

# Analysis and Synthesis of Weighted Marked Graph Petri Nets: Exact and Approximate Methods

Raymond Devillers, Thomas Hujsa

► **To cite this version:**

Raymond Devillers, Thomas Hujsa. Analysis and Synthesis of Weighted Marked Graph Petri Nets: Exact and Approximate Methods. *Fundamenta Informaticae*, Polskie Towarzystwo Matematyczne, 2019, 169 (1-2), pp.1 - 30. 10.3233/FI-2019-1837 . hal-02332309

**HAL Id: hal-02332309**

**<https://hal.laas.fr/hal-02332309>**

Submitted on 24 Oct 2019

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

## **Analysis and Synthesis of Weighted Marked Graph Petri Nets: Exact and Approximate Methods**

**Raymond Devillers**

*Département d'Informatique, Université Libre de  
Bruxelles, B-1050 Brussels, Belgium*  
*rdevil@ulb.ac.be*

**Thomas Hujsa\***

*LAAS-CNRS, Université de Toulouse, CNRS,  
Toulouse, France*  
*thujsa@laas.fr*

---

**Abstract.** Numerous real-world systems can be modeled with Petri nets, which allow a combination of concurrency with synchronizations and conflicts. To alleviate the difficulty of checking their behaviour, a common approach consists in studying specific subclasses. In the converse problem of Petri net synthesis, a Petri net of some subclass has to be constructed efficiently from a given specification, typically from a labelled transition system (Its) describing the behaviour of the desired net.

In this paper, we focus on a notorious subclass of persistent Petri nets, the weighted marked graphs (WMGs), also called generalised (or weighted) event (or marked) graphs or weighted T-nets. In such nets, edges have multiplicities (weights) and each place has at most one ingoing and one outgoing transition. Although extensively studied in previous works and benefiting from strong results, both their analysis and synthesis can be further investigated.

We provide new behavioural properties of WMGs expressed on their reachability graph, notably backward persistence and strong similarities between any two sequences sharing the same starting state and the same destination state. Besides, we design a general synthesis procedure aiming at the WMG class. Finally, when no solution to the synthesis problem exists, i.e., when the given Its is not WMG-solvable, we show how to construct a WMG whose reachability graph is a minimal over-approximation of the given Its.

**Keywords:** Weighted Petri net, analysis, synthesis, marked graph, event graph, theory of regions, approximation, persistence.

---

\*Supported by the STAE foundation/project DAEDALUS, Toulouse, France.

## 1. Introduction

Petri nets have proved useful to model numerous artificial and natural systems. Their weighted version allows weights on arcs, making possible the bulk consumption or production of tokens, hence a more compact representation of the systems.

Many fundamental properties of Petri nets are decidable, although often hard to check. Given a bounded Petri net, a naive analysis can be performed by constructing its finite reachability graph, whose size may be considerably larger than the net size. To avoid such a costly computation, subclasses are often considered, allowing to derive efficiently their behaviour from their structure only. This approach has led to various polynomial-time checking methods dedicated to several subclasses, the latter being defined by structural restrictions in many cases [1, 2, 3, 4, 5].

In the domain of Petri net synthesis, a specification has to be implemented by a Petri net, meaning that the behaviour of the Petri net obtained must correspond exactly to the specification. Classical representations of such a specification encompass labelled transition systems (Lts for short), which are rooted directed graphs with labels on the arcs, and a synthesis procedure is meant to build a Petri net of a specific subclass whose reachability graph is isomorphic to a given lts, when possible.

**Weighted marked graphs: applications and previous studies.** In this paper, we focus on marked graphs with weights (also called generalised event graphs or weighted T-nets), a subclass of weighted Petri nets in which each place has at most one input and one output. They can model Synchronous DataFlow graphs [6], which have been fruitfully used to design and analyse many real systems such as embedded applications, notably Digital Signal Processing (DSP) applications [7, 8, 9].

Various characterisations and polynomial-time sufficient conditions of structural and behavioural properties, notably of liveness, boundedness and reversibility, have been developed for WMGs [10, 11] and larger classes [12, 13, 2]. These nets are a special case of persistent systems, in which no transition firing can disable another transition.

**Petri net synthesis: previous studies.** Given a labelled transition system, previous works have proposed algorithms synthesizing a Petri net with an isomorphic reachability graph, sometimes aiming at a Petri net subclass [14, 15, 16]. In the latter case, the objective is to delineate properties of the lts that are specific to the target subclass, so as to determine sufficient and necessary conditions for its synthesizability within the subclass. Ideally, such specific conditions should be easier to check than generic ones, for instance during a pre-synthesis phase. When the given lts has no solution within the desired class, algorithms exist that build a minimal solvable over-approximation of the lts in some subclasses of Petri nets [17, 18].

Marked graphs, i.e., unit-weighted marked graphs, belong to the larger class of choice-free nets, in which each place has at most one output. Both classes benefit from dedicated synthesis algorithms that operate in polynomial time [14, 15, 19, 20, 21]. However, such methods do not yet exist for the intermediate class of marked graphs with arbitrary weights.

**Contributions.** In this paper, we further investigate the class of weighted marked graphs (WMGs), extending our previous conference paper [22]. We delineate new properties of these nets and propose a synthesis procedure aiming at this subclass. In case no WMG-solution exists, we also propose an algorithm building a minimal WMG-solvable over-approximation of the given lts.

First, we provide new structural and behavioural properties of WMGs: we give a comparison property on the sequences starting at the same state and reaching another common state, we show that WMGs are necessarily *backward persistent*, meaning that for all reachable states  $s_1, s_2, s_3$  such that  $s_2[a]s_1$  (i.e.,  $s_1$  is reached from  $s_2$  through the action with label  $a$ ) and  $s_3[b]s_1$ , there exists a reachable state  $s_4$  with  $s_4[b]s_2$  and  $s_4[a]s_3$ . We also develop conditions allowing the existence of a feasible sequence corresponding to a given Parikh vector.

Then, we delineate necessary conditions for the WMG-solvability of an lts, such as backward persistence and the existence of particular cycles. We show, with the help of a counter-example from another subclass, that these conditions are not sufficient for a WMG solution to exist.

Thereafter, we devise a WMG-synthesis procedure, specialising previous methods that were designed for the larger class of choice-free nets.

Finally, and this is the main addition with respect to [22], we show the existence of a minimal WMG-solvable over-approximation of any given lts and exploit a fixed point algorithm to compute it. This allows to exploit the synthesis procedure mentioned above to build a WMG whose language minimally includes the sequences specified by the given lts.

**Organisation of the paper.** In Section 2, we introduce general definitions, notations and properties. In Section 3, we recall some properties of persistent Petri nets and provide new structural and behavioural results on WMGs, including the proof of backward persistence. In Section 4, we describe a synthesis procedure for WMGs. In Section 5, we investigate minimal WMG-solvable over-approximations. As usual, the last section presents our conclusions and perspectives.

## 2. Classical Definitions, Notations and Properties

In the following, we define formally Petri nets, labelled transitions systems and related notions. We also recall classical properties of Petri nets in Proposition 2.1.

**Petri nets, incidence matrices, pre- and post-sets.** A *(Petri) net* is a tuple  $N = (P, T, W)$  such that  $P$  is a finite set of *places*,  $T$  is a finite set of *transitions*, with  $P \cap T = \emptyset$ , and  $W$  is a weight function  $W: ((P \times T) \cup (T \times P)) \rightarrow \mathbb{N}$  setting the weights on the arcs. A *marking* of the net  $N$  is a mapping from  $P$  to  $\mathbb{N}$ , i.e., a member of  $\mathbb{N}^P$ , defining the number of tokens in each place of  $N$ .

A *(Petri net) system* is a tuple  $\zeta = (N, M_0)$  where  $N$  is a net and  $M_0$  is a marking, often called *initial marking*. The *incidence matrix*  $C$  of  $N$  (and  $\zeta$ ) is the integer place-transition matrix with components  $C(p, t) = W(t, p) - W(p, t)$ , for each place  $p$  and each transition  $t$ . For any place  $p$  and any transition  $t$ , we denote by  $C(p)$  the row of  $p$  and by  $C(t)$  the column of  $t$ .

The *post-set*  $n^\bullet$  and *pre-set*  ${}^\bullet n$  of a node  $n \in P \cup T$  are defined as  $n^\bullet = \{n' \in P \cup T \mid W(n, n') > 0\}$  and  ${}^\bullet n = \{n' \in P \cup T \mid W(n', n) > 0\}$ .

**Firings and reachability in Petri nets.** Consider a net  $N = (P, T, W)$ . A transition  $t$  is *enabled* at a marking  $M$  if  $\forall p \in {}^\bullet t, M(p) \geq W(p, t)$ , in which case  $t$  can *occur* at or be *fired* from  $M$ . The firing of  $t$  from  $M$  leads to the marking  $M' = M + C(t)$  and we note it  $M[t]M'$ .

A finite (*firing*) *sequence*  $\sigma = t_1 \dots t_n$  of length  $n \geq 0$  on the set  $T$ , hence with  $t_1, \dots, t_n \in T$ , is a mapping  $\{1, \dots, n\} \rightarrow T$ . Infinite sequences are defined similarly as mappings  $\mathbb{N} \setminus \{0\} \rightarrow T$ .

A sequence  $\sigma$  of length  $n$  is *enabled* (or fireable, or feasible, or realisable) in a system  $\zeta = (N, M_0)$  if the successive states obtained,  $M_0[t_1]M_1 \dots [t_n]M_n$ , satisfy  $M_{k-1}[t_k]M_k$ ,  $\forall k \in \{1, \dots, n\}$ , in which case  $M_n$  is said to be *reachable* from  $M_0$ : we note this as  $M_0[\sigma]M_n$ . If  $n = 0$ ,  $\sigma$  is the *empty sequence*  $\epsilon$ , implying  $M_0[\epsilon]M_0$ . The set of markings reachable from  $M_0$  is noted  $[M_0]$ . A place  $p \in P$  contains *frozen tokens* if, for each reachable marking  $M \in [M_0]$ ,  $M(p) > 0$ . The number of frozen tokens on  $p$  from  $M_0$  is given by  $\min_{M \in [M_0]} M(p)$ , i.e. the maximal number of tokens that can never be used.<sup>1</sup>

The *reachability graph* of  $\zeta$ , noted  $RG(\zeta)$ , is the rooted directed graph  $(V, A, \iota)$  where  $V$  represents the set of vertices  $[M_0]$ ,  $A$  is the set of arcs labelled with transitions of  $T$  such that the arc  $M \xrightarrow{t} M'$  belongs to  $A$  if and only if  $M[t]M'$  and  $M \in [M_0]$ , and  $\iota = M_0$  is the root.

In Figure 1, a weighted system is pictured on the left. Its reachability graph is pictured on the right, where  $v^T$  denotes the transpose of vector  $v$ .

**Petri net subclasses.**  $(N, M_0)$  is *bounded* iff its reachability graph is finite, i.e. there exists an integer  $k$  such that, for each marking  $M$  reachable from  $M_0$  and each place  $p$ ,  $M(p) \leq k$ .  $N$  is *plain* if no arc weight exceeds 1; *choice-free* (CF for short, meaning that the structure does not permit any choice in the usage of tokens: in each place, the tokens may only be used by a single transition) [25, 26] (also called *place-output-nonbranching* in [27]) if  $\forall p \in P, |p^\bullet| \leq 1$ ; *fork-attribution* (FA, where a fork is a transition with at least two output places, and an attribution is a place with at least two input transitions) [26] if it is CF and, in addition,  $\forall t \in T, |t^\bullet| \leq 1$ ; a *weighted marked graph* (WMG, also called weighted T-nets in [10]) if it is CF and, in addition,  $\forall p \in P, |p^\bullet| \leq 1$ .

A WMG is pictured on the left of Figure 1. Well-studied subclasses encompass *marked graphs* [28] which are plain and fulfill  $|p^\bullet| = 1$  and  $|t^\bullet| = 1$  for each place  $p$ , and *T-nets* [1], which are plain and fulfill  $|p^\bullet| \leq 1$  and  $|t^\bullet| \leq 1$  for each place  $p$ . The dual notion of marked graphs is the class of *P-nets* (also called *S-nets*), where  $|t^\bullet| = 1$  and  $|t^\bullet| = 1$  for each transition  $t$ .

