachability set, i.e., $\mathcal{M} = R(N, M_0)$. Although in the worst case the BRG has the same complexity as the reachability graph, we note that in practice we can usually find a basis partition (T_E, T_I) in which $T_I \neq \emptyset$, under which the BRG is much more compact than the reachability graph, i.e., $|\mathcal{M}| << |R(N, M_0)|$ (see Example 3). Moreover, for nets in which all cycles pass through a few key transitions, by assigning to T_E only these transitions, the size of the BRG can be significantly smaller than that of the reachability graph, since most transitions are taken to be implicit and their firings are omitted in the BRG. In this subsection we identify an interesting class of networks of this type.

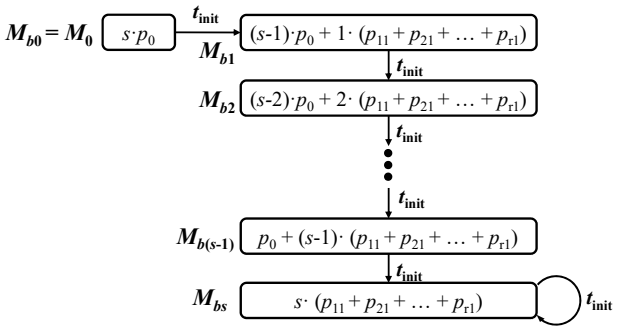

Figure 4: The BRG $\mathcal{B}$ of the net in Figure 3 for $T_E = \{t_{init}\}$, $T_I = T \setminus T_E$, and each workflow is a state machine that contains m places.

Definition 7 *[29] A Petri net $N = (P, T, Pre, Post)$ is a* workflow net *if: (1) N has an input place p_{in} and an output place p_{out} satisfying ${}^\bullet p_{in} = p_{out}^\bullet = \emptyset$, and (2) if we add a transition t_{reset} to N with $Pre(p_{in}, t_{reset}) = Post(p_{out}, t_{reset}) = 1$, then the resulting net is strongly connected.* □

Definition 8 *A Petri net $N = (P, T, Pre, Post)$ is a* parallel acyclic workflow net *if: (1) N has an idle place p_0, an initial transition t_{init}, an end transition t_{end}, and r acyclic workflows. Each workflow i has its input place p_{i1} and its output place p_{im_i}; (2) $Pre(p_0, t_{init}) = Post(p_0, p_{end}) = 1$, for all $i \in \{1, \ldots, r\}$, $Post(p_{i1}, t_{init}) = Pre(p_{im_i}, t_{end}) = 1$.* □

The general structure of a parallel acyclic workflow net is shown in Figure 3. It contains r acyclic workflows, i.e., each subnet between p_{i1} (which is the input place) and p_{im_i} (which is the output place) is an acyclic workflow net. Place p_0 contains s initial tokens, where s represents the capacity of the system. The firing of transition t_{init} triggers the start of a new task on all workflows, while the firing of transition t_{end} denotes that all workflows have finished a task.

Example 2 *Consider the reachability space of a parallel acyclic workflow net. Since there is no structural restriction on workflows except the acyclicity, none of Petri net structural reduction methods including those in [12, 13, 14, 15] is applicable. In such a case, $|R(N, M_0)|$ is exponential with respect to the number of workflows and is (at least) polynomial with respect to s and m. For instance, let us consider the case that each workflow i is a state machine composed by a sequence containing alternatively m places $p_{i1}, \ldots, p_{im}$ and $m - 1$ transitions $t_{i1}, \ldots, t_{i(m-1)}$. It is not difficult to compute the cardinality of the reachability set, which is*

$$|R(N, M_0)| = \sum_{i=0}^{s} \left[\binom{m+i-1}{i} \right]^r$$

that grows exponentially with r and polynomially with s and m.[2]

[2]Here we use $\binom{x}{y}$ to denote the combination number of "x choose y", i.e., $\binom{x}{y} = \frac{x!}{(x-y)!y!}$, where $x, y \in \mathbb{N}, x \geq y$.

On the contrary, by choosing the basis partition (T_E, T_I) such that $T_E = \{t_{init}\}$ and $T_I = T \backslash \{t_{init}\}$, the corresponding BRG is shown in Figure 4. Here one can see that the number of basis markings is $|\mathcal{M}| = s+1$, and this number does not depend on r (the number of workflows) and on the structure of the workflows. It depends but only linearly from the parameter s. We also note that if the workflows are not acyclic, the BRG approach can also be applied by extending the explicit transition set T_E to remove cycles from the T_I-induced subnet. □

Since Theorem 1 and Corollary 1 ensure that the reachability set of a Petri net can be exactly characterized by only basis markings, without any loss of information, the basis marking approach circumvents the need of constructing the entire reachability graph of a net and brings significant advantages from the point of view of the computational effort. In particular, the *generalized marking reachability problem* in Section V can be solved by analyzing the basis reachability graph. To determine whether a given marking M belongs to $R_I(M_b)$, an integer linear programming problem (ILPP) needs to be solved. Simulation results by Matlab show that the total computational load of solving $|\mathcal{M}|$ ILPPs is negligible (approximately 5%) in most cases comparing with that of constructing the BRGs. This is illustrated by the following example.

Example 3 (Ex. 1 Continued) *Let us consider again the Petri net in Figure 1 and a basis partition (T_E, T_I) with $T_I = \{t_2, t_3, t_5, t_6, t_8\}$ and $T_E = \{t_1, t_4, t_7\}$. For different parameters s and v the size of the reachability graph and the size of the BRG (by Algorithm 2) are listed in Table 1 as well as the time to construct them. The simulation is done on a workstation with Core-i5 1.7GHz/2.2GHz CPU using the standard Matlab toolboxes. We denote by* overtime *(o.t.) the fact that the Matlab program does not terminate within 8 hours (28,800s).*

From Table 1 one can see that with the increase of s and v, the size of the reachability graph ($|R(N, M_0)|$) grows much faster than that of the BRG ($|\mathcal{M}|$). As a result, the time to construct the BRG is significantly shorter than that to construct the reachability graph. In this example when the number of reachable markings exceeds $40,000$ (Run 5), the computation of the reachability graph cannot be done within 8 hours. However, Algorithm 2 can handle more than 10^8 reachable markings[3] (Run 8) before running overtime. □

At the end of this section we point out that although the strategy to construct a BRG based on solving $Y_{min}(M, t)$ for each basis marking is practically efficient, this strategy is not directly applicable to unbounded nets since Algorithm 2 may not terminate. Such cases are studied in Section VI, where the *complete minimal explanation set* is introduced, which also provides a different strategy to compute minimal explanations. Moreover, this strategy can be easily embedded in Algorithm 2.

[3]The number of reachable markings can be approximated by the possible token distributions in three independent P-invariants ($p_1p_2p_3p_4$, $p_5p_6p_7p_8$, and p_9p_{10}), i.e., $|R(N, M_0)| \approx [(s-1)(s-2)(s-3)/6]^2 \cdot (v+1)$.

Table 2: The basis partitions in Example 4

π	T_{I_i}	T_{E_i}
π_0	$\emptyset$	$t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8$
π_1	t_2, t_5, t_6, t_8	t_1, t_3, t_4, t_7
π_2	t_2, t_3, t_5, t_6, t_8	t_1, t_4, t_7
π_3	$t_2, t_3, t_4, t_5, t_6, t_8$	t_1, t_7
π_4	$t_1, t_2, t_4, t_5, t_6, t_7$	t_3, t_8

4 Determining Max-cardinality-T_I and Max-Set-T_I Basis Partitions

Given a Petri net $\langle N, M_0\rangle$, usually there exist multiple basis partitions, which implies that there exist more than one BRG which can be used as a compact representation of its reachability set.

The major concern is the cardinality of $\mathcal{M}$, i.e., the number of basis markings that correspond to a basis partition. However, minimizing the cardinality of $\mathcal{M}$ is a difficult task, since in general one cannot directly relate the number of basis markings to a given basis partition. On the other hand, the following proposition shows that the number of basis markings monotonically decreases as the implicit transition set T_I grows.

Proposition 1 *Given two basis partitions $\pi_1 = (T_{E_1}, T_{I_1})$ and $\pi_2 = (T_{E_2}, T_{I_2})$ on a net $\langle N, M_0\rangle$ where $N = (P, T, Pre, Post)$, it holds $(T_{I_1} \subseteq T_{I_2}) \Rightarrow (|\mathcal{M}_1| \geq |\mathcal{M}_2|)$, where $\mathcal{M}_1$ and $\mathcal{M}_2$ are the set of basis markings of N with respect to the partition π_1 and π_2, respectively.*

Proof: Let $\mathcal{M}(M_b)$ denote the set:

$$\{M \mid \exists t, \mathbf{y}, (M_b, (t, \mathbf{y}), M) \in \Delta\}.$$

Since $M_0 \in \mathcal{M}_1$, $M_0 \in \mathcal{M}_2$, for $t \in T_{E_2}$ and $(M_0, (t, \mathbf{y}_\sigma), M_1) \in \Delta_2$ in $\mathcal{B}_2$, we have $(M_0, (t, \mathbf{y}_{\sigma\uparrow T_{E_1}}), M_1) \in \Delta_1$ due to $T_{E_2} \supseteq T_{E_1}$, and hence $M_1 \in \mathcal{M}_1(M_0)$ holds. This indicates that $\mathcal{M}_1(M_0) \supseteq \mathcal{M}_2(M_0)$. This reasoning can be repeatedly applied for $M \in \mathcal{M}_1(M_0) \cap \mathcal{M}_2(M_0)$ and so on. Since $\mathcal{M}_1(\cdot) \supseteq \mathcal{M}_2(\cdot)$, $\mathcal{M}_1 = \bigcup \mathcal{M}_1(\cdot) \supseteq \bigcup \mathcal{M}_2(\cdot) = \mathcal{M}_2$ holds, which implies $|\mathcal{M}_1| \geq |\mathcal{M}_2|$. ■

Proposition 1 shows that if the set of implicit transitions of π_1 is a superset of that of π_2, then the number of basis markings under π_1 is less than, or at most equal to, that of basis markings under π_2.

Example 4 (Ex. 3 Continued) *Let us again consider the Petri net in Figure 1 and the five basis partitions listed in Table 2 satisfying $T_{I_0} \subset T_{I_1} \subset T_{I_2} \subset T_{I_3}$. The size of the BRGs ($|\mathcal{M}_i|$) and the time to construct them are listed in Tables 3 and 4, respectively. For instance, in Run 1 in Table 3, for $s = 2, v = 1$ and π_2, the number of basis markings is 14. We note that in the case $T_{I_0} = \emptyset$, the corresponding BRG is identical to the reachability graph, i.e., $\mathcal{M} = R(N, M_0)$.*

Table 3: The number of basis markings $\mathcal{M}$ at different instances in Example 4

Run	s	v	π_0	π_1	π_2	π_3	π_4
1	2	1	67	33	14	6	6
2	4	3	783	314	55	15	20
3	6	5	4298	1388	140	28	42
4	8	7	16026	4280	285	45	72
5	10	9	46981	10647	506	66	110
6	20	19	-	-	3311	231	420
7	30	29	-	-	10416	496	930
8	40	39	-	-	-	861	1640

Table 4: The time (sec) required to compute BRGs in Example 4.