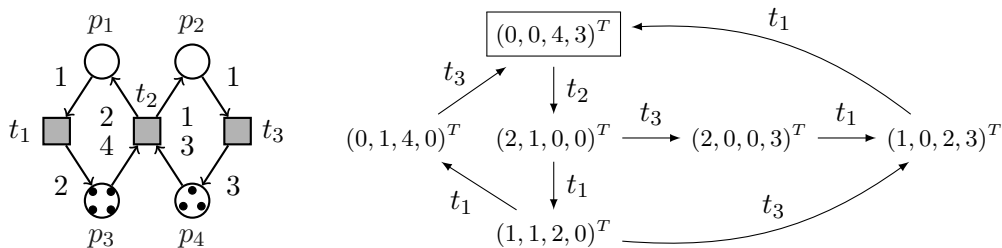


Figure 1: A WMG system  $\zeta$  and its reachability graph  $RG(\zeta)$  are pictured respectively on the left and on the right (markings are represented by vectors with indices corresponding successively to  $p_1, p_2, p_3$  and  $p_4$ ). The initial marking is boxed in  $RG(\zeta)$ .

**Lts and their relationship with Petri nets.** A *labelled transition system with initial state*, abbreviated *lts*, is a quadruple  $TS = (S, \rightarrow, T, \iota)$  where  $S$  is the set of *states*,  $T$  is the set of *labels*,  $\rightarrow \subseteq (S \times T \times S)$  is the *transition relation*, and  $\iota \in S$  is the *initial state*.

<sup>1</sup>In pure nets, i.e. nets where  $\bullet n \cap n^\bullet = \emptyset$  for each node, deleting frozen tokens from the initial marking preserves the set of feasible sequences, and simple conditions exist that compute potentially useful tokens [23, 24].

Two lts  $TS_1 = (S_1, \rightarrow_1, T, s_{01})$  and  $TS_2 = (S_2, \rightarrow_2, T, s_{02})$  are isomorphic if there is a bijection  $\beta: S_1 \rightarrow S_2$  with  $\beta(s_{01}) = s_{02}$  and  $(s, t, s') \in \rightarrow_1 \Leftrightarrow (\beta(s), t, \beta(s')) \in \rightarrow_2$ , for all  $s, s' \in S_1$ .

A label  $t$  is *enabled* at  $s \in S$  if  $\exists s' \in S: (s, t, s') \in \rightarrow$ , written  $s[t]$  or  $s[t]s'$ , in which case  $s'$  is *reachable* from  $s$  through the execution of  $t$ . We denote by  $s^\bullet$  the set  $\{s' \mid \exists t \in T, s[t]s'\}$ .

A label  $t$  is *backward enabled* at  $s$  if  $\exists s' \in S: (s', t, s) \in \rightarrow$ , written  $[t]s$  or  $s'[t]s$ . A (*firing*) *sequence*  $\sigma$  of length  $n \geq 0$  on the set of labels  $T$ , denoted by  $\sigma = t_1 \dots t_n$  with  $t_1, \dots, t_n \in T$ , is *enabled* at some state  $s_0$  if the successive states obtained,  $s_0[t_1]s_1 \dots [t_n]s_n$ , satisfy  $s_{k-1}[t_k]s_k$ ,  $\forall k \in \{1, \dots, n\}$ : we note  $s_0[\sigma]s_n$ . Similarly, other notions and notations, related to sequences and reachability in Petri nets, extend readily to labelled transition systems by replacing markings with states.

The reachability graph  $RG(\zeta)$  of a system  $\zeta = (N, M_0)$  can be represented by the labelled transition system  $TS = (S, \rightarrow, T, \iota)$  if an isomorphism  $\gamma: S \rightarrow [M_0]$  exists such that  $\gamma(\iota) = M_0$  and  $(s, t, s') \in \rightarrow \Leftrightarrow \gamma(s)[t]\gamma(s')$  for all  $s, s' \in S$ . If an lts  $TS$  is isomorphic to the reachability graph of a Petri net system  $\zeta = (N, M_0)$ , we say that  $\zeta$  *solves*  $TS$ , and that it *WMG-solves*  $TS$  if  $N$  is a WMG.

These notions are illustrated on the lts on the right of Figure 2, which is isomorphic to the reachability graph of the WMG in Figure 1; it is thus WMG-solvable.

**Vectors, semiflows and cycles.** The *support* of a vector is the set of the indices of its non-null components. Consider any net  $N = (P, T, W)$  with its incidence matrix  $C$ . A *T-vector* is an element of  $\mathbb{N}^T$ ; it is called *prime* if the greatest common divisor of its components is one (i.e., its components do not have a common non-unit factor). A *T-semiflow*  $\nu$  of the net is a non-null T-vector whose components are only non-negative integers (i.e.,  $\nu \succeq \mathbb{0}$ ) and such that  $C \cdot \nu = \mathbb{0}$ . A T-semiflow is called *minimal* when it is prime and its support is not a proper superset of the support of any other T-semiflow [26].

The *Parikh vector*  $\mathbf{P}(\sigma)$  of a finite sequence  $\sigma$  of transitions is a T-vector counting the number of occurrences of each transition in  $\sigma$ , and the *support* of  $\sigma$  is the support of its Parikh vector, i.e.,  $\text{supp}(\sigma) = \text{supp}(\mathbf{P}(\sigma)) = \{t \in T \mid \mathbf{P}(\sigma)(t) > 0\}$ . A (non-empty) cycle around a marking  $M$  is a non-empty sequence  $\sigma$  such that  $M[\sigma]M$ ; the Parikh vector of a non-empty cycle is a T-semiflow and a non-empty cycle is called *prime* if its Parikh vector is prime.

**Further notions.** Consider a lts  $TS = (S, \rightarrow, T, \iota)$ . For all states  $s, s' \in S$ , a sequence  $s[\sigma]s'$  is called a *cycle*, or more precisely a *cycle at (or around) state*  $s$ , if  $s = s'$ . A non-empty cycle  $s[\sigma]s$  is called *small* if there is no non-empty cycle  $s'[\sigma']s'$  in  $TS$  with  $\mathbf{P}(\sigma') \preceq \mathbf{P}(\sigma)$ . A *two-way uniform chain* of  $TS$  is a couple  $(\{s_i \in S \mid i \in \mathbb{Z}, \forall i, j \in \mathbb{Z} : i \neq j \Rightarrow s_i \neq s_j\}, \sigma \in T^+)$  such that  $\forall i \in \mathbb{Z}, s_i[\sigma]s_{i+1}$ , where  $T^+$  is the set of non-empty sequences on  $T$ .

In Figure 2, a two-way uniform chain is depicted on the left; on the right, the lts is finite, hence has no two-way uniform chain. The lts  $TS$  is:

- *totally reachable* if  $S = [\iota]$ ;
- *reversible* if  $\iota \in [s]$  for each state  $s \in [\iota]$ , meaning the strong connectedness of this lts when it is totally reachable;
- *weakly periodic* if for each couple  $(\{s_i \in S \mid i \in \mathbb{N}\}, \sigma \in T^+)$  such that  $\forall i \in \mathbb{N} s_i[\sigma]s_{i+1}$  ( $\sigma$  is thus a non-empty sequence of labels), either  $s_i = s_j \forall i, j \in \mathbb{N}$ , or  $i \neq j \Rightarrow s_i \neq s_j \forall i, j \in \mathbb{N}$ ;

- *strongly cycle consistent* if for each sequence  $s[\alpha]s'$ , the existence of cycles  $s_1[\beta_1]s_1$ ,  $s_2[\beta_2]s_2, \dots, s_n[\beta_n]s_n$  and of numbers  $k_1, k_2, \dots, k_n \in \mathbb{Q}$  such that  $\mathbf{P}(\alpha) = \sum_{i=1}^n k_i \cdot \mathbf{P}(\beta_i)$  implies that  $s = s'$ ;
- *deterministic* if, for all states  $s, s', s'' \in S$  and labels  $t, t' \in T$  such that  $s[t]s' \wedge s[t]s''$ , necessarily  $s' = s''$ ; it is *fully deterministic* if for all sequences  $\sigma$  and  $\sigma'$  such that  $\mathbf{P}(\sigma) = \mathbf{P}(\sigma')$ , we have, for all states  $s, s', s'' \in S$ :  $s[\sigma]s' \wedge s[\sigma']s'' \Rightarrow s' = s''$ ;
- *backward deterministic* if, for all states  $s, s', s'' \in S$  and labels  $t, t' \in T$  such that  $s'[t]s \wedge s''[t]s$ , necessarily  $s' = s''$ ; it is *fully backward deterministic* if, for all sequences  $\sigma$  and  $\sigma'$  such that  $\mathbf{P}(\sigma) = \mathbf{P}(\sigma')$ , we have, for all states  $s, s', s'' \in S$ :  $s'[\sigma]s \wedge s'[\sigma']s \Rightarrow s' = s''$ ;
- *persistent* if for all states  $s, s', s'' \in S$  and labels  $t', t'' \in T$  such that  $s[t']s'$  and  $s[t'']s''$  with  $t' \neq t''$ , there exists a state  $s''' \in S$  such that  $s'[t'' ]s'''$  and  $s''[t']s'''$ ; it is *backward persistent* if for all states  $s, s', s'' \in S$  and labels  $t', t'' \in T$  such that  $s'[t']s$  and  $s''[t'']s$  with  $t' \neq t''$ , there exists a state  $s''' \in S$  such that  $s'''[t'']s'$  and  $s'''[t']s''$ .

Figure 2 illustrates some of these notions. All notions defined for labelled transition systems apply to Petri nets through their reachability graphs. For example, a Petri net is reversible if its reachability graph is isomorphic to a reversible lts, meaning that the initial marking is reachable from every reachable marking.

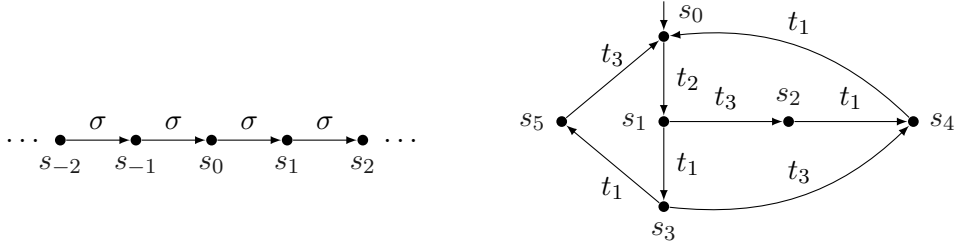


Figure 2: On the left, a two-way uniform chain based on  $\sigma$ . On the right, a labelled transition system with states  $\{s_0, s_1, s_2, s_3, s_4, s_5\}$ , labels  $\{t_1, t_2, t_3\}$  and initial state  $\iota = s_0$ . It is isomorphic to the reachability graph of Figure 1. The label  $t_2$  is enabled at  $s_0$  and  $t_3$  is backward enabled at  $s_0$ . The state  $s_1$  is reachable from  $s_0$  through the execution of  $t_2$ . Denote by  $\sigma$  the sequence  $t_2t_3t_1t_1$ . Then, the Parikh vector of  $\sigma$  is  $\mathbf{P}(\sigma) = (2, 1, 1)$  and its support is  $\text{supp}(\sigma) = \{t_1, t_2, t_3\}$ . Since  $s_0[\sigma]s_0$ ,  $\sigma$  is a cycle around state  $s_0$ . This lts is totally reachable, weakly periodic, fully deterministic and fully backward deterministic, strongly cycle consistent, persistent, backward persistent and reversible.

The following proposition recalls properties satisfied by every Petri net system (see e.g. [27]).

### Proposition 2.1. (Classical properties of Petri nets)

If  $\zeta = (N, M_0)$ , where  $N = (P, T, W)$ , is a Petri net system, then  $RG(\zeta)$  is totally reachable, weakly periodic, fully deterministic, fully backward deterministic, and strongly cycle consistent. Moreover it has no two-way uniform chain over the set  $S = \mathbb{N}^P$  of all the possible markings for  $N$ , meaning that no couple  $(\{M_i \in \mathbb{N}^P \mid i \in \mathbb{Z} \setminus \{0\}, \forall i, j \in \mathbb{Z} : i \neq j \Rightarrow s_i \neq s_j\}, \sigma \in T^+)$  exists such that  $\forall i \in \mathbb{Z}, M_i[\sigma]M_{i+1}$ .

### 3. Properties of WMGs and Larger Persistent Classes

In this section, we investigate the structure and behaviour of WMGs. For that purpose, we first recall notions and results relevant to persistent systems in Subsection 3.1. Then, in Subsection 3.2, for the class of WMGs, we show a property of the sequences sharing the same starting state and the same ending state, and we prove backward persistence. Finally, in Subsection 3.3, we propose conditions for the existence of feasible sequences corresponding to a given T-vector in WMGs.