Run	s	v	π_0	π_1	π_2	π_3	π_4
1	2	1	<0.1	0.2	<0.1	<0.1	<0.1
2	4	3	3.0	2.8	0.4	0.1	0.1
3	6	5	133	23	1.1	0.2	0.3
4	8	7	1771	156	2.5	0.3	0.5
5	10	9	15667	846	5.2	0.5	0.7
6	20	19	o.t.	o.t.	87	1.9	3.3
7	30	29	o.t.	o.t.	674	4.7	9.5
8	40	39	o.t.	o.t.	o.t.	9.9	22

From Tables 3 and 4 we can see that with the increase of the size of T_I both the size of the BRG and the time to construct it decrease. For instance, for Run 5 ($s = 10$ and $v = 9$), the number of all reachable markings is 46,981. When T_I increases from $\emptyset$ to $\{t_2, t_5, t_7\}$, $\{t_2, t_3, t_5, t_7\}$, and $\{t_2, t_3, t_4, t_5, t_7\}$, the number of basis markings in the corresponding BRG decreases rapidly from 46981 to 10467, 506, and 66 while the computational time decreases from 15667s (>4 hours) to 846s (14 minutes), 5s, and <1s, respectively. □

Definition 9 *A basis partition $\pi = (T_E, T_I)$ is called a* max-cardinality-T_I basis partition *if there does not exist any other basis partition $\pi' = (T'_E, T'_I)$ such that $|T'_I| > |T_I|$, i.e., it has a maximum cardinality set of implicit transitions.* □

By Proposition 1, one may intuitively prefer a basis partition which possesses a transition T_I with the largest cardinality. A max-cardinality-T_I basis partition can be found by solving the following discrete optimization problem.

Problem 1 *Given a net $N = (P, T, Pre, Post)$, determine a transition subset T_I such that:*

$$\begin{cases} \max & |T_I| \\ s.t. & T_I \subseteq T \\ & T_I\text{-induced subnet is acyclic}. \end{cases} \tag{2}$$

Unfortunately Problem 1 cannot be solved efficiently, since it is NP-hard with respect to the scale of the net. In fact, we show that the *maximal acyclic induced subgraph problem* that is known to be NP-complete [30] can be reduced to the decision version of Problem 1.

Definition 10 *Given a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, a graph $\mathcal{G}' = (\mathcal{V}', \mathcal{E}')$ is said to be an* induced subgraph *of $\mathcal{G}$ if $\mathcal{V}' \subseteq \mathcal{V}$ and $\mathcal{E}' = \mathcal{E} \cap (\mathcal{V}' \times \mathcal{V}')$.* □

In plain words, $\mathcal{G}'$ is an induced subgraph of $\mathcal{G}$ if it has exactly the edges that appear in $\mathcal{G}$ over the same vertex set.

Problem 2 *[Maximal Acyclic Induced Subgraph] Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and $k \in \mathbb{N}$, determine if there exists an acyclic induced subgraph $\mathcal{G}' = (\mathcal{V}', \mathcal{E}')$ such that $|\mathcal{V}'| \geq k$.*

Theorem 2 *In Problem 2 $|\mathcal{V}'| \geq k$ holds if and only if in Problem 1 $|T_I|_{max} \geq k$ holds.*

Proof: Given an arbitrary graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, we construct a Petri net $N = (P, T, Pre, Post)$ as follows:

1. For each vertex $v_i \in \mathcal{V}$, add a transition t_i to T;
2. For each $(v_i, v_j) \in \mathcal{E}$, add p_{ij} to P and let $Pre(p, t_i) = Post(p, t_j) = 1$.

Then from $\mathcal{G}$, a Petri net N can be obtained. It has the same topology as $\mathcal{G}$. Obviously the maximal induced acyclic subgraph $(\mathcal{V}', \mathcal{E}')$ has a cardinality $|\mathcal{V}'| \geq k$ if and only if in N some transition set $T_I \subseteq T$ whose induced subnet is acyclic satisfies $|T_I| \geq k$. ■

Although the problem to determine a max-cardinality-T_I basis partition is NP-hard, in practice a brute-force method like *Breadth-First-Search* may still be applied for nets with small structures. For a large-sized net, the following Algorithm 3 can be used to obtain a *max-set-TI basis partition* basis partition π in polynomial time. Here we use $\mathcal{C}_N$ to denote the *maximal strongly connected component* (MSCC) of a net N, and $|\mathcal{C}_N|$ denotes the total number of places and transitions in it. For simplicity we also use N' to denote the T_I-induced subnet.

Definition 11 *A basis partition $\pi = (T_E, T_I)$ is called a* max-set-T_I basis partition *if there does not exist other basis partition $\pi' = (T'_E, T'_I)$ such that $T'_I \supsetneq T_I$, i.e., it has a maximal set of implicit transitions.* □

Algorithm 3 works in a straightforward manner. The acyclicity of the T_I-induced subnet can be determined by checking the MSCC of N'. In the first stage of the algorithm, transitions in T_I are iteratively

Algorithm 3 Determination of a max-set-T_I basis partition

Input: A net $N = (P, T, Pre, Post)$
Output: A basis partition $\pi = (T_E, T_I)$
1: Let $T_I = T$;
2: **while** $|\mathcal{C}_{N'}| > 1$ **do**
3: Select an arbitrary $t \in \mathcal{C}_{N'}$, let $T_I = T_I \setminus \{t\}$;
4: **end while**
5: Let $T_{temp} = T \setminus T_I$;
6: **while** $T_{temp} \neq \emptyset$ **do**
7: Select an arbitrary $t \in T_{temp}$
8: Let $T_I = T_I \cup \{t\}$, $T_{temp} = T_{temp} \setminus \{t\}$;
9: If $|\mathcal{C}_{N'}| > 1$, then let $T_I = T_I \setminus \{t\}$;
10: **end while**
11: Output $\pi = (T_E, T_I)$.

removed to break all cycles. In the second stage, by iteratively running Steps 6 to 10 each transition not in T_I is tentatively put into T_I if it does not introduce a cycle. A max-set-T_I basis partition is finally obtained when T_I reaches a fixed point.

Theorem 3 *Algorithm 3 has polynomial complexity* $O(|P| \cdot |T|^2)$.

Proof: By considering N' as a digraph containing $|P| + |T_I|$ vertexes, computing $\mathcal{C}_{N'}$ has complexity $O(|\mathcal{V}| + |\mathcal{E}|)$ (e.g., by *Tarjan's Algorithm*) which is $O(|P| \cdot |T|)$. Such a procedure repeats at most $2 \cdot |T|$ times. Hence the global complexity of Algorithm 3 is $O(|P| \cdot |T|^2))$. ■

We conclude this section with the following comments. First, despite the fact that Proposition 1 ensures that the number of basis markings in the BRG usually decrease when the set T_I increase, we cannot conclude that $|\mathcal{M}_1| \leq |\mathcal{M}_2|$ if $|T_{I_1}| \geq |T_{I_2}|$, and it is also possible to construct examples in which $|T_{I_1}| > |T_{I_2}|$ and $|\mathcal{M}_1| < |\mathcal{M}_2|$. In fact, two different T_{I_1} and T_{I_2} with equal cardinality may produce sets $\mathcal{M}_1$ and $\mathcal{M}_2$ of rather different cardinality, as shown in Tables 1 and 3, entries π_3 and π_4. Second, there could be classes of PNs for which the basis partition naturally derives by the physical interpretation of transitions. For example, for Petri nets with unobservable transitions, if the unobservable subnet is acyclic, a natural basis partition could be $\pi = (T_E, T_I)$ with $T_E = T_o$ (observable transitions) and $T_I = T_{uo}$ (unobservable transitions). However, one may also apply another basis partition $\pi' = (T'_E, T'_I)$ where $T'_E \subset T_o$ and $T'_I \supset T_{uo}$ (T'_I-induced subnet is acyclic) to obtain a more compact BRG than that of π due to the monotonicity of $|\mathcal{M}|$ with respect to T_I.

5 Generalized Marking Reachability Problems Based on BRG Analysis

In this section we consider the marking reachability problems in which the set of target markings is characterized by an OR-AND GMEC, since in many supervisory control problems the forbidden states can be characterized by an OR-AND GMEC.

Definition 12 *A* Generalized Mutual Exclusion Constraint *(GMEC) is a pair* $(\boldsymbol{w}, k)$ *where* $\boldsymbol{w} \in \mathbb{Z}^m$ *and* $k \in \mathbb{N}$*. A GMEC defines a set of legal markings:*

$$\mathcal{L}_{(\boldsymbol{w},k)} = \{M \in \mathbb{N}^m \mid \boldsymbol{w}^T \cdot M \leq k\}$$

An OR-AND GMEC *is a set* $W = \{(\boldsymbol{W}_1, \boldsymbol{k}_1), \ldots, (\boldsymbol{W}_r, \boldsymbol{k}_r)\}$ *(*$(\boldsymbol{W}_i \in \mathbb{Z}^{m \times s_i}$ *and* $\boldsymbol{k} \in \mathbb{N}^{s_i}$*) that defines a set of legal markings:*

$$\mathcal{L}_W = \{M \in \mathbb{N}^m \mid (\exists i \in \{1, \ldots, r\}) \boldsymbol{W}_i^T \cdot M \leq \boldsymbol{k}_i\}.$$

□

Problem 3 *[Generalized Marking Reachability Problem] Given a Petri net* $\langle N, M_0 \rangle$ *and a set of markings* $\mathcal{L}_W$ *characterized by an OR-AND GMEC* W*, determine if* $R(N, M_0) \cap \mathcal{L}_W \neq \emptyset$*.* □

If $\mathcal{L}_W$ is a singleton, then Problem 3 reduces to the classical marking reachability problem, i.e., determine if a marking M is reachable.

In some practical cases, one would like to know not only if the target set is reachable but also the trajectory to reach it, i.e., to find a firing sequence from the initial marking to reach a target marking (among the target set) with a minimal total cost. We hence define the minimal cost problem as Problem 4.

Definition 13 *Given* $\langle N, M_0 \rangle$*, a* cost vector $\boldsymbol{g} \in \mathbb{N}^n$ *assigns a weight to each transition* $t \in T$*. The* cost *of a firing sequence* σ *is defined as* $\boldsymbol{g}^T \cdot \boldsymbol{y}_\sigma$*.* □

Problem 4 *Given* $\langle N, M_0 \rangle$*, a cost vector* $\boldsymbol{g}$*, and* $\mathcal{L}_W$*, determine a firing sequence* σ *with minimal cost such that* $M_0[\sigma\rangle M \in \mathcal{L}_W$*. Note that the minimal cost to reach* $\mathcal{L}_W$ *is* $+\infty$ *if no marking in* $\mathcal{L}_W$ *is reachable.*

It is clear that Problem 3 can be reduced to Problem 4 since the minimal cost to reach $\mathcal{L}_W$ is finite if and only if $\mathcal{L}_W$ is reachable. Although there exist a few methods based on Petri net refinement and abstraction [13, 14] to solve Problem 3, they highly depend on some particular substructures. Moreover, since the net

structure is abstracted by these methods, it is difficult to reconstruct a firing sequence with the minimal cost. In this section we show that if there exists a basis partition with respect to which the BRG is finite, then Problem 4 can be solved by the analysis of a weighted digraph associated with the BRG.

First, we present an algorithm to transform a given (finite) BRG into a weighted digraph, called the *basis cost graph*, denoted by $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Specifically, for two nodes $v_1, v_2 \in \mathcal{V}$, we use $g(v_1, v_2) = x \geq 0$ to denote that there exists an edge weighted x from v_1 to v_2 (i.e., $(v_1, v_2) \in \mathcal{E}$), and we use $g(v_1, v_2) = +\infty$ to denote that there is no edge from v_1 to v_2. Let $\mathbf{g}_I \in \mathbb{N}^{n_I}$ denote the cost vector $\mathbf{g}$ related to implicit transitions, i.e., $g_I(t) = g(t)$ for $t \in T_I$.

Algorithm 4 Construction of a basis cost graph

Input: A BRG $\mathcal{B} = (\mathcal{M}, Tr, \Delta, M_0)$, a cost vector $\mathbf{g}$, and an OR-AND GMEC W.
Output: A weighted digraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$.

Let $\mathcal{V} = \mathcal{M}$;
for all $M_1, M_2 \in \mathcal{V}$, let $g(M_1, M_2) = +\infty$;
for all $M_1, M_2 \in \mathcal{M}$, **do**
  for all $(M_1, (t, \mathbf{y}), M_2) \in \Delta$, **do**
    let $Q = \mathbf{g}_I^T \cdot \mathbf{y} + g(t)$;
    let $g(M_1, M_2) = \min\{g(M_1, M_2), Q\}$;
  end for
end for
let $\mathcal{V} = \mathcal{V} \cup \{M_{tar}\}$;
for all $M \in \mathcal{V}$, **do**
  let $g(M, M_{tar}) = +\infty$;
  for all $(\mathbf{W}_i, \mathbf{k}_i) \in W$, **do**
    determine $\mathbf{y} \in \mathbb{N}^{n_I}$:

$$\begin{cases} \min & \mathbf{g}_I^T \cdot \mathbf{y} \\ s.t. & M + C_I \cdot \mathbf{y} \geq \mathbf{0} \\ & \mathbf{W}_i^T \cdot (M + C_I \cdot \mathbf{y}) \leq \mathbf{k}_i \end{cases} \tag{3}$$

    let $g(M, M_{tar}) = \min\{g(M, M_{tar}), \mathbf{g}_I^T \cdot \mathbf{y}\}$;
  end for
end for
output $\mathcal{G} = (\mathcal{V}, \mathcal{E})$.

Algorithm 4 works as follows. In the first stage, a BRG is converted into a weighted digraph $(\mathcal{V}, \mathcal{E})$ in which the vertex set is the set of basis markings. For two vertices M_1 and M_2 such that there exists some t satisfying $(M_1, (t, \mathbf{y}), M_2) \in \Delta$, the edge weight from M_1 to M_2 is the minimal firing cost from M_1 to reach M_2, computed according to the minimal explanation from M_1 to reach M_2. In the second stage, a new *target vertex* M_{tar} representing the target set $\mathcal{L}_W$ is added to $\mathcal{V}$. For each basis marking M, the cost from M to reach $\mathcal{L}_W$ is assigned to the edge (M, M_{tar}).

Example 5 (Ex. 4 Continued) *Consider the net in Figure 1 and its BRG in Figure 2. Suppose that we have a target marking set $\mathcal{L}_W$ in which each marking satisfies $(M(p_4) \geq 1) \wedge (M(p_7) \geq 1)$ and a cost vector $\boldsymbol{g} = [3, 2, 2, 2, 1, 7, 1, 5]^T$. By Algorithm 4 the corresponding basis cost graph is depicted in Figure 5. The*

circles represent the basis markings and the solid arrows with weights represent the weighted directed edges. Each hollowed arrow with a weight represents the weighted edge from the corresponding basis marking (shaded) to the target vertex M_{tar} (not drawn in order to simplify the figure). □

Definition 14 *Given a basis cost graph $\mathcal{G}$ and a path $\mathcal{P} = M_0 M_1 \cdots M_{x-1} M_x M_{tar}$ where $M_0 \dots M_x \in \mathcal{M}$, we can reconstruct a trajectory*

$$M_0[\sigma_1 t_1\rangle M_1 \cdots [\sigma_x t_x\rangle M_x[\sigma_{x+1}\rangle M_{x+1}$$

in which (1) σ_i is the minimal explanation of t_i at M_{i-1}, for $1 \leq i \leq x$, (2) at M_{i-1} it is of the minimal cost to fire $\sigma_i t_i$ to reach M_i, for $1 \leq i \leq x$, and (3) at M_x it is of the minimal cost to fire σ_{x+1} to reach $M_{i+1} \in \mathcal{L}_W$. Such a sequence $\sigma = \sigma_1 t_1 \cdots \sigma_x t_x \sigma_{x+1}$ is called a feasible firing sequence *of $\mathcal{P}$.* □

Since the T_I-induced subnet is acyclic, a firing sequence σ can be easily recovered from $\mathbf{y}_\sigma$ by firing transitions from upstream to downstream. Now we show that Problem 4 can be reduced to the *shortest path problem* in the basis cost graph, which can be efficiently solved, e.g., by *Dijkstra's algorithm*, with complexity is $O(|\mathcal{M}|^2)$.

Theorem 4 *Given a BRG $\mathcal{B}$ with a cost vector $\mathbf{g}$ and an OR-AND GMEC $\mathcal{L}_W$, let $\mathcal{G}$ be the basis cost graph by Algorithm 4 and $\mathcal{P}_{min}$ be a shortest path from M_0 to the target vertex M_{tar} in $\mathcal{G}$. Then any feasible firing sequence σ of $\mathcal{P}_{min}$ is a minimal cost firing sequence from M_0 to reach $\mathcal{L}_W$. In particular, the minimal cost is $+\infty$ if and only if $\mathcal{L}_W$ is not reachable.*

Proof: Given σ as a feasible firing sequence of $\mathcal{P}_{min}$, it is trivial that the minimal cost is $+\infty$ if and only if $\mathcal{L}_W$ is not reachable. Now we claim that there does not exist $\sigma' \neq \sigma$ such that $M_0[\sigma'\rangle M \in \mathcal{L}_W$ and $\mathbf{g}^T \cdot \mathbf{y}_{\sigma'} < \mathbf{g}^T \cdot \mathbf{y}_\sigma$. We prove this by contradiction.

Suppose that there exists another firing sequence $\sigma' = \sigma_1 t_1 \sigma_2 t_2 \cdots \sigma_x t_x \sigma_{x+1}$ where $t_i \in T_I$ for $1 \leq i \leq x$, $\sigma_i \in T_I^*$ for $1 \leq i \leq x+1$, $M_0[\sigma'\rangle M \in \mathcal{L}_W$, and $\mathbf{g}^T \cdot \mathbf{y}_{\sigma'} < \mathbf{g}^T \cdot \mathbf{y}_\sigma$. The firing of σ' from M_0 can be written as:

$$M_0[\sigma_1 t_1\rangle M_1[\sigma_2 t_2\rangle M_2 \cdots M_{x-1}[\sigma_x t_x\rangle M_x[\sigma_{x+1}\rangle M \in \mathcal{L}_W.$$

If σ_1 is a minimal explanation of t_1, then M_1 is a basis marking. If σ_1 is not minimal, there necessarily exists a minimal explanation σ_1' satisfying $M[\sigma_1'\rangle M_1'[t_1\rangle M_1''$. Then there must exist a firing sequence $\sigma_1'' \in T_I^*$ satisfying $\mathbf{y}_{\sigma_1''} = \mathbf{y}_{\sigma_1} - \mathbf{y}_{\sigma_1'} - C_I(\cdot, t_1)$ and $M_1''[\sigma_1''\rangle M_1$. By

$$\begin{cases} M_0 + C \cdot \mathbf{y}_{\sigma_1} + C_I(\cdot, t_1) = M_1'' \\ M_1'' + C \cdot (\mathbf{y}_{\sigma_1} - \mathbf{y}_{\sigma_1'} - C_I(\cdot, t_1)) = M_1 \geq \mathbf{0}, \end{cases} \tag{4}$$

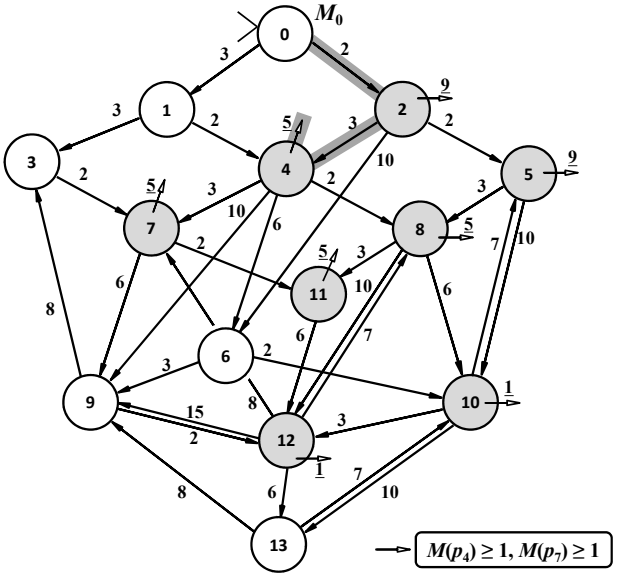

Figure 5: The basis cost graph for Examples 5 and 6.

we have $M_1'' + C_I \cdot \mathbf{y}_{\sigma_1''} = M_1 \geq \mathbf{0}$, and σ_1'' is firable at M_1'' since the T_I-induced subnet is acyclic. Then we have the following firing trajectory:

$$M_0[\sigma_1' t_1\rangle M_1''[\sigma_1'' \sigma_2 t_2\rangle M_2 \cdots M_{x-1}[\sigma_x t_x\rangle M_x[\sigma_{x+1}\rangle M.$$

Now, if $\sigma_1'' \sigma_2$ is not a minimal explanation of t_2, by letting M_1'' be the initial marking, the above procedure can be repeated. Finally we can obtain a new firing sequence $\bar{\sigma}' = \sigma_1' t_1 \sigma_2' t_2 \cdots \sigma_x' t_x \sigma_{x+1}'$ satisfying:

$$M_0[\sigma_1' t_1\rangle M_1''[\sigma_2' t_2\rangle M_2'' \cdots M_{x-1}''[\sigma_x' t_x\rangle M_x''[\sigma_{x+1}'\rangle M \in \mathcal{L}_W$$

where each σ_i' is a minimal explanation of t_i for $1 \leq i \leq x$. This indicates that all M_i''s are basis markings for $1 \leq i \leq x$. Notice that $\bar{\sigma}'$ has the same Parikh vector as σ'. We have

$$\mathbf{g}^T \cdot \mathbf{y}_{\bar{\sigma}'} = \mathbf{g}^T \cdot \mathbf{y}_{\sigma'} < \mathbf{g}^T \cdot \mathbf{y}_{\sigma}.$$

This indicates that there exists a path in $(\mathcal{V}, \mathcal{E})$ with a lower cost than $\mathcal{P}_{min}$, which contradicts the fact that $\mathcal{P}_{min}$ is the shortest path in the graph. ■

By Theorem 4, Problem 4 can be solved if there exists a basis partition such that the BRG is finite, by determining the shortest path in its basis cost graph.