#### 3.1. Previous Results and Notions Related to Persistence

In addition to the general properties of Petri nets mentioned in Proposition 2.1, we recall results and notions useful to the study of persistent systems.

The following result is dedicated to WMGs and extracted from [10, 26].

**Proposition 3.1. (Minimal T-semiflow and cycles in WMGs)**

Let  $N$  be a connected WMG net. If  $N$  has some T-semiflow, then it has a unique minimal (hence prime) one, that we shall denote by  $\pi$ . The support of  $\pi$  is the whole set of transitions ( $\text{supp}(\pi) = T$ ) and for any other T-semiflow  $\nu$  we have  $\nu = k \cdot \pi$  for some integer  $k > 0$ . Moreover, for any marking  $M_0$ , if the reachability graph  $RG(\zeta)$  of the WMG net system  $\zeta = (N, M_0)$  contains some non-empty cycle, then the Parikh vector of each small cycle of  $RG(\zeta)$  equals  $\pi$ .

The following notion of residues is useful for the study of persistent systems.

**Definition 3.2. (Residues)**

Let  $T$  be a set of labels and  $\tau, \sigma \in T^*$  two (possibly empty) sequences over this set. The (*left*) residue of  $\tau$  with respect to  $\sigma$ , denoted by  $\tau \overset{\bullet}{\ominus} \sigma$ , arises from cancelling successively in  $\tau$  the leftmost occurrences of all symbols from  $\sigma$ , read from left to right. Inductively:  $\tau \overset{\bullet}{\ominus} \varepsilon = \tau$ ;  $\tau \overset{\bullet}{\ominus} t = \tau$  if  $t \notin \text{supp}(\tau)$ ;  $\tau \overset{\bullet}{\ominus} t$  is the sequence obtained by erasing the leftmost  $t$  in  $\tau$  if  $t \in \text{supp}(\tau)$ ; and  $\tau \overset{\bullet}{\ominus} (t\sigma) = (\tau \overset{\bullet}{\ominus} t) \overset{\bullet}{\ominus} \sigma$ .

For example,  $acbcacbc \overset{\bullet}{\ominus} abcb = cacc$  and  $abcb \overset{\bullet}{\ominus} acbcacbc = b$ .

We deduce the following properties of residues, stemming from the definitions of residues and Parikh vectors.

**Corollary 3.3. (Explicit formula for the Parikh vector of a residue)**

For any two sequences  $\tau$  and  $\sigma$  and transition  $t \in T$ ,  $\mathbf{P}(\sigma \overset{\bullet}{\ominus} \tau)(t) = \max(0, \mathbf{P}(\sigma)(t) - \mathbf{P}(\tau)(t))$ .

**Corollary 3.4. (Disjoint support of residues)**

For any two sequences  $\tau$  and  $\sigma$ , the residues  $\delta_1 = \tau \overset{\bullet}{\ominus} \sigma$  and  $\delta_2 = \sigma \overset{\bullet}{\ominus} \tau$  have disjoint supports:  $\text{supp}(\delta_1) \cap \text{supp}(\delta_2) = \emptyset$ . Consequently,  $\delta_1 \overset{\bullet}{\ominus} \delta_2 = \delta_1$  and  $\delta_2 \overset{\bullet}{\ominus} \delta_1 = \delta_2$ .

**Proof:**

For any label  $t$ ,  $\mathbf{P}(\tau)(t) = \mathbf{P}(\sigma)(t) \Rightarrow \mathbf{P}(\delta_1)(t) = \mathbf{P}(\delta_2)(t) = 0$ ,  $\mathbf{P}(\tau)(t) > \mathbf{P}(\sigma)(t) \Rightarrow \mathbf{P}(\delta_2)(t) = 0$  and  $\mathbf{P}(\tau)(t) < \mathbf{P}(\sigma)(t) \Rightarrow \mathbf{P}(\delta_1)(t) = 0$ . In any case,  $t \notin \text{supp}(\delta_1) \cap \text{supp}(\delta_2)$ .  $\square$



**Corollary 3.5. (Equivalent extensions)**

For any two sequences  $\tau$  and  $\sigma$ ,  $\mathbf{P}(\sigma(\tau \bullet \sigma)) = \mathbf{P}(\tau(\sigma \bullet \tau))$  and, for each transition  $t \in T$ , if we denote by  $CP$  this common Parikh vector, we have  $CP(t) = \max(\mathbf{P}(\sigma)(t), \mathbf{P}(\tau)(t))$ .

**Proof:**

For any  $t \in T$ ,  $\mathbf{P}(\sigma(\tau \bullet \sigma))(t)$  is  $\mathbf{P}(\sigma)(t)$  if there are more  $t$ 's in  $\sigma$  than in  $\tau$ . On the contrary, if there are more  $t$ 's in  $\tau$  than in  $\sigma$ ,  $\mathbf{P}(\sigma(\tau \bullet \sigma))(t)$  is the number of  $t$ 's in  $\sigma$  plus the number of additional  $t$ 's in  $\tau$ . Hence the claimed property.  $\square$

**Corollary 3.6. (Null residues)**

For any two sequences  $\tau$  and  $\sigma$ ,  $\mathbf{P}(\sigma) = \mathbf{P}(\tau) \Leftrightarrow \mathbf{P}(\sigma \bullet \tau) = \mathbb{0} = \mathbf{P}(\tau \bullet \sigma)$ .

**Proof:**

For any  $t \in T$ , if there are more  $t$ 's in  $\sigma$  than in  $\tau$ ,  $\mathbf{P}(\sigma)(t) > \mathbf{P}(\tau)(t)$  and  $\mathbf{P}(\sigma \bullet \tau)(t) > 0$ , and symmetrically if there are fewer  $t$ 's in  $\sigma$  than in  $\tau$ . The property results.  $\square$

In the literature, the following result has proved to be a cornerstone in the study of persistence.

**Theorem 3.7. (Keller's theorem [29])**

Let  $(S, \rightarrow, T, \iota)$  be a deterministic, persistent lts. Let  $\tau$  and  $\sigma$  be two label sequences enabled at some state  $s$ . Then  $\tau(\sigma \bullet \tau)$  and  $\sigma(\tau \bullet \sigma)$  are both enabled at  $s$  and lead to the same state.

Applying Theorem 3.7, we obtain the following result directly.

**Proposition 3.8. (Persistence and determinism)**

Let  $TS$  be a persistent lts. If  $TS$  is also deterministic, then it is fully deterministic.

## 3.2. Equivalent Sequences and Backward Persistence

In the following, we provide new properties on the reachability graph of WMGs. Since non-connected nets can be studied by analysing each connected component separately, we restrict our attention to connected nets.

For the class of WMGs, we first provide in Lemma 3.10 a property of the sequences starting from a same state  $s$  and leading to the same state  $s'$ . Then, we prove the backward persistence of WMGs in Theorem 3.11. To achieve it, we need to define the reverse of a net and of a firing sequence.

**Definition 3.9. (Reverse nets and sequences)**

The *reverse* of a net  $N$ , denoted by  $-N$ , is obtained from  $N$  by reversing all the arcs while keeping the weights. We denote by  $\sigma^{-1}$  the sequence  $\sigma$  followed in reverse order. For example, if  $\sigma = t_1 t_2 t_3$ , then  $\sigma^{-1} = t_3 t_2 t_1$ .

The class of WMGs is closed under reverse, contrarily to the class of CF nets.

Lemma 3.10 highlights strong similarities in the reachability graph between two sequences sharing the same starting state and the same destination state. The proof makes use of reverse sequences feasible in reverse WMGs.

**Lemma 3.10. (Equivalent sequences in WMGs)**

Let  $N$  be a connected WMG. Assume the existence of markings  $M, M_1$  and sequences  $\sigma, \sigma'$  such that  $M[\sigma]M_1$  and  $M[\sigma']M_1$ . If  $N$  has no T-semiflow, then  $\mathbf{P}(\sigma) = \mathbf{P}(\sigma')$ . Otherwise, either  $\mathbf{P}(\sigma) = \mathbf{P}(\sigma')$ , or there exists an integer  $k > 0$  such that  $\mathbf{P}(\sigma) = \mathbf{P}(\sigma') + k \cdot \pi$  or  $\mathbf{P}(\sigma) + k \cdot \pi = \mathbf{P}(\sigma')$ , where  $\pi$  is the unique minimal T-semiflow of  $N$ .

**Proof:**

Let us assume that  $\mathbf{P}(\sigma) \neq \mathbf{P}(\sigma')$ . We show in the following that  $N$  has necessarily a T-semiflow in this case, proving the first claim by contraposition.

Since  $N$  is a WMG, it is persistent. Defining  $\tau = \sigma \bullet \sigma'$  and  $\tau' = \sigma' \bullet \sigma$ , applying Keller's theorem (Theorem 3.7), we have for some marking  $M_2$  that  $M_1[\tau]M_2$  and  $M_1[\tau']M_2$ . By Corollary 3.4,  $\tau$  and  $\tau'$  have disjoint supports,  $\tau \bullet \tau' = \tau$  and  $\tau' \bullet \tau = \tau'$ . Thus, applying Keller's theorem, a marking  $M_3$  is reached from  $M_2$  by firing  $\tau$  or  $\tau'$ . Iterating this process up to any positive integer  $i$ , some marking  $M_{i+1}$  is reached from  $M_i$  with  $M_i[\tau]M_{i+1}$  and  $M_i[\tau']M_{i+1}$ .

Now, in the reverse net  $-N$ , which is also a WMG, since  $M_2[(\tau)^{-1}]M_1$  and  $M_2[(\tau')^{-1}]M_1$ , still with disjoint supports, we can construct markings  $M_0, M_{-1} \dots$  such that  $\forall i \in \mathbb{Z}, M_i[(\tau)^{-1}]M_{i-1}$  and  $M_i[(\tau')^{-1}]M_{i-1}$ , i.e., also  $M_{i-1}[\tau]M_i$  and  $M_{i-1}[\tau']M_i$ . If all  $M_i$ 's are (pairwisely) different, this leads to a two-way uniform chain for the system  $(N, M_1)$ , contradicting Proposition 2.1. Consequently, for some  $i, j \in \mathbb{Z}$  with  $i \neq j$ , we have  $M_i = M_j$ , and since  $\sigma, \sigma'$  are different, they are not both empty and  $\tau, \tau'$  cannot be both empty. Thus,  $N$  has a T-semiflow, proving the first claim of the lemma.

For the second claim, either  $\mathbf{P}(\sigma) = \mathbf{P}(\sigma')$  or  $\mathbf{P}(\sigma) \neq \mathbf{P}(\sigma')$ . Consider the latter case: from the first part of the proof, taking the same notation, there is a positive integer  $n$  such that  $\tau^n$  and  $\tau'^n$  are cycles appearing in the reachability graph of the system  $(N, M)$ . Since the supports of  $\tau$  and  $\tau'$  are not both empty, Proposition 3.1 applies: there is a unique minimal T-semiflow  $\pi$ , whose support is  $T$ , and integers  $k, k' \geq 0$  exist such that  $\mathbf{P}(\tau^n) = k \cdot \pi$  and  $\mathbf{P}(\tau'^n) = k' \cdot \pi$ , where  $k > 0$  or  $k' > 0$ . Since the supports of  $\tau$  and  $\tau'$  are disjoint and the support of any cycle is  $T$ , then either  $k' = 0, \tau' = \epsilon, \tau$  is a cycle and  $\mathbf{P}(\sigma) \succeq \mathbf{P}(\sigma')$ , or  $k = 0, \tau = \epsilon, \tau'$  is a cycle and  $\mathbf{P}(\sigma') \succeq \mathbf{P}(\sigma)$ . Thus, either  $\mathbf{P}(\sigma) = \mathbf{P}(\sigma') + \mathbf{P}(\tau) = \mathbf{P}(\sigma') + q \cdot \pi$  for an integer  $q > 0$  or  $\mathbf{P}(\sigma') = \mathbf{P}(\sigma) + \mathbf{P}(\tau') = \mathbf{P}(\sigma) + q \cdot \pi$  for an integer  $q > 0$ . Hence the claim.  $\square$

In [10], in the proof of Theorem 4.8, it is mentioned that each WMG is backward persistent without a proof, based on the fact that the reverse of a WMG is still a WMG, hence a persistent net. However, this property needs to be proved carefully: since  $M_1[a]M$  and  $M_2[b]M$ , Keller's theorem implies the existence of a marking  $M'$  reachable from  $(-N, M_1)$  and  $(-N, M_2)$ , such that  $M'[a]M_2$  and  $M'[b]M_1$  in the original system; however, the reachability of  $M'$  in the original system, under the assumption of reachability for  $M_1$  and  $M_2$ , is not obvious. In the following, we show it is indeed the case.