Example 6 (Ex. 5 Continued) *Consider the net in Figure 1 with its basis cost graph given in Figure 5. By*

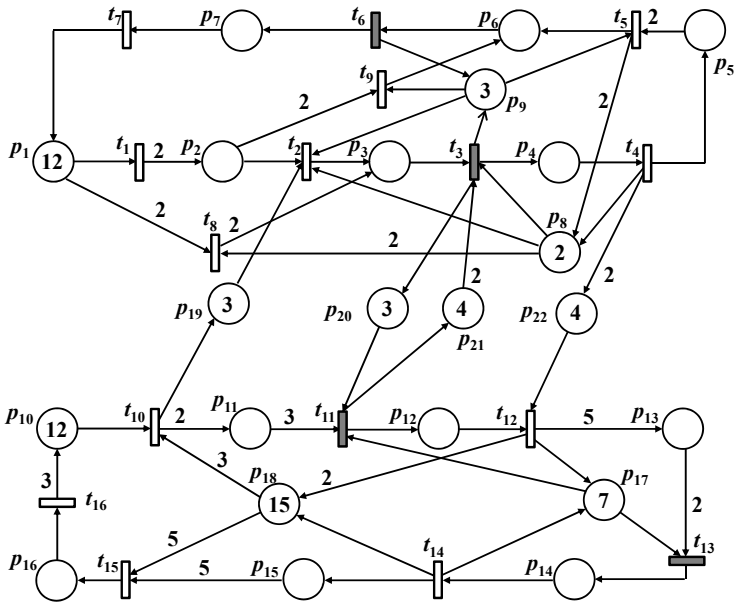

Figure 6: A manufacturing example.

solving the shortest path problem we obtain a minimal path $\mathcal{P}_{min} = M_0[\sigma_1 t_4\rangle M_2[\sigma_2 t_1\rangle M_4[\sigma_3\rangle M \in \mathcal{L}_W$ *(marked in gray). From the minimal explanation vectors,* σ_1*,* σ_2*, and* σ_3 *can be easily reconstructed. Hence we have the following firing trajectory:*

$$M_0[\sigma\rangle M \in \mathcal{L}_W, \sigma = t_4 t_1 t_2 t_3 t_5$$

which is of the minimal cost to reach a marking satisfying $M(p_4) \geq 1$ *and* $M(p_7) \geq 1$. □

By Theorem 4, Problem 4 can be solved by first constructing the BRG and then solving the shortest path problem in its cost graph. The main complexity of this approach comes from the computation of the BRG and solving Eq. (3). To solve the shortest path problem in the cost graph containing $|\mathcal{M}|+1$ vertices) is much easier than that in the reachability graph containing $|R(N, M_0)|$ vertices. Although for each basis marking a series of ILPPs need to be solved, simulations by Matlab show that the total computational load of solving Eq. (3) is usually negligible (approximately 5%) comparing with that of constructing the BRG. Moreover, if the incidence matrix of the net is *totally unimodular*, then the ILPP for checking if M belongs to $R_I(M_b)$ can be relaxed to a *linear programming problem* and hence can be solved in polynomial time [31], which could further reduces the computational effort.

We conclude this section with a complete example taken from the manufacturing domain.

Example 7 *Consider the Petri net* $\langle N, M_0\rangle$ *in Figure 6. It represents a manufacturing system that contains two production lines* $p_1 t_1 p_2 t_2 p_3 t_3 p_4 t_4 p_5 t_5 p_6 t_6 p_7 t_7 p_1$*,* $p_{10} t_{10} p_{11} t_{11} p_{12} t_{12} p_{13} t_{13} p_{14} t_{14} p_{15} t_{15} p_{16} t_{16} p_{10}$*, three*

types of stationary robots p_9, p_{17}, p_{18}, and two types of mobile robots $p_{19}p_{20}$ and $p_{21}p_{22}$. This net, which has 22 places and 16 transitions connected with weighted arcs, has a reachability space containing $1,393,559$ markings. By using a basis partition (T_I, T_U) where $T_I = \{t_3, t_6, t_{11}, t_{13}\}$ (marked as solid bars) there are only 618 basis markings that can be computed by Matlab in 9s.

Now suppose that we are interested in the problem: finding a shortest sequence from M_0 to reach some marking that enables t_3, t_6, t_{11}, and t_{13} simultaneously, while $\boldsymbol{g} = \boldsymbol{1}$ (note that the problem in this case reduces to the minimal firing sequence problem). The target set $\mathcal{L}_W$ can be described by the following OR-AND GMEC $W = \{(\boldsymbol{W}, \boldsymbol{k})\}$ where $\boldsymbol{W} = -\boldsymbol{1}$ and $\boldsymbol{k} = -\hat{M}$, $\hat{M} = p_3 + p_6 + p_8 + 3p_{11} + 2p_{13} + p_{17} + p_{20} + 2p_{21}$ (i.e., $\hat{M}$ is the minimal marking that enables t_3, t_6, t_{11}, and t_{13} simultaneously). In the basis cost graph constructed by Algorithm 4, there are 91 basis marking from which some marking(s) in $\mathcal{L}_W$ can be reached. Finally by solving the shortest path problem in the basis cost graph containing 619 nodes, the shortest firing sequence is:

$$M_0[t_{10}^2 t_{11} t_1^2 t_2 t_9 t_{10} t_{12}\rangle M.$$

□

At the end of this section we point out that in an unbounded net, the marking reachability problem as well as the minimal cost firing problem studied in this section is even more hard to solve. The classical *coverability graph* does not always provide a sufficient condition for reachability since some crucial information is abstracted. In the next section we study the finiteness of the BRG. For an unbounded net, if we can find a basis partition such that the BRG is finite, then the marking reachability problem in it can also be solved by this approach analogously.

6 Basis Reachability Graph for Unbounded Petri Nets

As discussed in Section III, for a bounded net its BRG is always finite regardless the basis partition. For unbounded nets, however, the BRG can be either finite or infinite. Due to this reason, Algorithm 2 cannot be directly applied to arbitrary unbounded nets since it may not terminate. Therefore the plant nets in previous works [22] are always assumed to be bounded.

In this section we study the conditions for finiteness of a BRG. We introduce a stopping criterion to determine the finiteness of the basis marking set. By this approach BRGs can be used as a compact representation of the reachability space of not only bounded Petri nets but also of some meaningful unbounded ones. All previous discussed results in this paper, e.g., Algorithm 4, can be applied to unbounded nets if its reachability set can be compacted as a finite BRG.

6.1 Some Elementary Results

We first show that if a net does not contain source transitions, then the BRG is finite if and only if the net is bounded.

Definition 15 *A transition t is called a* source transition *if it has no input place, i.e., for all places p, $Pre(p,t) = 0$ holds. The set of all source transitions is denoted as T_s.* □

Theorem 5 *Given a Petri net $\langle N, M_0\rangle$ with $T_s = \emptyset$ and an arbitrary basis partition $\pi = (T_E, T_I)$, the basis marking set $\mathcal{M}$ is finite if and only if $\langle N, M_0\rangle$ is bounded.*

Proof: (If) Since $\mathcal{M} \subseteq R(N, M_0)$, a finite $R(N, M_0)$ implies a finite $\mathcal{M}$ under arbitrary basis partitions.

(Only if) For $\langle N, M_0\rangle$, suppose that $\mathcal{M}$ is finite under $\pi = (T_E, T_I)$. By Corollary 1, $R(N, M_0) = \bigcup_{M_b \in \mathcal{M}} R_I(M_b)$ holds. Since the T_I-induced subnet is acyclic and there is no source transition in it, it implies that for any $M_b \in \mathcal{M}$ its implicit reach $R_I(M_b)$ must be finite. Hence we have $|R(N, M_0)| \leq \sum_{M_b \in \mathcal{M}} |R_I(M_b)| < +\infty$, which indicates that the net is bounded. ■

Now let us consider Petri nets with source transitions. The introduction of source transitions (which makes the net obviously unbounded) does not always make the BRG infinite, and an example will be given in the end of this section. However, the following two properties trivially hold.

Property 1 *[Necessary Condition] The BRG for a partition $\pi = (T_E, T_I)$ is finite only if $T_s \cap T_E = \emptyset$.*

Property 2 *[Necessary Condition] Given a Petri net $\langle N, M_0\rangle$ where $N = (P, T, Pre, Post)$ and $T_s \neq \emptyset$, the BRG is finite only if the net $\langle \tilde{N}, M_0\rangle$ where $\tilde{N} = (P, T \setminus T_s, \tilde{Pre}, \tilde{Post})$ (i.e., all source transitions are removed) is bounded.*

Property 1 trivially holds. For Property 2, we observe that if $\langle \tilde{N}, M_0\rangle$ is unbounded, by Theorem 5 $\mathcal{M}(\tilde{N}, M_0, \tilde{\pi})$ is infinite. Since each firing producing a basis marking in $\mathcal{M}(\tilde{N}, M_0, \tilde{\pi})$ can also produce a basis marking in $\mathcal{M}(N, M_0, \pi)$, which indicates that $\mathcal{M}(N, M_0, \pi)$ is infinite.

Property 1 requires that all source transitions be implicit, otherwise each firing of them would produce a new basis marking. Property 2 indicates that a finite BRG may exist only in case that the unboundedness of the net stems from the existence of source transitions. However, Properties 1 and 2 are still not sufficient to ensure the finiteness of a BRG. To find for a sufficient condition for the finiteness of a BRG we introduce the notion of *complete minimal explanation set*.

6.2 Complete Minimal Explanation Vector Set

Definition 16 *Given a net N, a basis partition π, and a transition $t \in T_E$, the* complete minimal explanation set *of t is defined as: $\Sigma_c(t) = \{\sigma \in T_I^* \mid \exists M \in \mathbb{N}^m, \sigma \in \Sigma_{min}(M,t)\}$, i.e., $\Sigma_c(t)$ consists of all sequences σ's that are minimal explanations of t at some marking (not necessarily reachable). We also define $Y_c(t) = \{\boldsymbol{y}_\sigma \in \mathbb{N}^{n_I} \mid \sigma \in \Sigma_c(t)\}$ as the* complete minimal explanation vector set. □

Proposition 2 *The complete minimal explanation vector set $Y_c(t)$ is finite.*

Proof: Let $Y = \bigcup_{M \in \mathbb{N}^m} Y(M,t)$ be the set consisting of all explanation vectors of t. By Definition 16 the set of minimal elements by the $\leq$ ordering of Y is $Y_c(t)$. Therefore $Y_c(t)$ is finite since the set of minimal vectors by the $\leq$ ordering in any set of nonnegative integer vectors is finite by Dickson's Lemma [32]. ■

We now propose Algorithm 5 that, as shown in Theorem 6, allows one to compute the complete minimal explanation vector set $Y_c(t)$. We use $\mathbf{I}_{a\times b}$ and $\mathbf{0}_x$ to denote the $a \times b$ unitary matrix and the x-dimensional zero vector, respectively. Note that in Step 12 if $A(i^*,\cdot)$ is nonnegative, i.e., there is no $A(i^*,j) < 0$, the condition is considered as being satisfied.

Algorithm 5 Computation of a matrix related to $Y_c(t)$

Input: A Petri net N, a basis partition $\pi = (T_E, T_I)$, and a transition $t \in T_E$
Output: A matrix $[\mathbf{D} \mid \mathbf{B}]$

1: Let $\Gamma = \begin{bmatrix} C_I^T & I_{n_I \times n_I} \\ \mathbf{A} & \mathbf{B} \end{bmatrix}$ where $\mathbf{A} = -Pre(\cdot,t)^T$, $\mathbf{B} = \mathbf{0}_{n_I}^T$;
2: **while** row $[A(i^*,\cdot) \mid B(i^*,\cdot)]$ with no tag exists **do**
3: Choose an element $A(i^*,j^*) < 0$ with no tag;
4: Let $\mathcal{Q}^+ = \{i \mid C_I^T(i,j^*) > 0\}$;
5: **for all** $i \in \mathcal{Q}^+$, **do**
6: Let Row_{new}=$[A(i^*,\cdot) \mid B(i^*,\cdot)] + \Gamma(i,\cdot)$;
7: **if** Row_{new} not exists in $[\mathbf{A} \mid \mathbf{B}]$, **then**
8: add row Row_{new} to $[\mathbf{A} \mid \mathbf{B}]$;
9: **end if**
10: **end for**
11: Mark $A(i^*,j^*)$ as "old";
12: **if** for all $A(i^*,j) < 0$ are marked as "old" **then**
13: Mark row $[A(i^*,\cdot) \mid B(i^*,\cdot)]$ as "old";
14: **end if**
15: **end while**
16: Remove tags. Let $\mathbf{D}$ be the matrix: $\forall A(i,j) \leq 0, D(i,j) = -A(i,j)$, otherwise $D(i,j) = 0$;
17: **if** exist rows i^*, i^{**} in $[\mathbf{D} \mid \mathbf{B}]$ such that $D(i^*,\cdot) \lneq D(i^{**},\cdot)$, $B(i^*,\cdot) \lneq B(i^{**},\cdot)$, **then**
18: delete row i^{**} in $[\mathbf{D} \mid \mathbf{B}]$;
19: **end if**
20: Output $[\mathbf{D} \mid \mathbf{B}]$.

Algorithm 5 is inspired by Algorithm 3.5 in [28], which computes $Y_{min}(M,t)$ from a given M and t. In the algorithm we recognize two stages: Stage 1 (Steps 1 to 15) and Stage 2 (Steps 16 to 20). Stage 1 generates

the full list of all possible explanation vectors of t. In Step 1, the initial matrix $\mathbf{A} = -Pre(\cdot, t)$ indicates that any marking $M \geq -A(1,\cdot) = Pre(\cdot, t)$ would enable t by firing the explanation corresponding to the vector $\mathbf{B} = B(1,\cdot) = \mathbf{0}$. In Step 2 a row of $\mathbf{A}$ with no tag (which indicates that it has not been checked yet) is selected. In Step 3 a place p with some negative component is selected. In Step 4 the set $\mathcal{Q}^+$ consists of indices each of which corresponds to a transition whose firing would increase the tokens in place p. For each index $i \in \mathcal{Q}^+$, by Steps 6 and 7 a new row Row_{new} is obtained by adding $\Gamma(i,\cdot)$ onto $[A(i^*,\cdot) \mid B(i^*,\cdot)]$. The new row $Row_{new} = [A_{new} \mid B_{new}]$ indicates that any marking $M \geq -A_{new}$ would enable t by firing the explanation corresponding to the vector B_{new}. Step 11 marks the place as "old", and by Step 12 the row is marked as "old" if all negative nodes are marked as "old". This procedure is repeatedly applied until all rows in $[\mathbf{A} \mid \mathbf{B}]$ are marked as "old". In the second stage, the matrix $\mathbf{A}$ is converted to $\mathbf{D}$ by Step 16, and by Step 17 all non-minimal elements in $[\mathbf{D} \mid \mathbf{B}]$ are removed. Finally the algorithm ends and outputs the matrix $[\mathbf{D} \mid \mathbf{B}]$. An example to illustrate Algorithm 5 is given in Appendix.

We now show how the set $Y_c(t)$ is related to the output of Algorithm 5.

Theorem 6 *Assume that the matrix $[\boldsymbol{D} \mid \boldsymbol{B}]$ computed by Algorithm 5 has r rows. For each row i, define $\mathbf{y}_i = B(i,\cdot)^T$ and $M_i = D(i,\cdot)^T$, and let $Y = \{\mathbf{y}_i \mid i = 1,\ldots,r\}$. The following properties hold:*

1) For $i = 1,\ldots,r$, the vector $\mathbf{y}_i$ is an explanation vector of t at M_i;

2) For $i = 1,\ldots,r$, for any marking $M \not\geq M_i$, the vector $\mathbf{y}_i$ is not an explanation vector of t at M; and for any marking $M \geq M_i$, the vector $\mathbf{y}_i$ is an explanation vector of t at M;

3) The set Y is the complete set of minimal explanation vectors, i.e., $Y = Y_c(t)$.

Proof: (Part 1) By the matrix manipulation, we have $A(i,\cdot) = C_I \cdot B(i,\cdot) - Pre(\cdot, t)$. Hence $M_i + C_I \cdot B(i,\cdot) - Pre(\cdot, t) = M_i + A(i,\cdot) = D(i,\cdot) + A(i,\cdot) \geq \mathbf{0}$, which indicates that $\mathbf{y}_i$ is an explanation of t at M_i.

(Part 2) By the definition of $D(i,\cdot)$, it is obvious that $D(i,\cdot)$, i.e., M_i, is the minimal nonnegative vector such that $D(i,\cdot) + A(i,\cdot) \geq \mathbf{0}$. On the other hand, if $M[\sigma\rangle M'[t\rangle$, it is trivial that for any $M' \geq M$, $M[\sigma\rangle M'[t\rangle$ holds.

(Part 3) We prove $Y \subseteq Y_c(t)$ and $Y \supseteq Y_c(t)$.

($Y \subseteq Y_c(t)$) Suppose that at a marking M there exists a minimal explanation σ_0 to enable t. Since the implicit subnet is acyclic, σ_0 can be rearranged as a new sequence $\sigma = (t_1)^{x_1} \cdots (t_k)^{x_k}$ in which $t_i \neq t_j$ for $i \neq j$ and for any $t_i \in {}^{\bullet\bullet} t_j$, $(t_i)^{x_i}$ appears prior to $(t_j)^{x_j}$. Let $A_0 = A(1,\cdot)$. By Algorithm 5, at the j-th iteration, the j-th single transition in σ will be picked and the corresponding row in Γ is added to A_{j-1} to obtain a new row A_j. Therefore when Step 15 ends, in $\mathbf{B}$ there exists a row $\mathbf{y}_\sigma = \mathbf{y}_{\sigma_0}$. Since $\mathbf{y}_\sigma$ is minimal,

there does not exist another row i^{**} in $[\mathbf{D} \mid \mathbf{B}]$ such that $D(i^*, \cdot) \lneq D(i^{**}, \cdot)$ and $B(i^*, \cdot) \lneq B(i^{**}, \cdot)$. Therefore $\mathbf{y}_\sigma$ remains in $\mathbf{B}$ till the end.

($Y \supseteq Y_c(t)$) By contradiction, if $B(i^*, \cdot)$ is not a minimal explanation at M, then by $Y \subseteq Y_c(t)$ there must exist some other row i^{**} such that $B(i^{**}, \cdot) \lneq B(i^*, \cdot)$. Since $B(i^{**}, \cdot)$ is the minimal explanation at M, $M \geq D(i^{**}, \cdot)$ holds. Hence by Step 18 row i^* should be already removed. A contradiction is reached. ■

Algorithm 5 requires a number of iterations equal to that of paths in the implicit subnet leading to transition t. In the worst case, this number may grow exponentially with the diameter (i.e., length of the maximal path) of the implicit subnet. However, in most practical cases this number is quite reasonable and Algorithm 5 has good performance, since each iteration mainly consists of simple additions of vectors.

By Theorem 6 all rows in $\mathbf{B}$ compose the complete minimal explanation vector set $Y_c(t)$. Moreover, for a minimal explanation vector $B(i, \cdot)$, the corresponding $M_i = D(i, \cdot)$ is the minimal marking at which $B(i, \cdot)$ is an explanation of t. Therefore, the matrix $[\mathbf{D} \mid \mathbf{B}]$ can also be used as an index to compute the minimal explanation of $Y_{min}(M, t)$ at a marking M.

Proposition 3 *Given a marking M and a transition $t \in T_E$, its minimal explanation vector set $Y_{min}(M, t)$ can be obtained by the following steps: (1) put all row vectors $B(i^*, \cdot)$ into Y_{temp} if $M \geq D(i^*, \cdot)$, and then (2) compute the minimal element set of Y_{temp} with respect to $\leq$, which is $Y_{min}(M, t)$.*

Proof: Straightforward from Part 2 of Theorem 6. ■

We conclude this subsection with a comment. In following subsections we show that the finiteness of a BRG cannot be determined by simply checking the cover relation among existing basis markings (see Example 9) as we did in checking the finiteness of the reachability graph (i.e., the boundedness of the net). In such cases the complete minimal explanation vector set plays an important role to determine the finiteness of a BRG.

6.3 Potential Explanation Net

In this subsection we introduce a construct called the *potential explanation net* which will be used to characterize the finiteness of the BRG.