**Theorem 3.11. (Backward persistence of WMGs)**

In a connected WMG system  $\zeta = (N, M_0)$ , let us assume that markings  $M_1, M_2, M$  are reachable and that, for two different labels  $a$  and  $b$ ,  $M_1[a]M$  and  $M_2[b]M$ . Then, a marking  $M'$  is reachable in  $\zeta$  such that  $M'[a]M_2$  and  $M'[b]M_1$ .

**Proof:**

Let us write  $N = (P, T, W)$  and introduce two sequences  $\sigma_1$  and  $\sigma_2$  enabled in  $\zeta$  such that  $M_0[\sigma_1]M_1$  and  $M_0[\sigma_2]M_2$ . From the previous remarks, we know that  $M'$  is reachable in the reverse system  $(-N, M_1)$ , hence belongs to  $\mathbb{N}^P$ . It remains to show that  $M'$  is reachable in  $\zeta$ .

From Lemma 3.10, either  $\mathbf{P}(\sigma_1 a) = \mathbf{P}(\sigma_2 b)$  or, without loss of generality,  $\mathbf{P}(\sigma_1 a) = \mathbf{P}(\sigma_2 b) + k \cdot \pi$ , where  $\pi$  is the unique minimal T-semiflow of  $N$ , with support  $T$ . Then, in either case,  $b$  occurs at least once in  $\sigma_1$ . For the reverse net  $-N$ , we have  $M_1[\sigma_1^{-1}]M_0$ , and from Keller's theorem, we have  $M_1[b]M'$  and  $M_2[a]M'$ ; we also have  $M_0[b \bullet \sigma_1^{-1}]M''$  and  $M'[\sigma_1^{-1} \bullet b]M''$ . Since  $b$  occurs in  $\sigma_1$ , hence also in  $\sigma_1^{-1}$ ,  $b \bullet \sigma_1^{-1} = \epsilon$  and  $M'' = M_0$ . Going back to  $\zeta$ , we deduce that  $M_0[(\sigma_1^{-1} \bullet b)^{-1}]M'$ , thus  $M'$  is reachable in  $\zeta$ .  $\square$

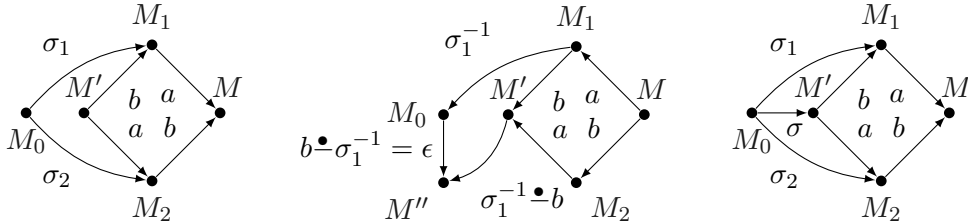


Figure 3: Illustration of the proof of Theorem 3.11: the initial assumptions are depicted on the left, the sequences in the reverse system  $(-N, M)$  are depicted in the middle, where the sequence leading to  $M''$  from  $M_0$  equals  $\epsilon$ , implying that  $M'' = M_0$ . In the original system, we deduce the reachability of  $M'$  from  $M_0$  on the right, with  $\sigma = (\sigma_1^{-1} \bullet b)^{-1}$ .

In order to show the sensibility of Theorem 3.11 to the hypotheses, Figure 4 shows it is not true for FA systems (hence also for CF systems). However, the only difference in the definition of FA and WMG systems is that the constraints on the cardinality of predecessors concerns transitions in the first case and places in the second one.

**3.3. Fireability of T-vectors in WMGs**

In this subsection, we develop conditions for the existence of enabled sequences corresponding to given Parikh vectors. For that purpose, we borrow some vocabulary from [10, 5] as follows: for a net with incidence matrix  $C$ , we say that a marking  $M$  is *potentially reachable* from a marking  $M_0$  if a T-vector  $\nu$  exists such that  $M = M_0 + C \cdot \nu$ . If, additionally, a sequence  $\sigma$  is enabled in  $(N, M_0)$  such that  $\mathbf{P}(\sigma) = \nu$ , we also say that  $\nu$  is enabled (or fireable, or feasible, or realisable) at  $M_0$ . This is inspired by the classical state equation, stating that if  $M_0[\sigma]M$  in a Petri net, then  $M = M_0 + C \cdot \mathbf{P}(\sigma)$ .

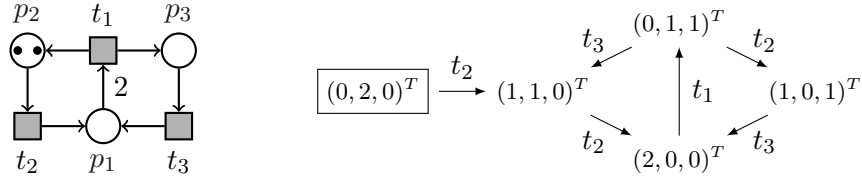


Figure 4: A Fork-Attribution (FA) system on the left and its reachability graph on the right, where the initial marking is boxed. The FA system is not backward persistent, since the marking  $(1, 1, 0)^T$  can be reached from two predecessors by firing  $t_2$  and  $t_3$  respectively from the initial marking and from  $(0, 1, 1)^T$ , but the initial marking has no predecessor.

### Lemma 3.12. (Enabled T-vectors in WMGs)

Let  $N = (P, T, W)$  be a WMG with incidence matrix  $C$ . Let  $M$  be a marking and  $\nu \in \mathbb{N}^T$  be a T-vector such that  $M + C \cdot \nu \geq 0$ . Let  $T_1$  be the support of  $\nu$ ,  $P_1 = \bullet T_1 \cap T_1^\bullet$ ,  $\sigma'$  a transition sequence such that  $\nu \leq \mathbf{P}(\sigma')$ , and  $M'$  be a marking such that  $\forall p \in P_1 : M'(p) = M(p)$ . Then, if  $M'[\sigma']$ , there is a firing sequence  $M[\sigma]$  such that  $\mathbf{P}(\sigma) = \nu$ .

#### Proof:

By induction on the size of  $\nu$ . If  $\nu = \mathbb{0}$ , the property is clearly true. Otherwise, let  $t$  be the first transition of  $T_1$  occurring in  $\sigma'$ , i.e.,  $\sigma' = \sigma'_1 t \sigma'_2$  with  $\nu(t_i) = 0$  for each  $t_i$  in  $\sigma'_1$ .

Assume that  $\neg M[t]$ , then for some  $p \in \bullet t$ ,  $M(p) < W(p, t)$ . Since  $M + C \cdot \nu \geq 0$ , there is  $t' \in \bullet p$ ,  $t' \neq t$ , such that  $t' \in T_1$ , and  $t'$  is unique in  $\bullet p$  since  $N$  is a WMG. This contradicts the fact that  $M'[\sigma'_1 t]$  since  $t'$  does not occur in  $\sigma'_1$ . Hence, we assume that  $M[t]M_1$  and  $M'[\sigma'_1]M''[t]M'_1[\sigma'_2]$ .

Since the net is a WMG, the only transitions able to modify the places in  $P_1$  are in  $T_1$ . Thus, we have  $M(p) = M'(p) = M''(p)$  and  $M_1(p) = M'_1(p)$  for each  $p \in P_1$  (no transition of  $T_1$  belongs to  $\sigma'_1$ ). Let us denote by  $\delta_t$  the T-vector with value 1 for  $t$ , 0 elsewhere.

Hence, the induction hypothesis applies to  $\nu - \delta_t \leq \mathbf{P}(\sigma'_2)$  from the markings  $M_1$  and  $M'_1$ , and there is a firing sequence  $M_1[\sigma_1]$  with  $\mathbf{P}(\sigma_1) = \nu - \delta_t$ . Thus, the sequence  $\sigma = t \sigma_1$  with Parikh vector  $\nu$  is enabled at  $M$ . The lemma results.  $\square$

Instantiating Lemma 3.12 with  $M = M'$ , we deduce the following corollary.

### Corollary 3.13. (Potential reachability in WMGs)

Let  $N = (P, T, W)$  be a WMG with incidence matrix  $C$ . Let  $M$  be any marking and  $\nu \in \mathbb{N}^T$  be a T-vector such that  $M' = M + C \cdot \nu \geq 0$ . Let  $\sigma$  be a transition sequence such that  $\nu \leq \mathbf{P}(\sigma)$ . Then, if  $M[\sigma]$ , there is a firing sequence  $M[\sigma']M'$  such that  $\mathbf{P}(\sigma') = \nu$ .

Lemma 3.12 and Corollary 3.13 are not valid in the class of FA systems. Indeed, denoting by  $C$  the incidence matrix of the system in Figure 4, by  $M_0$  its initial marking  $(0, 2, 0)^T$  and by  $\nu$  the T-vector  $(1, 1, 1)^T$ , we have:  $M_0 = M_0 + C \cdot \nu$ , and the sequence  $\sigma = t_2 t_2 t_1 t_3$  is enabled at  $M_0$ , with  $\mathbf{P}(\sigma) \geq \nu$ . However, there is no initially feasible sequence whose Parikh vector equals  $\nu$ .

We present following the notion of a *maximal execution vector* in order to obtain Theorem 3.15 on potentially reachable markings below.

**Definition 3.14. (Maximal execution vector in WMGs)**

Let  $\zeta$  be a WMG system whose set of transitions is  $T$ . We denote by  $maxex_\zeta : T \rightarrow \mathbb{N} \cup \{\infty\}$  the extended T-vector satisfying:  $\forall t \in T, maxex_\zeta(t)$  is the maximal number of times  $t$  may be executed in firing sequences of  $\zeta$ , allowing the case  $maxex_\zeta(t) = \infty$ .

**Theorem 3.15. (Potential reachability in WMGs, revised)**

Let  $\zeta = (N, M_0)$  be a WMG with incidence matrix  $C$ . Let  $\nu \in \mathbb{N}^T$  be a T-vector such that  $M = M_0 + C \cdot \nu \geq 0$ . Let  $maxex_\zeta$  be the maximal execution vector of  $\zeta$ . Then, there exists a firing sequence  $M_0[\sigma]M$  with  $\mathbf{P}(\sigma) = \nu$  if and only if  $\nu \leq maxex_\zeta$ .

**Proof:**

Suppose that  $M_0[\sigma]M$  with  $\mathbf{P}(\sigma) = \nu$ . Since  $\sigma$  can be fired at  $M_0$ , we deduce, from the definition of  $maxex_\zeta$ , that  $\forall t \in T, maxex_\zeta(t) \geq \mathbf{P}(\sigma)(t)$ , thus  $\nu \leq maxex_\zeta$ .

Conversely, suppose that  $\nu \leq maxex_\zeta$ . Then, for each  $t \in T$ , since  $\nu(t) \leq maxex_\zeta(t)$ , there is a finite firing sequence  $M_0[\sigma_t]$  such that  $\nu(t) \leq \mathbf{P}(\sigma_t)(t)$ . By persistence and Keller's theorem (applied  $|T| - 1$  times), there is a finite firing sequence  $M_0[\sigma']$  such that  $\forall t \in T : \mathbf{P}(\sigma_t)(t) \leq \mathbf{P}(\sigma')(t)$ , hence  $\nu \leq \mathbf{P}(\sigma')$  and Corollary 3.13 applies.  $\square$

In this section, we delineated several properties on the reachability graph of WMGs. In the next section, we exploit some of these conditions, notably persistence and backward persistence, to synthesise a WMG from a given lts, when possible.

## 4. Synthesis of Connected, Bounded, Weakly Live WMGs

In the domain of Petri net synthesis from labelled transition systems, the aim is to build a Petri net system whose reachability graph is isomorphic to a given lts, when it exists. Usually, one has to check first some necessary structural properties of the lts, in a pre-synthesis phase. In some rare cases, such conditions have been proven sufficient for ensuring the existence of a solution (sometimes a unique minimal one) in the class considered and for driving the synthesis process [19, 20]. However, in most cases, the known synthesis methods need a combination of such necessary structural conditions with computational checks and constructions [30, 31, 27, 32, 14, 15].

In this section, we focus on finite, totally reachable and *weakly live* lts, the latter property meaning that each label of  $T$  occurs at least once in the lts (if this is not the case, one may simply drop the useless transitions from  $T$ ). We build a procedure synthesising a connected WMG solving such lts when possible.

First, in Subsection 4.1, we highlight necessary structural conditions of WMG-solvability, notably persistence, backward persistence and the existence of specific cycles. We also build a counter-example showing that these conditions, when satisfied, are not sufficient to ensure WMG-solvability.