Definition 17 *Given a net N with its incidence matrix C, a basis partition $\pi = (T_E, T_I)$, and a sequence $\sigma \in T_I^*$, their* merged *transition is a new transition t_σ such that $C(\cdot, t_\sigma) = \sum y_\sigma(i) \cdot C(\cdot, t_i)$.* □

Transition t_σ is a transition whose firing is equivalent to the firing of all transitions in σ, and will be used as an intermediate tool shortly. Since the implicit subnet is acyclic, the state equation gives a necessary and sufficient condition for reachability, and hence $M[\sigma\rangle$ if and only if $M[t_\sigma\rangle$.

Definition 18 *Given a Petri net $\langle N, M_0\rangle$ with $N = (P, T, Pre, Post)$ and a basis partition $\pi = (T_E, T_I)$, its* potential explanation net *is a Petri net $\hat{N} = (\hat{P}, \hat{T}, \hat{Pre}, \hat{Post})$ constructed as follows:*

1. *its place set is $\hat{P} = P$;*
2. *for each transition $t \in T_E$, $|Y_c(t)|$ transitions denoted by $(t, \sigma_1), \ldots, (t, \sigma_{|Y_c(t)|})$ belong to $\hat{T}$;*
3. *for place $p \in {}^\bullet t \cap t_{\sigma_i}{}^\bullet$,*

$$\begin{cases} \hat{Pre}(p, (t, \sigma_i)) = \\ \quad \max\{Pre(p, t) - Post(p, t_{\sigma_i}), 0\}, \\ \hat{Post}(p, (t, \sigma_i)) = \\ \quad Post(p, t) - \min\{Pre(p, t) - Post(p, t_{\sigma_i}), 0\}; \end{cases} \tag{5}$$

for other place p,

$$\begin{cases} \hat{Pre}(p, (t, \sigma_i)) = Pre(p, t) + Pre(p, t_{\sigma_i}), \\ \hat{Post}(p, (t, \sigma_i)) = Post(p, t) + Post(p, t_{\sigma_i}). \end{cases} \tag{6}$$

□

In other words, transition (t, σ) can be considered as a transition merged from t and t_σ, while the places p's belonging to ${}^\bullet t \cap t_{\sigma_i}{}^\bullet$ are particularly treated. We give an example to illustrate the construct in Definition 18.

Example 8 *Consider the net N in Figure 7 with a basis partition $T_E = \{t_2, t_3\}$ (shaded bars) and $T_I = \{t_1, t_4\}$. The complete minimal explanation vector set of t_2, i.e., $Y_c(t_2)$, consists of four elements: $\mathbf{0}$, $\mathbf{y}_{t_1}$, $\mathbf{y}_{t_4}$, and $\mathbf{y}_{t_1 t_4}$. Then for t_2 four transitions (t_2, ε), (t_2, t_1), (t_2, t_4), and $(t_2, t_1 t_4)$ are added to $\hat{T}$ and the corresponding column vectors in $\hat{Pre}$ and $\hat{Post}$ are computed by Definition 18. Note that there are two self-loops: $p_2 \leftrightarrow (t_2, t_1)$ and $p_2 \leftrightarrow (t_2, t_1 t_4)$. For t_3 the set $Y_c(t_3)$ consists of only $\mathbf{0}$, and hence (t_3, ε) is added to $\hat{T}$. The resulting potential explanation net $\hat{N}$ is shown in the middle of Figure 7. We also note that intermediate transitions t_{σ_i}'s do not belong to $\hat{T}$.* □

The next proposition shows that the potential explanation net $\hat{N}$ could simulate the original net N.

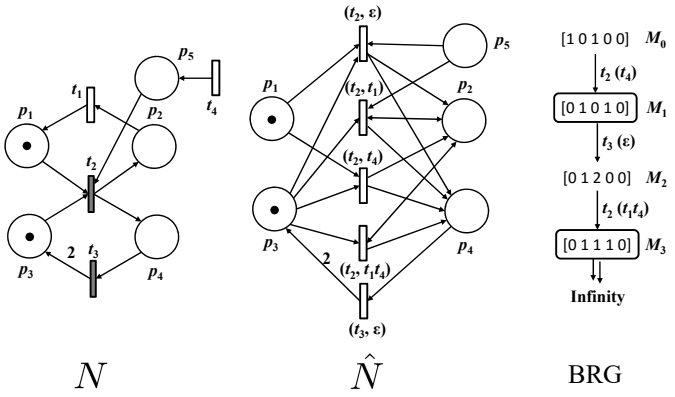

Figure 7: The Petri net for Examples 8 and 9.

Proposition 4 *Given a net N and its potential explanation net $\hat{N}$, for arbitrary M_1 and M_2, $M_1[(t,\sigma)\rangle_{\hat{N}} M_2 \Leftrightarrow (M_1[\sigma t\rangle_N M_2, \sigma \in Y(M_1,t))$.* [4]

Proof: ($\Rightarrow$) Suppose $M_1[(t,\sigma\rangle_{\hat{N}} M_2$. For place $p \in {}^\bullet t \cap t_\sigma^\bullet$, it holds $M_1(p) \geq \hat{Pre}(p,(t,\sigma)) \geq Pre(p,t) - Post(p,t_\sigma)$. Therefore $M_1(p) + Post(p,t_\sigma) \geq Pre(p,t)$. For place $p \notin {}^\bullet t \cap t_\sigma^\bullet$, we have $M(p) \geq \hat{Pre}(p,(t,\sigma)) = Pre(p,t) + Pre(p,t_\sigma)$. Hence $M_1[\sigma\rangle_N M'[t\rangle_N M_2$.

($\Leftarrow$) Suppose $M_1[t_\sigma t\rangle_N M_2$. For place $p \in {}^\bullet t \cap t_\sigma^\bullet$, it holds $M_1(p) - Pre(p,t_\sigma) + Post(p,t_\sigma) \geq Pre(p,t)$. Therefore $M_1(p) \geq Pre(p,t) - Post(p,t_\sigma) + Pre(p,t_\sigma) \geq \hat{Pre}(p,(t,\sigma))$. For place $p \notin {}^\bullet t \cap t_\sigma^\bullet$, we have $M(p) \geq Pre(p,t) + Pre(p,t_\sigma) = \hat{Pre}(p,(t,\sigma))$. Hence $M_1[(t,\sigma)\rangle_{\hat{N}} M_2$. ■

To avoid possible confusions, in the sequel we use $\sigma \in T_I^*$ to denote a sequence of implicit transitions $t_1 t_2 \cdots t_k$ in N while we use $\lambda \in \hat{T}^*$ to denote a firing sequence $(t_1,\sigma_1)(t_2,\sigma_2)\cdots(t_k,\sigma_k)$ in $\hat{N}$. Now we present the first important result of this section.

Theorem 7 *[Sufficient Condition] Given a net $\langle N, M_0\rangle$ and a basis partition π, its BRG is finite if $\langle \hat{N}, M_0\rangle$ is bounded.*

Proof: We show that the BRG of $\langle N, M_0\rangle$ can be simulated by its marked potential explanation net $\langle \hat{N}, M_0\rangle$. Suppose that in N at M_0 an explicit transition $t \in T_E$ can be enabled by a minimal explanation $\sigma \in T_I^*$, $M_0[\sigma t\rangle_N M_1 \in \mathcal{M}$ holds. Then in $\hat{N}$ by firing the transition (t,σ) a same marking can be reached, i.e., $M_0[(t,\sigma)\rangle_{\hat{N}} M_1$. This reasoning can be repeatedly applied and hence $\mathcal{M} \subseteq R(\hat{N}, M_0)$ is true. ■

Corollary 2 *[Sufficient Condition] If the potential explanation net $\hat{N}$ is structurally bounded, the basis marking set $\mathcal{M}$ is bounded for any given initial marking M_0.*

[4] We use $[\cdot\rangle_N$ and $[\cdot\rangle_{\hat{N}}$ to denote the firing of $(\cdot)$ in N and $\hat{N}$, respectively.

Definition 19 *In a potential explanation net $\hat{N}$, a transition (t_1, σ_1) is said to be* covered by *a transition (t_2, σ_2) if $t_1 = t_2$ and $\boldsymbol{y}_{\sigma_1} \lneq \boldsymbol{y}_{\sigma_2}$.* □

The converse of Theorem 7 does not hold: typically we have $\mathcal{M} \subsetneq R(\hat{N}, M_0)$. The reason is that if at some marking $M \in \mathcal{M}$, both (t, σ_1) and (t, σ_2) are enabled and (t, σ_2) covers (t, σ_1), then σ_2 is not a minimal explanation of t at M. Hence $M[(t, \sigma_2)\rangle M' \notin \mathcal{M}$, i.e., the marking M' is not necessarily a basis marking. This is the reason why we cannot determine the finiteness of a BRG by simply checking the cover relation among existing basis markings. To take this into consideration the following two definitions are introduced.

Definition 20 *Given a net N, a sequence σ^+ is a* T-increase *if $C \cdot \boldsymbol{y}_{\sigma^+} \gneq \boldsymbol{0}$. The* unlimited output set *of a T-increase σ^+ is a set of places defined as $\mathcal{U}_{\sigma^+} = \{p \in P \mid C(p, \cdot) \cdot \boldsymbol{y}_{\sigma^+} > 0\}$.* □

Definition 21 *In a net N, the* insufficient input set *of a transition t at a marking M is defined as $\mathcal{I}(M, t) = \{p \in {}^\bullet t \mid M(p) < \hat{Pre}(p, t)\}$.* □

By $\mathcal{U}_{\sigma^+}$ we denote the set of places whose number of tokens increases by the firing of σ^+, and the set $\mathcal{I}(M, t)$ denotes the set of places that have insufficient tokens to enable t at M. Now we present the second important result of this section.

Theorem 8 *[Sufficient and Necessary Condition] Given a net $\langle N, M_0\rangle$ and a basis partition π, the basis marking set $\mathcal{M}$ is infinite if and only if $\exists M_{b_1} \in \mathcal{M}$ such that the following two conditions are satisfied:*

1. *in $\hat{N}$ a T-increase $\lambda^+ \in \hat{T}^*$ is activated at M_{b_1}, i.e.,*

$$M_{b_1}[(t_1, \sigma_1)\rangle_{\hat{N}} M_{b_2}[(t_2, \sigma_2)\rangle_{\hat{N}} \cdots [(t_k, \sigma_k)\rangle_{\hat{N}} M_{b_{k+1}};$$

2. *in the trajectory $M_{b_1}[\lambda^+\rangle_{\hat{N}} M_{b_{k+1}}$ where $\lambda^+ = (t_1, \sigma_1)(t_2, \sigma_2)\cdots(t_k, \sigma_k)$, for each $1 \le j \le k$, for all (t_j, σ'_j) covered by (t_j, σ_j), it holds $\mathcal{I}(M_{b_j}, (t_j, \sigma'_j)) \nsubseteq \mathcal{U}_{\lambda^+}$.*

Proof: (If) Suppose that in the basis marking set $\mathcal{M}$ there exists a basis marking M_{b_1} at which both conditions are satisfied. By Condition 1) the T-increase λ^+ can repeatedly fire at M_{b_1}, and by each firing a new sequence of increasing basis markings is generated. Since Condition 2) is satisfied, no matter how many times λ^+ fires, no (t_j, σ'_j) covered by (t_j, σ_j) can be enabled at $M_{b_j} + q \cdot C \cdot \mathbf{y}_{\lambda^+}$, by firing λ^+ arbitrary q times. This indicates that σ_j is always a minimal explanation of t_j at $M_{b_j} + k \cdot C \cdot \mathbf{y}_{\lambda^+}$. Therefore by repeatedly firing λ^+ at M_{b_1} an increasing basis marking sequence can be generated infinitely long, which indicates that $\mathcal{M}$ is infinite.