Section 4.2 recalls the bases of Petri net synthesis, i.e., the notions of regions and separation problems, and specialises them to WMG nets.

Then, in Subsection 4.3, we highlight constraints induced by each place and we delineate two subsets of the lts states that are particularly relevant to WMG-synthesis. By focusing the analysis on these states, the number of checking steps is potentially reduced.

Finally, in Subsections 4.4 and 4.5, we define systems of constraints for two kinds of lts shapes: the cyclic case, i.e., when the lts is strongly connected (hence reversible), and the acyclic case, i.e., when the lts does not contain any cycle. We show these two cases contain all the lts that are solvable by a connected, bounded and weakly live WMG. Also, the number of constraints is reduced by checking only the relevant states defined in Subsection 4.3. When these systems have a solution, we obtain a WMG solving the lts. To extend this method to all the WMG-solvable lts, the decomposition technique developed in [33, 34, 35] to factorise a lts into *prime factors*<sup>2</sup> can finally be exploited.

#### 4.1. Necessary Conditions for Solvability with Connected WMGs

For a synthesis into a connected WMG to succeed, the given lts must satisfy the conditions of Proposition 2.1, the properties described in Proposition 3.1, as well as persistence and backward persistence, as proved in Theorem 3.11. The boundedness of the WMG obtained stems from the finiteness of the lts. We capture part of these conditions with the following notation **b** and **c** and explain the relationship between the existence of a cycle in the lts and property **c**.

**Properties b and c.** For any lts  $TS = (S, \rightarrow, T, \iota)$ , we denote by:

- **b** (for basic) the property:  $TS$  is finite, weakly periodic, deterministic and backward deterministic, persistent and backward persistent, totally reachable;
- **c** (for cyclic) the property:  $TS$  is strongly connected, all its small cycles have the same prime Parikh vector  $\pi$  with support  $T$ , and  $\mathbf{P}(\alpha)$  is a multiple of  $\pi$  for each cycle  $\alpha$ .

Let us consider the case in which the finite lts contains a cycle. Then, the (finite) reachability graph of any connected and bounded WMG solving this lts contains a cycle. Thus, from Proposition 3.1, the cycle contains all transitions. By Corollary 4 in [26], the system is live, and by backward persistence, it is also reversible, implying the strong connectedness of the reachability graph. Consequently, we have to consider only two cases: the given lts is either acyclic or is strongly connected, the second case being considered in property **c**.

Without loss of generality, we assume in the sequel that the considered lts are weakly live. The following lemma presents relationships between properties relevant to the synthesis.

##### Lemma 4.1. (Determinism, reversibility, cycle consistence)

Let us consider a weakly live lts  $TS = (S, \rightarrow, T, \iota)$ .

- 1) If  $TS$  satisfies **b**, it also satisfies the full determinism and full backward determinism.
- 2) If  $TS$  satisfies **b** and is acyclic, all the sequences between any two states have the same Parikh vector.
- 3) If  $TS$  satisfies **b** and contains a small prime cycle with support  $T$  and Parikh vector  $\pi$ , then  $TS$  satisfies property **c**, there is a small prime cycle around each state, each arc belongs to a small prime cycle,  $TS$  satisfies the strong cycle consistence; also, for any two states  $s_1$  and  $s_2$ , there is a sequence from  $s_1$  to  $s_2$  whose Parikh vector  $\delta$  is not greater than or equal to  $\pi$ , and each other sequence  $\sigma$  from  $s_1$  to  $s_2$  satisfies  $\mathbf{P}(\sigma) = \mathbf{P}(\delta) + k \cdot \pi$  for a non-negative integer  $k$ .

<sup>2</sup>An lts is prime when it is not isomorphic to a disjoint product of non-trivial sub-lts (where disjunction applies to the label sets of the two lts) ; it then corresponds to a connected net, when solvable.

**Proof:**

1) Full determinism and full backward determinism arise directly from determinism and backward determinism and from persistence and backward persistence, with the aid of Proposition 3.8 applied to  $TS$  and to its reverse version.

2) If the lts is acyclic, satisfies **b** and, for some  $s \in S$  and  $s' \in [s]$ , we have  $s[\alpha]s'$  as well as  $s[\beta]s'$  with  $\mathbf{P}(\alpha) \neq \mathbf{P}(\beta)$ , so that  $\alpha \overset{\bullet}{\rightarrow} \beta$  or  $\beta \overset{\bullet}{\rightarrow} \alpha$  is non-empty (both of them may be non-empty). Then, as in the proof of Lemma 3.10, with the aid of Keller's theorem 3.7 and Corollary 3.4, we can build a uniform chain  $s'[\alpha \overset{\bullet}{\rightarrow} \beta]s_1[\alpha \overset{\bullet}{\rightarrow} \beta]s_2 \cdots$  and  $s'[\beta \overset{\bullet}{\rightarrow} \alpha]s_1[\beta \overset{\bullet}{\rightarrow} \alpha]s_2 \cdots$ . Since the lts is finite, there must exist positive integers  $i$  and  $j$  such that  $i < j$  and  $s_i = s_j$ , forming a non-empty cycle, hence a contradiction with the acyclicity.

3) In the rest of the proof, we suppose that the lts satisfies **b** and contains a small prime cycle  $\alpha$  around some state  $s \in S$  with support  $T$ . Determinism and persistence imply that cycles can be pushed forward Parikh-equivalently, i.e., if  $s[\alpha]s \wedge s[t]s'$ , then  $s'[\alpha']s'$  for some  $\alpha'$  with  $\mathbf{P}(\alpha') = \mathbf{P}(\alpha)$  (applying Keller's theorem). Symmetrically, backward determinism and backward persistence imply that cycles can be pushed backward Parikh-equivalently.

Now, consider any non-empty cycle  $\beta$  around some state  $s'$  in  $TS$ . Both cycles  $\alpha$  and  $\beta$  can be pushed backward Parikh-equivalently to the initial state  $\iota$  (since  $s$  and  $s'$  are reachable from  $\iota$  by total reachability). Using Keller's theorem, both support-disjoint sequences  $\alpha \overset{\bullet}{\rightarrow} \beta$  and  $\beta \overset{\bullet}{\rightarrow} \alpha$  are feasible at  $\iota$  and lead to some marking  $s_0$ . Since  $(\alpha \overset{\bullet}{\rightarrow} \beta)^n$  and  $(\beta \overset{\bullet}{\rightarrow} \alpha)^n$  are feasible at  $\iota$  for every positive integer  $n$  while the lts is finite, there exists a positive integer  $m$  such that  $(\alpha \overset{\bullet}{\rightarrow} \beta)^m$  and  $(\beta \overset{\bullet}{\rightarrow} \alpha)^m$  are cycles. Since the lts is also weakly periodic, deterministic and backward deterministic, both  $\alpha \overset{\bullet}{\rightarrow} \beta$  and  $\beta \overset{\bullet}{\rightarrow} \alpha$  are cycles. Since  $\alpha, \beta \neq \epsilon$  and  $\text{supp}(\alpha) = T$ , we have  $\mathbf{P}(\alpha \overset{\bullet}{\rightarrow} \beta) \leq \mathbf{P}(\alpha)$ . Hence, if  $\alpha \overset{\bullet}{\rightarrow} \beta \neq \epsilon$ , it forms a smaller cycle, contradicting the fact that  $\alpha$  is already a small cycle. Thus, necessarily,  $\alpha \overset{\bullet}{\rightarrow} \beta = \epsilon$ , implying that  $\mathbf{P}(\beta) \geq \mathbf{P}(\alpha)$ . Suppose that  $\mathbf{P}(\beta)$  is not a multiple of  $\mathbf{P}(\alpha)$ . Denote by  $k$  the largest integer such that  $\mathbf{P}(\beta) \geq k \cdot \mathbf{P}(\alpha)$  and  $\beta' = \beta \overset{\bullet}{\rightarrow} \alpha^k \neq \epsilon$ . Necessarily,  $\mathbf{P}(\alpha) \not\geq \mathbf{P}(\beta')$  and  $\mathbf{P}(\beta') \not\geq \mathbf{P}(\alpha)$ , implying that  $\mathbf{P}(\alpha) \not\geq \mathbf{P}(\alpha \overset{\bullet}{\rightarrow} \beta') \geq 0$ , where  $\alpha \overset{\bullet}{\rightarrow} \beta'$  is a cycle, contradicting the fact that  $\alpha$  is a small cycle. We deduce that  $\mathbf{P}(\beta)$ , as well as each other Parikh vector of each non-empty cycle of the lts, is a multiple of  $\mathbf{P}(\alpha) = \pi$ .

Hence, from total reachability and persistence, there is a small prime cycle (with Parikh vector  $\pi$ ) around the initial state, as well as around any state.

Since there is a small cycle with support  $T$  around each state, by Keller's theorem each arc can be extended into a cycle:  $s[t]s'$  implies there is a sequence  $s'[\gamma]s$  with  $\mathbf{P}(t\gamma) = \pi$ . As a consequence,  $TS$  is reversible, thus strongly connected.

For any cycle  $s[\beta]s$ , from the above, we have that  $\mathbf{P}(\beta) = k \cdot \pi$  for some integer  $k \geq 0$ . Now, if a sequence  $s[\gamma]s'$  is such that  $k_1 \cdot \mathbf{P}(\gamma) = k_2 \cdot \pi$  for some positive integers  $k_1, k_2$ , since  $\pi$  is prime  $k_1$  must divide  $k_2$ ; let us denote by  $k'$  the integer  $k_2/k_1$ . We have  $\mathbf{P}(\gamma) = k' \cdot \pi$  so that by full determinism  $s = s'$ , hence the strong cycle consistence.

Consider a sequence  $s[\alpha]s'$  in  $TS$ . Suppose that  $\mathbf{P}(\alpha) \geq \pi$ . Since there is a small cycle  $\gamma$  with Parikh vector  $\pi$  around  $s$ , we build a shorter sequence by applying Keller's theorem as follows:  $\alpha \overset{\bullet}{\rightarrow} \gamma$  is fireable at  $s$  and leads to  $s'$ . We can build such shorter sequences until we get a sequence  $s[\alpha']s'$  with  $\mathbf{P}(\alpha') \not\geq \pi$  and  $\mathbf{P}(\alpha) = \mathbf{P}(\alpha') + k \cdot \pi$  for some integer  $k > 0$ . If we start from another sequence  $s[\beta]s'$ , we get similarly  $s[\beta']s'$  with  $\mathbf{P}(\beta') \not\geq \pi$ , and  $\mathbf{P}(\beta) = \mathbf{P}(\beta') + h \cdot \pi$  for a non-negative integer  $h$ . If  $\mathbf{P}(\beta') \neq \mathbf{P}(\alpha')$ , then we have two cases.

In the first case, one of them is greater than the other one, without loss of generality  $\mathbf{P}(\beta') \geq$

$\mathbf{P}(\alpha')$ , in which case, with Keller's theorem, we can construct from  $s'$  the cycle  $\beta' \bullet \alpha'$  with  $\mathbf{P}(\beta' \bullet \alpha') \not\geq \mathbf{P}(\pi)$  a contradiction with the fact that the Parikh vector of every cycle of  $TS$  is a multiple of  $\pi$  (with support  $T$ ).

In the second case,  $\beta' \bullet \alpha'$  and  $\alpha' \bullet \beta'$  are both non-empty with disjoint supports (see Corollary 3.4), and these supports are smaller than  $T$ . In this second case, we construct from  $s_1 = s'$  a chain  $s_1[\sigma]s_2[\sigma]s_3 \cdots$ , with  $\sigma = \beta' \bullet \alpha'$  as well as for  $\sigma = \alpha' \bullet \beta'$ . Since the lts is finite, we must have  $s_i = s_j$  for some  $i < j$ , hence a cycle with Parikh vector  $n \cdot \pi = (j - i) \cdot \mathbf{P}(\sigma)$  for some positive integer  $n$ , which is incompatible with the fact that the support of  $\pi$  is  $T$  and the support of  $\sigma$  does not contain all transitions.

Thus, we get a contradiction in both cases, implying that  $\mathbf{P}(\beta') = \mathbf{P}(\alpha')$ . We deduce that for every sequence  $\sigma$  from  $s_1$  to  $s_2$ , there exists a sequence  $\delta \not\geq \pi$  from  $s_1$  to  $s_2$  such that  $\mathbf{P}(\sigma) = \mathbf{P}(\delta) + k \cdot \pi$  for some non-negative integer  $k$ .  $\square$

**Insufficiency of the necessary conditions b and c for WMG-solvability.** In Figure 5, we provide an example of an FA system whose reachability graph satisfies all conditions of properties **b** and **c** but is not WMG-solvable. Its reachability graph is represented by the lts of Figure 6. We deduce that these conditions, when satisfied by a given lts, are not sufficient for ensuring the existence of a solution in the WMG subclass. Indeed, in Figure 5, each possible attempt of a construction leads to a contradiction, as detailed next.

- Non-existence of a WMG solution with six places:

To obtain a WMG solution structured as on the right of Figure 5, the following sequences must be feasible at the initial marking  $M_0$ :

$$\begin{array}{ll} t_3 t_1 \Rightarrow M_0(p_1) \geq 3 & t_3 t_1 t_3 t_2 t_1 t_3 t_2 t_1 t_3 t_2 t_3 \Rightarrow M_0(p_4) \geq 9 \\ t_3 t_1 t_2 t_3 t_1 t_3 t_2 t_1 \Rightarrow M_0(p_2) \geq 3 & t_3 t_1 t_3 \Rightarrow M_0(p_5) \geq 6 \\ t_2 t_3 t_1 t_2 \Rightarrow M_0(p_3) \geq 9 & t_2 t_3 t_1 t_2 t_3 t_1 t_3 t_1 t_2 \Rightarrow M_0(p_6) \geq 12 \end{array}$$

The sequence  $\sigma = t_2 t_3 t_1 t_3 t_1 t_3 t_2 t_1 t_3 t_3$  is then feasible in such a constrained WMG but is not enabled in the FA system  $\zeta$ .

- Non-existence of a WMG solution with fewer places:

Since the previous WMG system with six places and its necessary marking are too permissive, we deduce that the same contradicting sequence  $\sigma$  is also feasible in all less constrained WMGs, typically obtained by removing some places while retaining the necessary initial marking in the other places.

### Checking the necessary conditions in a pre-synthesis phase.

Properties **b** and **c** (or  $\neg \mathbf{c}$ ), to be checked in the pre-synthesis phase, are similar to the properties to be checked for CF-synthesis and are carefully analysed, formally as well as practically, in [15]. In particular, the complexity is at most quadratic in  $|S|$  and  $|T|$ .

In the next subsection, we exhibit subsets of states of the lts whose analysis is sufficient to ensure some constraints, allowing to perform fewer operations.



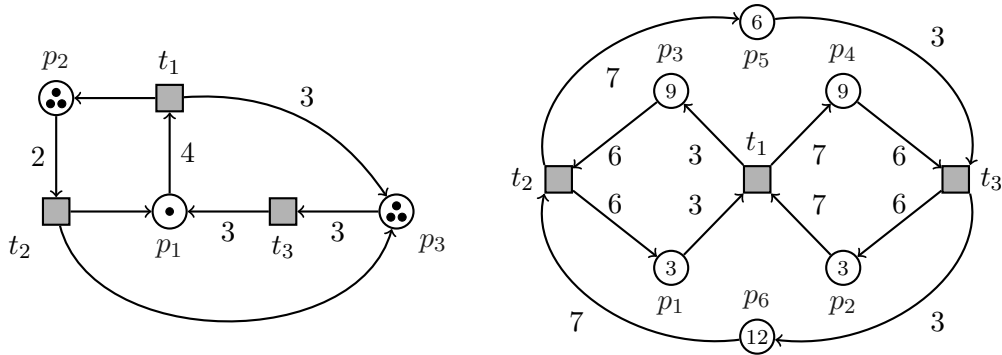


Figure 5: A Fork-Attribution (FA) system  $\zeta$  is pictured on the left. Its minimal prime T-semiflow  $\pi = (6, 3, 7)$  equals the Parikh vector of each small cycle of  $RG(\zeta)$ . The latter is persistent and backward persistent, reversible, finite and fulfills properties **b** and **c**. The most constrained WMG solution  $\zeta'$  whose reachability graph could be isomorphic to  $RG(\zeta)$  is depicted on the right: its weights are directly deduced from  $\pi$  (using the fact that, for each place  $p$ ,  $C(p) \cdot \pi = 0$ ) and, in each place, the given amount of tokens is necessary to enable the sequences of  $RG(\zeta)$ . However, this necessary initial marking already enables a sequence that is not feasible in  $\zeta$ , namely  $\sigma = t_2t_3t_1t_3t_1t_3t_2t_1t_3t_3$ . Since every possible variant  $\zeta''$  of  $\zeta'$  is less constrained than  $\zeta'$ , each such  $\zeta''$  also enables  $\sigma$ . We deduce that no WMG solves  $RG(\zeta)$ .

## 4.2. Regions and Separation Problems

Following the seminal works of Ehrenfeucht and Rozenberg [36, 37] on the synthesis of elementary nets, i.e., plain nets where markings are restricted to safe ones (no more than one token may occur in any place), the general strategy to synthesise a Petri net from a labelled transition system consists in building *regions* of the given lts. For the synthesis of elementary nets, regions are subsets of states in the lts that behave coherently with respect to the various labels. Its generalisation for Petri nets [30] looks like a place in a net together with its markings corresponding to the various states of the lts.

### Definition 4.2. (Regions)

A region of an lts  $TS = (S, \rightarrow, T, \iota)$  is a triple of functions  $r = (\mathcal{R}, \mathcal{B}, \mathcal{F}) \in \mathbb{N}^S \times \mathbb{N}^T \times \mathbb{N}^T$  such that for all  $(s, t, s') \in \rightarrow$ , both  $\mathcal{R}(s) \geq \mathcal{B}(t)$  and  $\mathcal{R}(s') = \mathcal{R}(s) - \mathcal{B}(t) + \mathcal{F}(t)$  hold.

Interpreting any region as a place in a Petri net,  $\mathcal{R}(s)$  is the marking of that place corresponding to state  $s$  (hence  $\mathcal{R}(\iota)$  corresponds to its initial marking).  $\mathcal{F}(t)$  (for *forward*) corresponds to the weight of the arc from that place to transition  $t$ , and  $\mathcal{B}(t)$  (for *backward*) corresponds to the weight of the arc from transition  $t$  to that place.

Synthesising a net then amounts to find regions solving two kinds of *separation* problems.

### Definition 4.3. (Separation problems)

Let  $TS = (S, \rightarrow, T, \iota)$  be an lts. A *state separation problem (SSP)* is a pair  $\{s, s'\}$  of states in  $S$  with  $s \neq s'$ . It is solved by a region  $(\mathcal{R}, \mathcal{B}, \mathcal{F})$  when  $\mathcal{R}(s) \neq \mathcal{R}(s')$  (meaning the region allows to discriminate between  $s$  and  $s'$ ). An *event/state separation problem (ESSP)* consists of a state  $s \in S$  and a label  $t \in T$  ( $s, t$ ) such that  $\neg s[t]$ . It is solved by a region  $(\mathcal{R}, \mathcal{B}, \mathcal{F})$  when  $\mathcal{R}(s) < \mathcal{B}(t)$  (meaning the place allows to exclude a forbidden transition from some state).

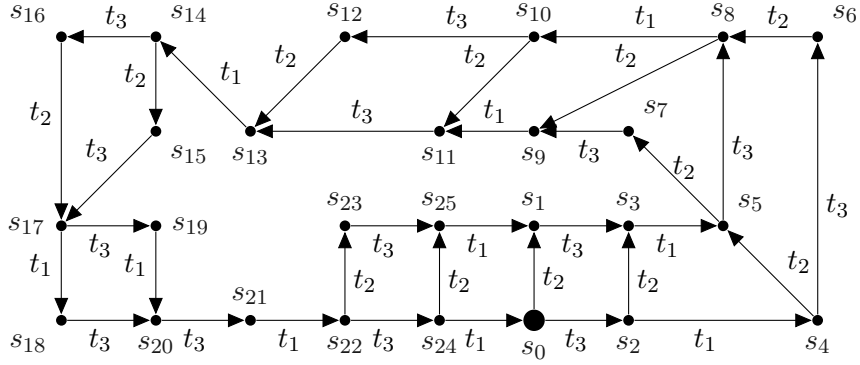


Figure 6: The reachability graph of the FA system  $\zeta$  on the left of Figure 5. Its initial state is denoted by  $s_0$  and is represented by the node larger than the other ones. The sequence  $\sigma$  defined in the latter figure is indeed not feasible in this reachability graph, since the last occurrence of  $t_3$  is not enabled from  $s_{16}$ . This RG fulfills properties **b** and **c**, but is not WMG-solvable.

The sets  $SSP_{TS}$  and  $ESSP_{TS}$  contain all  $SSP$  instances of  $TS$ , and all  $ESSP$  instances of  $TS$ , respectively. The set of all separation problems of  $TS$  is  $SP_{TS} = SSP_{TS} \cup ESSP_{TS}$ .

In summary, from the regions theory [30], constructing a Petri net solution to the problem of synthesis from a given lts, when such a solution exists, amounts to build a set of places (or regions) satisfying the following necessary and sufficient conditions:

- The number of tokens in any place must remain non-negative at each reachable marking described by a state in  $S$ .
- For each state  $s$  not allowing  $b$ , there must exist a place  $p$  such that  $M_s(p) < W(p, b)$ , where  $M_s$  is the marking associated to  $s$  (event/state separation).
- Any two different states  $s', s''$  must be distinguished by a place  $p'$  such that  $M_{s'}(p') \neq M_{s''}(p')$  (state separation).

In many cases (see for instance [38]) however, and in particular for the WMG-synthesis (see Theorem 4.8 below), but also for CF-synthesis (see for example [31, 20, 15]), the last constraint, i.e., the state separation property, arises from the other two and from the assumptions on  $TS$ .

In order to specialise these ideas to the synthesis of WMG-nets, we introduce WMG-regions by constraining the number of ‘input’ and ‘output’ transitions, together with related notions, and a way to construct such regions.

**Definition 4.4. (WMG-Regions)**

A *WMG-region* of an lts  $TS = (S, \rightarrow, T, \iota)$  is a triple of functions  $r = (\mathcal{R}, \mathcal{B}, \mathcal{F}) \in \mathbb{N}^S \times \mathbb{N}^T \times \mathbb{N}^T$  such that for all  $(s, t, s') \in \rightarrow$ , both  $\mathcal{R}(s) \geq \mathcal{B}(t)$  and  $\mathcal{R}(s') = \mathcal{R}(s) - \mathcal{B}(t) + \mathcal{F}(t)$  hold, together with  $|\{t | \mathcal{B}(t) > 0\}| \leq 1$  and  $|\{t | \mathcal{F}(t) > 0\}| \leq 1$ .

Allowing  $\mathcal{R} \in \mathbb{Z}^S$  in the above defines a *WMG-semi-region*. A *minimal WMG-region* is a WMG-region with the additional constraint that a state  $s$  exists with  $\mathcal{R}(s) = 0$  (expressing the absence of frozen tokens).

A WMG-region may be seen as a place of the kind illustrated in Figure 7 together with markings coherent with the various states of the lts. A WMG-semi-region is similar but allowing negative ‘markings’. Without the constraints on  $\mathcal{F}$  (number of input transitions), we get the CF-regions exploited for instance in [15].

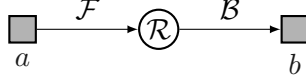


Figure 7: Graphical interpretation of a WMG-region.

**Corollary 4.5. ((Semi-)regions, constant additions and constant divisions)**

If  $c$  is a constant non-negative function on  $S$  and  $(\mathcal{R}, \mathcal{B}, \mathcal{F})$  is a WMG-region,  $(\mathcal{R}+c, \mathcal{B}, \mathcal{F})$  is also a WMG-region and  $(\mathcal{R}-c, \mathcal{B}, \mathcal{F})$  is a WMG-semi-region. Moreover, if  $TS$  is finite,  $(\mathcal{R}, \mathcal{B}, \mathcal{F})$  is a WMG-semi-region and  $c = \min_{s \in S} \mathcal{R}(s)$ , then  $(\mathcal{R}-c, \mathcal{B}, \mathcal{F})$  is a minimal WMG-region. Finally, if  $(\mathcal{R}, \mathcal{B}, \mathcal{F})$  is a WMG-region and  $d \in \mathbb{N}$  divides both  $\mathcal{B}$  and  $\mathcal{F}$ ,  $(\lfloor \mathcal{R}/d \rfloor, \mathcal{B}/d, \mathcal{F}/d)$  is also a WMG-region.