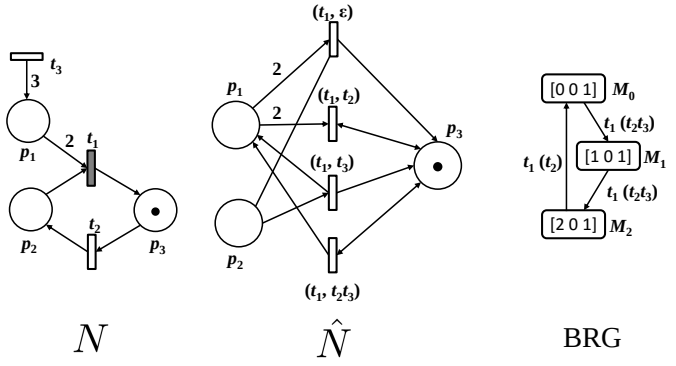

Figure 8: The Petri net for Example 9.

(Only If) If $\mathcal{M}$ is infinite, then in $\mathcal{M}$ there exists a basis marking M_{b_1} from which a repetitive firing sequence $M_{b_1}[\sigma_1 t_1 \cdots \sigma_k t_k\rangle_N M_{b_{k+1}}$ can fire, where $\sigma_i \in T_I^*, t_i \in T_E$. Let $\lambda^+ = (t_1, \sigma_1)\cdots(t_k, \sigma_k)$. Therefore Condition 1) is satisfied. Now we prove that M_{b_1} and λ^+ also satisfy Condition 2) by contradiction.

Suppose that in the trajectory $M_{b_1}[\lambda^+\rangle_{\hat{N}} M_{b_{k+1}}$ there exists a basis marking M_{b_j} with a (t_j, σ_j') covered by (t_j, σ_j) such that $\mathcal{U}_{\lambda^+} \supseteq \mathcal{I}(M_{b_j}, (t, \sigma_j'))$. Without loss of generality we assume that such a basis marking is M_{b_1}, i.e., $j = 1$. For each time $\sigma_1 t_1 \cdots \sigma_k t_k$ fires, all insufficient input places of (t_1, σ_1') will receive some tokens. Hence by repeatedly firing λ^+ for enough times we necessarily reach some $M_{b_1}^*$ at which (t_1, σ_1') is enabled, which implies that σ_i is no longer a minimal explanation to enable t_i at $M_{b_i}^*$. This indicates that such an infinite long sequence cannot occur in the BRG, which contradicts the assumption. ■

Although Theorem 8 involves the potential explanation net $\hat{N}$, in practice to determine if a BRG is finite we do not need to preliminarily construct $\hat{N}$. In fact, the conditions required by Theorem 8 can be embedded into Algorithm 2, providing an on-the-fly stopping criterion whenever the BRG is found to be infinite. This embedding can be done as follows.

1. During the execution of Algorithm 2, once a basis marking M_b (regardless if it already exists in $\mathcal{M}$) is obtained, check if there exists some basis marking M_b' such that $M_b \gneq M_b'$;

2. If such a marking M_b' exists, compute the unlimited place set $\mathcal{U}$ by checking $M_b' - M_b$. For each trajectory from M_b' to M_b, check that for each $M_j[\sigma_j t_j\rangle M_{j+1}$ if $\exists \sigma_j' \leq \sigma_j$ such that $\mathcal{U} \supseteq \mathcal{I}(M_j, (t_j, \sigma_j'))$. The set $\mathcal{I}(M_j, (t_j, \sigma_j'))$ can be computed by checking all $\sigma_j' \leq \sigma_j$ in $Y_c(t_j)$, and this step can be done by vector comparison for at most $|Y_c(t)|$ times;

3. If Step 2 is verified, then Algorithm 2 claims that the BRG is infinite and quits, otherwise it continues.

Example 9 *Consider the net in Figure 7 in which $T_E = \{t_2, t_3\}$ and $T_I = \{t_1, t_4\}$. By applying Algorithm 2, from the initial marking $M_0 = [1, 0, 1, 0, 0]^T$ the first four basis markings are shown on the right in Figure 7.*

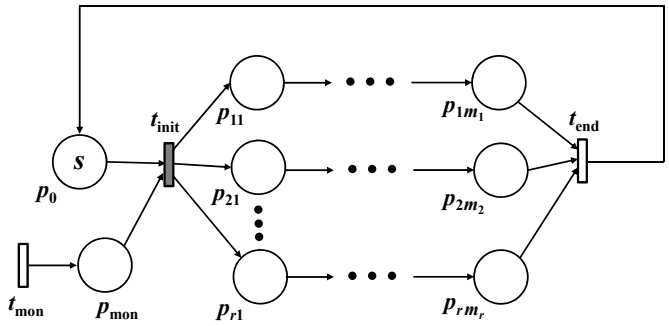

Figure 9: The unbounded Petri net for Example 10.

Since $M_1[(\varepsilon)t_3\rangle M_2[(t_1t_4)t_2\rangle M_3 \gneq M_1$, by computing $M_3 - M_1$, we have $\mathcal{U} = \{p_3\}$. This sequence involves two steps: $M_1[(\varepsilon)t_3\rangle$ and $M_2[(t_1t_4)t_2\rangle$. For $M_1[(\varepsilon)t_3\rangle$, there does not exist an implicit sequence covered by ε. For $M_2[(t_1t_4)t_2\rangle$, there are three possible implicit sequences covered by t_1t_4: ε, t_1, and t_4. Since $\mathcal{I}(M_2, (t_2, \varepsilon)) = \{p_1, p_5\}$, $\mathcal{I}(M_2, (t_2, t_1)) = \{p_5\}$, and $\mathcal{I}(M_2, (t_2, t_4)) = \{p_1\}$, none of them is a subset of $\mathcal{U}$, which indicates that from M_1 no matter how the tokens in p_3 increase by firing $t_3t_1t_4t_2$, none of ε, t_1, and t_4 can be a minimal explanation of t_2. Therefore such a trajectory can repeatedly occur in the BRG and hence it is infinite.

On the other hand, consider the net in Figure 8 in which $T_E = \{t_1\}$ and $T_I = \{t_2, t_3\}$. By applying Algorithm 2, from $M_0 = [0, 0, 1]^T$ a basis marking $M_1 = [1, 0, 1]^T$ can be obtained. Since $M_0[(t_2t_3)t_1\rangle M_1 \gneq M_0$, we have $\mathcal{U} = \{p_1\}$. There exist three sequences ε, t_2, and t_3 covered by t_2t_3, and hence we have $\mathcal{I}(M_0, (t_1, \varepsilon)) = \{p_1, p_2\}$, $\mathcal{I}(M_0, (t_1, t_2)) = \{p_1\}$, and $\mathcal{I}(M_0, (t_1, t_3)) = \{p_2\}$. By $\mathcal{I}(M_0, (t_1, t_2)) \subseteq \mathcal{U}$, although t_2t_3 is a minimal explanation of t_1 at M_0, if the sequence $t_2t_3t_1$ fires sufficient times, $M(p_1)$ would be large enough such that t_2t_3 would no longer a minimal explanation of t_1. As a result, although $M_1 \gneq M_0$, from M_0 the sequence $t_2t_3t_1$ cannot repeatedly occur for infinite times in the BRG. The same reason holds for $M_2 \gneq M_1$. Hence by Algorithm 2 we finally obtain a finite BRG that consists of three basis markings. It also coincides with the fact that the finiteness of a BRG cannot be determined by simply checking the covering relation among existing basis markings. □

Example 10 (Ex. 2 Continued) *Consider the net in Figure 9 consisting of the net in Figure 2 plus a place p_{mon} and a transition t_{mon}. Here, workflows can only be triggered with the permission of a monitor (i.e., $M(p_{mon}) \geq 1$). In such a case, the classical* coverability graph *does not always provide a sufficient condition for reachability. However, by choosing the basis partition (T_E, T_I) where $T_E = \{t_{init}\}$ and $T_I = T \setminus \{t_{init}\}$, we can obtain a finite BRG that is exactly the one in Figure 4, in which $|\mathcal{M}| = s + 1$ does not depend on r (the number of workflows) and on the structure of the workflows. The number of basis markings depends but only linearly from the parameter s.* □

7 Conclusion

In this paper a compact representation of the reachability space of a Petri net, called the basis reachability graph, is proposed, and its properties are extensively studied. The marking reachability problem in a Petri net can be solved by a practically efficient algorithm based on the basis reachability graph. The BRG-based method has wide applicability since it does not rely on particular substructures of the net, and it can be used to precisely characterize the reachability set of not only bounded nets but also some meaningful unbounded systems. One of our future studies would focus on the case that the BRGs. By marking the unbounded places in the BRG as ω we could obtain a *basis coverability graph* which provides a sufficient condition but contains more information than a classical coverability graph. Furthermore, other methods in the literature, such as the BDD [11] and the stubborn set [12] methods, may possibly be incorporated with our approach to further improve its performance. We will investigate this in our future work.

References

[1] F. Basile, R. Cordone, and L. Piroddi, "Integrated design of optimal supervisors for the enforcement of static and behavioral specifications in Petri net models," *Automatica*, vol. 49, no. 11, pp. 3432–3439, 2013.

[2] Z. Y. Ma, Z. W. Li, and A. Giua, "Design of optimal Petri net controllers for disjunctive generalized mutual exclusion constraints," *IEEE Transactions on Automatic Control*, vol. 60, pp. 1774–1785, 2015.

[3] Y. F. Chen and Z. W. Li, "Design of a maximally permissive liveness-enforcing supervisor with a compressed supervisory structure for flexible manufacturing systems," *Automatica*, vol. 47, no. 5, pp. 1028–1034, 2011.

[4] A. Nazeem and S. Reveliotis, "Designing compact and maximally permissive deadlock avoidance policies for complex resource allocation systems through classification theory: the nonlinear case," *IEEE Transactions on Automatic Control*, vol. 57, no. 7, pp. 1670–1684, 2012.

[5] A. Giua, F. DiCesare, and M. Silva, "Generalized mutual exclusion constraints for Petri nets with uncontrollable transitions," in *In Proceedings of the IEEE Int. Conf. on Systems, Man, and Cybernetics*, Chicago, USA, 1992, pp. 947–949.