In order to build a WMG-region of a finite totally reachable lts  $TS = (S, \rightarrow, T, \iota)$ , we may choose  $a, b \in T$ ,  $\mathcal{F}(a), \mathcal{B}(b) \in \mathbb{N}$  (with  $\gcd(\mathcal{F}(a), \mathcal{B}(b)) = 1$ , unless  $\mathcal{F}(a) = 0 = \mathcal{B}(b)$ ) and  $\mathcal{R}(\iota) = 0$ . Then, for each  $s \in S$ , we choose a path  $\iota[\alpha]s$  (for instance while constructing a spanning tree) and define  $\mathcal{R}(s) = \mathcal{F}(a) \cdot \mathbf{P}(\alpha)(a) - \mathcal{B}(b) \cdot \mathbf{P}(\alpha)(b)$ . We check that for each arc  $(s, t, s') \in \rightarrow$ , if  $t \notin \{a, b\}$  then  $\mathcal{R}(s) = \mathcal{R}(s')$ ; if  $t = a \neq b$  then  $\mathcal{R}(s') = \mathcal{R}(s) + \mathcal{F}(a)$ ; if  $t = b \neq a$  then  $\mathcal{R}(s') = \mathcal{R}(s) - \mathcal{B}(b)$ , and if  $t = a = b$  then  $\mathcal{R}(s') = \mathcal{R}(s) + \mathcal{F}(a) - \mathcal{B}(b)$ . It is also possible to see  $\mathcal{F}(a)$  and  $\mathcal{B}(b)$  as variables, the previous constraints as a system of equations, and solve it. If it works, we get a WMG-semi-region, and with the constant shift  $\min_{s \in S} \mathcal{R}(s)$  explained in Corollary 4.5 we get a minimal WMG-region.

We may now define explicitly the various kinds of separation problems.

**Definition 4.6. (WMG-Separation problems)**

Let  $TS = (S, \rightarrow, T, \iota)$  be an lts. A state separation problem  $\{s, s'\}$  (with  $s \neq s'$ ) is (WMG-)solved by a WMG-region  $(\mathcal{R}, \mathcal{B}, \mathcal{F})$  when  $\mathcal{R}(s) \neq \mathcal{R}(s')$ . An event/state separation problem  $(s, t)$  is (WMG-)solved by a WMG-region  $(\mathcal{R}, \mathcal{B}, \mathcal{F})$  when  $\mathcal{R}(s) < \mathcal{B}(t)$ .

We shall denote by  $SSP_{-WMG}(TS)$  the set of state separation problems that cannot be WMG-solved for  $TS$  and by  $ESSP_{-WMG}(TS)$  the set of event/state separation problems that cannot be WMG-solved for  $TS$ .

For instance, if  $TS$  is not deterministic and  $s[t]s_1 \wedge s[t]s_2$  for some  $s \in S$  and  $t \in T$  with  $s_1 \neq s_2$ , we have  $\{s_1, s_2\} \in SSP_{-WMG}(TS)$ . The same is true if  $s_1[t]s \wedge s_2[t]s$ , or more generally if  $s[\alpha_1]s_1 \wedge s[\alpha_2]s_2$  with  $\mathbf{P}(\alpha_1) = \mathbf{P}(\alpha_2)$ , or  $s_1[\alpha_1]s \wedge s_2[\alpha_2]s$  with  $\mathbf{P}(\alpha_1) = \mathbf{P}(\alpha_2)$ .

These pairs  $\{s_1, s_2\}$  are not separable even with general Petri net regions, where there is no constraint on the non-null values of  $\mathcal{B}$  and  $\mathcal{F}$ , but there are also cases which are separable by general regions but not by WMG ones. This is illustrated by the lts on the bottom left of Figure 10:  $\{s_1, s_2\}$  is separated by the regions corresponding to the places in the net on the bottom right of that Figure, but it is easy to see that there is no WMG-region separating them (the only WMG-region is the one with null  $\mathcal{B}$  and  $\mathcal{F}$ ).

In order to analyse the state separation problems, it may be observed that, thanks to Corollary 4.5, we can consider WMG-semi-regions instead of WMG ones; hence we can fix  $\mathcal{R}(\iota) = 0$ . Any constant shift applied to a solution will preserve the state separations, hence will allow to get a WMG-region separating the same states. To check the (non-)separability of a pair of states  $\{s_1, s_2\}$  with  $s_1 \neq s_2$ , we thus have to check if a pair of transitions  $a$  and  $b$  exists (here we allow  $b = a$ ) such that the system of inequations in  $\mathcal{F}(a)$  and  $\mathcal{B}(b)$  in  $\mathbb{N}$  composed by

- $\mathcal{R}(\iota) = 0$ ,
- $\forall (s, x, s') \in \rightarrow$ :  
 $\mathcal{R}(s') = \mathcal{R}(s) + (\mathcal{F}(a) \text{ if } x = a, 0 \text{ otherwise}) - (\mathcal{B}(b) \text{ if } x = b, 0 \text{ otherwise})$   
 (this will allow to build the function  $\mathcal{R}$  progressively from  $\iota$ ),
- $\mathcal{R}(s_1) \neq \mathcal{R}(s_2)$ ,

has a solution or not.

For the event/state separation problems, we cannot consider WMG-semi-regions since constant shifts may enable new transitions. However, we can restrict our attention to minimal WMG-regions, since, in each Petri net solution, tokens that are never used can be removed from the initial marking while preserving the structure of the reachability graph, yielding a new solution in which each place is eventually emptied. Hence, for each pair  $s \in S$  and  $t \in T$  such that  $\neg s[t]$ , we have to check if there exist a pair of transitions  $a$  and  $b$  (we allow again  $b = a$ ) and a state  $s' \in S$ , such that the system of inequations in  $\mathcal{F}(a)$  and  $\mathcal{B}(b)$  in  $\mathbb{N}$  composed by

- $\mathcal{R}(s') = 0$ ,
- $\forall (s_1, x, s_2) \in \rightarrow$ :  
 $\mathcal{R}(s_2) = \mathcal{R}(s_1) + (\mathcal{F}(a) \text{ if } x = a, 0 \text{ otherwise}) - (\mathcal{B}(b) \text{ if } x = b, 0 \text{ otherwise})$   
 (this will allow to build the function  $\mathcal{R}$  progressively from  $s'$ ),
- $\forall s_3 \in S : \mathcal{R}(s_3) \geq 0$ ,
- $\mathcal{R}(s) < \mathcal{B}(t)$

has a solution or not.

However, following the last part of the proof of Theorem 8 in [15], it occurs that the state separation problems are automatically solvable if the event/state ones are solvable and if property **b** is satisfied. To show this is also true for our problems, we need first to recall the notion of *home states*, which we also use in the next subsection.

**Definition 4.7. (Home states)**

A state  $\tilde{s} \in S$  of a totally reachable lts  $TS = (S, \rightarrow, T, \iota)$  is a home state if  $\forall s \in S : \tilde{s} \in [s]$ , i.e.  $\tilde{s}$  is reachable from each reachable state.

**Theorem 4.8. (Redundant state-separation checking)**

If a lts  $TS$  satisfies property **b** and all event/state separation problems are solved by WMG-regions, then all the state separation problems are also solved by the same WMG-regions.

**Proof:**

It is not hard to see [39] that in a finite, totally reachable, deterministic, and persistent lts (hence in particular when property **b** is satisfied), home states always exist.

Since all event/state separation problems are solved by WMG-regions, we can build a WMG system  $\zeta$  with exactly the same language as the given lts (i.e.,  $M_0[\alpha]$  in  $\zeta \Leftrightarrow \iota[\alpha]$  in  $TS$ ).

Let us assume that some state separation problem is not solved by the constructed WMG-regions: there are thus two different states  $s_1$  and  $s_2$  that correspond to the same marking. In that case, let  $s_1[\beta]q_1$  be a path to a home state  $q_1$  of  $TS$ . Since  $s_1$  and  $s_2$  correspond to the same marking and the languages are the same,  $s_2[\beta]q_2$  for some state  $q_2$  corresponding to the same marking as  $q_1$ . Since  $q_1$  is a home state, there is a path  $q_2[\alpha]q_1$ . Since the languages are the same and  $q_1, q_2$  correspond to the same marking, we have  $q_2[\alpha]q_1[\alpha]q_3[\alpha]q_4 \dots$ , and from the finiteness and weak periodicity of  $TS$ , we must have  $q_1 = q_2$ . But then, by backward determinism, we also have  $s_1 = s_2$ .  $\square$

### 4.3. Constraints and Subsets of States Relevant to WMG-Synthesis

In the following, we describe some constraints that must be fulfilled in order to synthesise a WMG. Also, we define two subsets of the states of the given lts that are sufficient to check in order to fulfill several constraints over all states, decreasing potentially the size of the systems of constraints to solve.

A WMG synthesis amounts to build places (or WMG-regions) of the kind schematised in Figure 8.

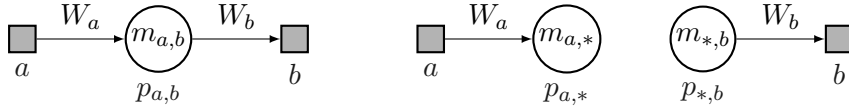


Figure 8: Possible types of places for the synthesis of a WMG  $(N, M_0)$ , with initial marking  $m_{a,b} = M_0(p_{a,b})$ ,  $m_{a,*} = M_0(p_{a,*})$  and  $m_{*,b} = M_0(p_{*,b})$ .

**Constraints related to places in the WMG.** Note that a place  $p_{a,*}$  is equivalent to a place  $p_{a,b}$  with  $W_b = 0$ , and a place  $p_{*,b}$  is equivalent to a place  $p_{a,b}$  with  $W_a = 0$ . In a place  $p_{a,b}$ , we can always choose  $W_a$  and  $W_b$  relatively prime without loss of generality, with an adequate initial marking  $M_0$ . Indeed, if  $W_a = k \cdot W'_a$  and  $W_b = k \cdot W'_b$ , the place  $p_{a,b}$  behaves the same as  $p'_{a,b}$  with weights  $W'_a$  and  $W'_b$  and initial marking  $\lfloor m_{a,b}/k \rfloor$ . If  $a$  and  $b$  are the same label, then we have a single transition and the place is equivalent to either a place  $p_{a,*}$ ,  $p_{*,a}$  or no place at all, depending on the sign of the difference between  $W_a$  and  $W_b$ . In a place  $p_{a,*}$ , the initial marking  $M_0(p_{a,*})$  may always be chosen as 0 and the weight  $W_a$  as 1. In a place  $p_{*,b}$ , we must have  $W_b \leq M_0(p_{*,b})$  (otherwise the lts would not be weakly live), and the weight  $W_b$  can always be chosen as 1, with an adequate choice of the initial marking  $M_0$ .

If  $T = \emptyset$ ,  $TS$  is reduced to its initial state and the (minimal) solution is the empty Petri net. If  $T = \{a\}$  is a singleton, either  $TS$  is acyclic, in the form of a single chain, and the minimal solution is a place  $p_{*,a}$ , with an initial marking deduced from the length of the chain, or it is a loop  $\iota[a]\iota$  with a minimal solution reduced to a transition  $a$  without any place. Hence, in the following, we assume without loss of generality that  $|T| > 1$ . We shall also assume that the lts to be synthesised satisfies property **b** and either acyclicity or **c**.

$M_0$  is the marking corresponding to the initial state  $\iota$ ; consider any state  $s \in S$  with a shortest sequence from  $\iota$  to  $s$ , meaning that no other sequence from  $\iota$  to  $s$  has a smaller Parikh

vector. By Lemma 4.1 (point 2 or 3), such a sequence exists, and all such sequences from  $\iota$  to  $s$  share the same Parikh vector  $\Delta_s$ . The marking corresponding to state  $s$  is given by  $M_s(p_{a,b}) = M_0(p_{a,b}) + \Delta_s(a) \cdot W_a - \Delta_s(b) \cdot W_b$ .

**Two subsets of states relevant to the WMG-synthesis.** The constraints on the positivity of place markings and on their usage to solve event/state separations are linked to two particular subsets of states of  $TS$ : for each label  $x \in T$ , we define

$$OX(x) = \{r \in S \mid r[t] \Rightarrow t = x\} \text{ and } NXX(x) = \{s \in S \mid \neg s[x] \wedge \forall s' \in s^\bullet : s'[x]\}$$

For each state  $s$  in  $OX(x)$  (the notation stemming from “Only X”), the only arc starting at  $s$ , if any, is labelled  $x$ . Let us consider a place  $p_{a,b}$  and a longest sequence without  $a$  starting from some state  $s$ . This sequence is finite since the lts is finite, and each cycle along the sequence, if any, has support  $T$ , hence contains an  $a$ . Thus, we reach a state  $r$  either without successor (this may only occur if the lts is acyclic) or with a single output  $a$ , hence in  $OX(a)$  in both cases, and  $M_s(p_{a,b}) \geq M_r(p_{a,b})$ . As a consequence, to check that all markings of  $p_{a,b}$  reachable from  $\iota$  are non-negative, we only have to check the states in  $OX(a)$ :  $(\forall r \in OX(a) : M_r(p_{a,b}) \geq 0) \Leftrightarrow (\forall s \in S : M_s(p_{a,b}) \geq 0)$ . This ensures that the initial marking of  $p_{a,b}$  is large enough.