[6] Z. Y. Ma, Z. W. Li, and A. Giua, "A method to verify the controllability of language specifications in Petri nets based on basis marking analysis," in *In Proceedings of the 54th IEEE Conf. on Decision and Control*, Osaka, Japan, 2015, pp. 1675–1681.

[7] Y. Tong, Z. W. Li, and A. Giua, "On the equivalence of observation structures for Petri net generators," *IEEE Transactions on Automatic Control*, vol. 61, no. 9, 2016.

[8] E. W. Mayr, "Persistence of vector replacement systems is decidable," *Acta Informatica*, vol. 15, no. 3, pp. 309–318, 1981.

[9] R. J. Lipton, "The reachability problem requires exponential space," Research Report 62, Department of Computer Science, Yale University, Tech. Rep., 1976.

[10] J. Desel and J. Esparza, *Free Choice Petri Nets*. Cambridge University Press, Cambridge, UK, 1995.

[11] E. Pastor, J. Cortadella, and M. A. P. na, "Structural methods to improve the symbolic analysis of Petri nets," in *Application and Theory of Petri Nets 1999*, ser. Lecture Notes in Computer Science, S. Donatelli and J. Kleijn, Eds. Springer Berlin Heidelberg, 1999, vol. 1639, pp. 26–45.

[12] A. Valmari, "Stubborn sets for reduced state space generation," in *Advances in Petri Nets 1990*, ser. Lecture Notes in Computer Science, G. Rozenberg, Ed. Springer Berlin Heidelberg, 1991, vol. 483, pp. 491–515.

[13] F. DiCesare, H. Harhalakis, J. M. Proth, M. Silva, and F. B. Vernadat, *Practice of Petri Nets in Manufacturing*. Chapman and Hall, 1993.

[14] I. Suzuki and T. Murata, "A method for stepwise refinement and abstraction of Petri nets," *Journal of Computer and System Sciences*, vol. 27, no. 1, pp. 51–76, 1983.

[15] P. Küngas, "Petri net reachability checking is polynomial with optimal abstraction hierarchies," in *Abstraction, Reformulation and Approximation*. Springer, 2005, pp. 149–164.

[16] S. Christensen and L. Petrucci, "Modular analysis of Petri nets," *The Computer Journal*, vol. 43, no. 3, pp. 57–64, 2000.

[17] T. Miyamoto and K. Horiguchi, "Modular reachability analysis of Petri nets for multiagent systems," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 43, no. 6, pp. 1411–1423, 2013.

[18] J. Ezpeleta, J. M. Colom, and J. Martinez, "A Petri net based deadlock prevention policy for flexible manufacturing systems," *IEEE Transactions on Robotics and Automation*, vol. 11, no. 2, pp. 173–184, 1995.

[19] Z. W. Li and M. C. Zhou, "Elementary siphons of Petri nets and their application to deadlock prevention in flexible manufacturing systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, vol. 34, no. 1, pp. 38–51, 2004.

[20] ——, "Control of elementary and dependent siphons in Petri nets and their application," *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, vol. 38, no. 1, pp. 133–148, 2008.

[21] M. Cabasino, A. Giua, and C. Seatzu., "Fault detection for discrete event systems using Petri nets with unobservable transitions," *Automatica*, vol. 46, no. 9, pp. 1531–1539, 2010.

[22] M. Cabasino, A. Giua, M. Pocci, and C. Seatzu, "Discrete event diagnosis using labeled Petri nets: an application to manufacturing systems," *Control Engineering Practice*, vol. 19, no. 9, pp. 989–1001, 2011.

[23] Y. Tong, Z. W. Li, C. Seatzu, and A. Giua, "Verification of current-state opacity using Petri nets," in *In Proceedings of the 34th American Control Conference*, Chicago, USA, 2015, pp. 1935–1940.

[24] ——, "Verification of initial-state opacity in DES," in *In Proceedings of the 54th IEEE Conf. on Decision and Control*, Osaka, Japan, 2015, pp. 344–349.

[25] ——, "Verification of state-based opacity using Petri nets," *IEEE Transactions on Automatic Control*, p. Submitted, 2015.

[26] Y. Ru, M. Cabasino, A. Giua, and C. Hadjicostis, "Supervisor synthesis for discrete event systems with arbitrary forbidden state specifications," in *In Proceedings of the 47th IEEE Conf. on Decision and Control*, Cancun, Mexico, 2008, pp. 1048–1053.

[27] D. Corona, A. Giua, and C. Seatzu, "Marking estimation of Petri nets with silent transitions," *IEEE Transactions on Automatic Control*, vol. 52, no. 9, pp. 1695–1699, 2007.

[28] A. Giua and C. Seatzu, "Fault detection for discrete event systems using Petri nets with unobservable transitions," in *In Proceedings of the 44th Int. Conf. on Decision and Control and European Control Conference*, Seville, Spain, 2005, pp. 6323–6328.

[29] W. M. P. van der Aalst, "The application of Petri nets to workflow management," *Journal of Circuits, Systems and Computers*, vol. 8, no. 1, pp. 21–66, 1998.

[30] S. Janson, T. Luczak, and A. Rucinski, *Random Graphs*. John Wiley & Sons, Inc., 2000.

[31] A. J. Hoffman and J. B. Kruskal, "Integral boundary points of convex polyhedra," in *50 Years of integer programming 1958-2008*, e. Michael Jünger, Ed. Berlin Heidelberg: Springer-Verlag, 2010, pp. 49–76.

[32] L. E. Dickson, "Finiteness of the odd perfect and primitive abundant numbers with n distinct prime factors," *American Journal of Mathematics*, vol. 35, no. 4, pp. 413–422, 1913.

Appendix

In this appendix we illustrate how to compute $Y_c(t)$ by Algorithm 5 through the following example. Consider the net in Figure 10, in which $T_E = \{t\}, T_I = \{t_1, t_2, t_3\}$. First, the matrix is initialized as:

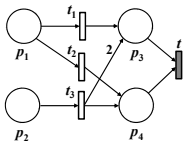

Figure 10: An example to illustrate Algorithm 5.

$$\left[\begin{array}{c|c} C_I^T & \mathbf{1} \\ \hline \mathbf{A} & \mathbf{B} \end{array}\right] = \left[\begin{array}{cccc|ccc} -1 & 0 & 1 & 0 & 1 & & \\ -1 & 0 & 0 & 1 & & 1 & \\ 0 & -1 & 2 & 1 & & & 1 \\ \hline 0 & 0 & -1 & \underline{-1} & 0 & 0 & 0 \end{array}\right]$$

Now in the first column of **A** there are two negative components to be picked. Suppose that we pick the underlined one. By respectively adding Rows 2 and 3 in $[C_I^T \mid \mathbf{1}]$ onto this row, two new rows are obtained and the matrix $[\mathbf{A} \mid \mathbf{B}]$ becomes:

$$\left[\begin{array}{cccc|ccc} 0 & 0 & \underline{-1} & -1_{old} & 0 & 0 & 0 \\ -1 & 0 & -1 & 0 & 0 & 1 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 1 \end{array}\right]$$

Then that -1 is marked as "old". Next we select the underlined -1. By the similar procedure the matrix is updated as:

$$\left[\begin{array}{cccc|ccc} 0 & 0 & -1_{old} & -1_{old} & 0 & 0 & 0 \\ -1 & 0 & \underline{-1} & 0 & 0 & 1 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 1 \end{array}\right]$$

Since Row 5 is identical to Row 3, we remove Row 5, i.e., it is not added into the matrix. By selecting the underlined -1 we have:

$$\left[\begin{array}{cccc|ccc} 0 & 0 & -1_{old} & -1_{old} & 0 & 0 & 0 \\ -1_{old} & 0 & -1_{old} & 0 & 0 & 1 & 0 \\ 0 & -1_{old} & 1 & 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & \underline{-1} & 1 & 0 & 0 \\ -2 & 0 & 0 & 0 & 1 & 1 & 0 \\ -1 & -1 & 1 & 1 & 0 & 1 & 1 \end{array}\right]$$

Note that since it is not possible to increase the -1 in Row 2 Column 1 as well as the -1 in Row 3 Column 2, they are also marked as "old". Then we select the underlined -1 to proceed:

$$\left[\begin{array}{cccc|ccc} 0 & 0 & -1_{old} & -1_{old} & 0 & 0 & 0 \\ -1_{old} & 0 & -1_{old} & 0 & 0 & 1 & 0 \\ 0 & -1_{old} & 1 & 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & -1_{old} & 1 & 0 & 0 \\ -2 & 0 & 0 & 0 & 1 & 1 & 0 \\ -1 & -1 & 1 & 1 & 0 & 1 & 1 \\ -2 & 0 & 0 & 0 & 1 & 1 & 0 \\ -1 & -1 & 2 & 0 & 1 & 0 & 1 \end{array}\right]$$

Row 7 is identical to Row 5 and thus is removed. Since it is not possible to increase any negative components, all of them are marked as "old" and then Algorithm 5 moves to Step 16. Then we have:

$$[\mathbf{A} \mid \mathbf{B}] = \left[\begin{array}{cccc|ccc} 0 & 0 & -1 & -1 & 0 & 0 & 0 \\ -1 & 0 & -1 & 0 & 0 & 1 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & -1 & 1 & 0 & 0 \\ -2 & 0 & 0 & 0 & 1 & 1 & 0 \\ -1 & -1 & 1 & 1 & 0 & 1 & 1 \\ -1 & -1 & 2 & 0 & 1 & 0 & 1 \end{array}\right]$$

From $\mathbf{A}$ we can compute $\mathbf{D}$. By Step 18 Rows 6 and 7 are removed since they cover Row 3. Then Algorithm 5 ends and we finally have:

$$[\mathbf{D} \mid \mathbf{B}] = \left[\begin{array}{cccc|ccc} 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 2 & 0 & 0 & 0 & 1 & 1 & 0 \end{array}\right]$$

The matrix **B** indicates that $Y_c(t) = \{\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3, \mathbf{y}_4, \mathbf{y}_5\}$ where $\mathbf{y}_1 = [0,0,0]^T, \mathbf{y}_2 = [0,1,0]^T, \mathbf{y}_3 = [0,0,1]^T, \mathbf{y}_4 = [1,0,0]^T$, and $\mathbf{y}_5 = [1,1,0]^T$. Each $\mathbf{y}_i$ is an explanation of t at M if and only if $M \geq D(i,\cdot)$.

Now suppose that we want to compute $Y_{min}(M,t)$ where $M = [2,2,0,1]^T$. Then we check **D** to verify if $M \geq D(i,\cdot)$. Since $M \geq D(3,\cdot), D(4,\cdot), D(5,\cdot)$, it indicates that $\mathbf{y}_3, \mathbf{y}_4$, and $\mathbf{y}_5$ are explanations of t at M. By $\mathbf{y}_4 \lneq \mathbf{y}_5$, we delete $\mathbf{y}_5$. Hence we have:

$$Y_{min}(M,t) = \{\mathbf{y}_3, \mathbf{y}_4\} = \{[0,0,1]^T, [1,0,0]^T\}.$$