For each state  $s$  in  $NXX(x)$ ,  $x$  cannot be executed at  $s$  (hence the first two letters  $NX$  of the notation), but in each next state  $s'$ , if any,  $x$  is enabled (hence the last letter  $X$  of the notation). Let us assume that a place  $p_{a,b}$  (allowing all valid transitions, as expressed above through  $OX(a)$ ) can be used to exclude performing  $b$  at some state  $s$  (we thus assume  $\neg s[b]$  and the initial marking of  $p_{a,b}$  is small enough), meaning  $M_s(p_{a,b}) < W_b$ . If  $s'[t]s$  with  $\neg s'[b]$  (which implies  $t \neq b$ ), then  $M_s(p_{a,b}) \geq M_{s'}(p_{a,b})$ , so that the same place  $p_{a,b}$  disables  $b$  at  $s'$ . Moreover, the longest chains of states excluding to perform  $b$  are necessarily finite since  $b$  occurs in any non-empty cycle and the lts is finite; hence they all end in states of  $NXX(b)$ . As a consequence, in order to exclude performing  $b$  when necessary, one only has to find, for each state  $r \in NXX(b)$ , a place  $p_{a,b}$  such that  $M_r(p_{a,b}) < W_b$ . We thus have that, for each  $b \in T$ ,  $(\forall s \in S : \neg s[b] \Rightarrow \exists p_{a,b} \text{ with } M_s(p_{a,b}) < W_b) \Leftrightarrow (\forall s \in NXX(b) : \exists p_{a,b} \text{ with } M_s(p_{a,b}) < W_b)$ . We may mention that, in some cases, a same place  $p_{a,b}$  can be used for several states in  $NXX(b)$ .

In our case, for any label  $x$ , the states in  $NXX(x)$  have a very special shape, highlighted in the following lemma whose proof is illustrated in Figure 9.

**Lemma 4.9. (Single outputs of the states in  $NXX$ )**

Let  $TS = (S, \rightarrow, T, \iota)$  be a lts satisfying property **b**. If  $x, a, b \in T$ ,  $r \in NXX(x)$ ,  $r[a]$  and  $r[b]$ , then  $a = b$ .

**Proof:**

Let us assume that  $r[a]r_1$  and  $r[b]s_1$  with  $a \neq b$ . Since  $r \in NXX(x)$ , we have  $r_1[x]r_2$  and  $s_1[x]s_2$  for some states  $r_2, s_2$ . By persistence (and determinism), we also have  $r_1[b]s$ ,  $s_1[a]s$ ,  $s[x]s'$ ,  $r_2[b]s'$  and  $s_2[a]s'$  for some  $s, s'$ . By backward persistence, we then have  $s''[a]r_2$  and  $s''[b]s_2$  for some  $s''$ , as well as  $s'''[x]s''$  and  $s'''[b]s_1$  for some  $s'''$ . Finally, by backward determinism,  $s''' = r$  and  $r[x]$ , contradicting the fact that  $r \in NXX(x)$ .  $\square$

If  $TS$  is acyclic, by persistence there is a unique (maximal) state without successor; it is then the unique home state; let us call it  $s_\infty$ . We then have  $s_\infty \in \bigcap_{x \in T} NXX(x)$ . If  $TS$  is cyclic, there is

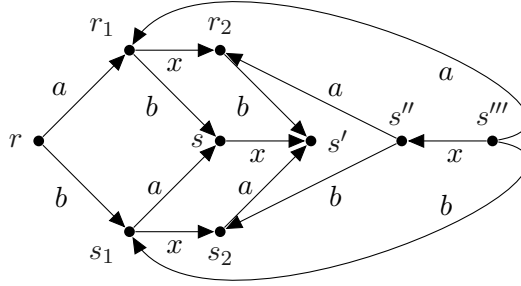


Figure 9: Illustration of the proof of Lemma 4.9.

no such state, but every state is a home state. In any case, we may have several states  $s$  in some  $NXX(x)$  and label  $a \neq x$  with  $s[ax]$ .

Lemma 4.9 becomes wrong when backward persistence is lifted: a state  $r$  and distinct transitions  $a, b, x$  may then exist such that  $r[ax]$ ,  $r[bx]$  and  $\neg r[x]$ . This state belongs to  $NXX(x)$ , but it is possible to obtain this situation in the reachability graph of a choice-free system, in which places are allowed to have several inputs: two inputs  $a$  and  $b$  of some place  $p$  may be enabled by the same state, in such a way that any firing of  $a$  or  $b$  enables the output  $x$  of  $p$ .

The next subsections are devoted to WMG-synthesis procedures, assuming that property **b** is fulfilled by the lts. We define systems of inequalities that exploit the structure and properties of WMGs we just exhibited and are consequently easier to solve than in the bounded choice-free case [14, 15].

#### 4.4. Computational Synthesis in the General Cyclic Case

Let  $TS = (S, \rightarrow, T, \iota)$  be a lts satisfying properties **b** and **c**, denoting by  $\pi$  the unique minimal Parikh vector of small cycles, with support  $T$ . Each place  $p_{a,b}$  must satisfy  $W_a \cdot \pi(a) = W_b \cdot \pi(b)$ , thus we can choose  $W_a = \pi(b)$  and  $W_b = \pi(a)$  (or any proportional values<sup>3</sup>, in particular  $\pi(b)/\gcd(\pi(a), \pi(b))$  and  $\pi(a)/\gcd(\pi(a), \pi(b))$ ), and the only parameter that still needs to be fixed is the initial marking (plus the exact pairs  $a, b$  for which we need those places).

For each  $b \in T$ , we need such a place  $p_{a,b}$  if there is a state  $s \in NXX(b)$  such that  $s[ab]$  (otherwise, there is no way to enable a  $b$  after an  $a$  when  $b$  is not directly enabled). We denote by  $pred(b)$  the set  $\{a \in T \mid \exists s \in NXX(b), s[ab]\}$  and, for any  $a \in pred(b)$ ,  $NXX(a, b) = \{s \in NXX(b) \mid s[ab]\}$ .

For each  $a, b \in T$  such that  $a \in pred(b)$ , since  $W_a = \pi(b)$  and  $W_b = \pi(a)$ , we have to solve the following constraints (in  $M_0$ , over the non-negative integers):

<sup>3</sup>This is the only way to define an adequate  $p_{a,b}$ ; in particular, there is no  $p_{*,b}$  or  $p_{a,*}$  in a bounded cyclic WMG net.

$$\begin{cases} \forall s \in OX(a) : M_0(p_{a,b}) \geq \Delta_s(b) \cdot \pi(a) - \Delta_s(a) \cdot \pi(b) \\ \forall s \in NXX(a, b) : M_0(p_{a,b}) < \Delta_s(b) \cdot \pi(a) - \Delta_s(a) \cdot \pi(b) + \pi(a) \end{cases}$$

This amounts to first compute

$$M_0(p_{a,b}) = \max_{s \in OX(a)} \{ \Delta_s(b) \cdot \pi(a) - \Delta_s(a) \cdot \pi(b) \}$$

and then to check that, for each  $s \in NXX(a, b)$ ,

$$M_0(p_{a,b}) < \Delta_s(b) \cdot \pi(a) - \Delta_s(a) \cdot \pi(b) + \pi(a).$$

If each such system of constraints is solvable, we obtain a WMG solution of  $TS$ . Otherwise, there is no solution and the reason is known.

#### 4.5. Computational Synthesis in the General Acyclic Case

In the acyclic case, we may first apply the factorisation techniques of [33, 34, 35] to check if the given lts is prime (i.e., it cannot be further factorised) and thus has a chance to have a connected solution. The weights  $W_a$  and  $W_b$  around the place  $p_{a,b}$  are not constrained by a T-semiflow. Thus, we may need variants of such places (differing by the weights  $W_a$ ,  $W_b$  and the initial marking). We may also need places  $p_{*,b}$  and  $p_{a,*}$ ; in particular, a place  $p_{*,b}$  with  $W_b = 1$  and  $M_0 = \Delta_{s_\infty}(b)$  excludes executing  $b$  at the final state  $s_\infty$ . Such a place may be redundant with other ones, but we do not aim here at building an optimal solution (i.e., with the least number of places): we focus on the existence of a solution and on its construction.

In this acyclic case, the enabledness of labels is described by the first set of constraints below, using again the sufficient condition stating that the markings at states from  $OX(a)$  must be non-negative. The last constraint expresses that the place is useful for excluding some transition from some state.

For each  $b \in T$ ,  $a \in \text{pred}(b)$  and  $s \in NXX(a, b)$ , we have to solve the following constraints (in  $M_0(p_{a,b}), W_a, W_b \in \mathbb{N}$ ):

$$\begin{cases} \forall s' \in OX(a) : M_0(p_{a,b}) \geq \Delta_{s'}(b) \cdot W_b - \Delta_{s'}(a) \cdot W_a \\ M_0(p_{a,b}) < \Delta_s(b) \cdot W_b - \Delta_s(a) \cdot W_a + W_b. \end{cases}$$

To solve such a system, we can first consider the system in  $W_a$  and  $W_b$ :

$$\forall s' \in OX(a) : \Delta_s(b) \cdot W_b - \Delta_s(a) \cdot W_a + W_b > \Delta_{s'}(b) \cdot W_b - \Delta_{s'}(a) \cdot W_a$$

, i.e.,  $\forall s' \in OX(a) : [\Delta_s(b) - \Delta_{s'}(b) + 1] \cdot W_b > [\Delta_s(a) - \Delta_{s'}(a)] \cdot W_a$

and then check if there exists a solution satisfying:

$$\Delta_s(b) \cdot W_b - \Delta_s(a) \cdot W_a + W_b > 0.$$

If each such system of constraints is solvable, we obtain a WMG solution of  $TS$ . Otherwise, no solution exists and we know the reason.



## 5. Approximate Synthesis

In case a WMG-synthesis fails, constructing an approximate solution can be an adequate option in various situations. There are several ways to proceed.

For instance, instead of considering unlabelled Petri nets, we may shift to labelled ones, i.e., nets where transitions have a label so that, in firing sequences as in reachability graphs, each transition is replaced by its label. There may be various solutions, and then we may try to find one with a minimal number of transitions. There is always a labelled Petri net solution with a maximal number of transitions, where each state  $s$  is associated to a different place  $p_s$ , with an empty initial marking but the initial one which has one token, and each arc  $\alpha = s[t]s'$  is associated to a fresh transition  $t_\alpha$  with label  $t$ , but the result is a P-net and it may even happen that there is no labelled WMG-solution. This is illustrated by Figure 10.

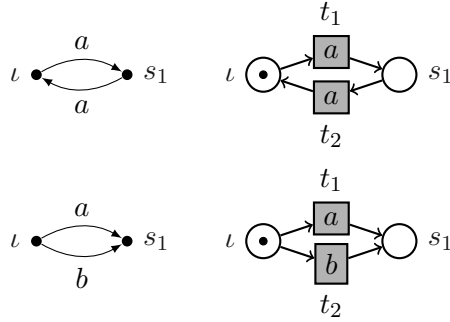


Figure 10: Above, on the left, an lts which is not WMG-solvable; on the right, a labelled WMG-solution. Below, on the left, another lts which is not WMG-solvable; on the right, a labelled solution which is a P-net but not a WMG one, and there is no labelled WMG-solution.

Hence, in the sequel, we exploit instead a method used in [17, 18], based on lts homomorphisms.

### Definition 5.1. (Lts homomorphism)

An lts homomorphism from lts  $TS = (S, \rightarrow, T, \iota)$  to lts  $TS' = (S', \rightarrow', T', \iota')$ , is a partial function  $f : S \rightarrow S'$ , defined for  $\iota$  with  $f(\iota) = \iota'$  and  $(s_1; t; s_2) \in \rightarrow \Rightarrow (f(s_1); t; f(s_2)) \in \rightarrow'$  when  $f$  is defined for  $s_1$ , which implies that  $f$  is also defined for  $s_2$  and for every state reachable from any state for which  $f$  is defined. If such a function  $f$  exists, we note  $TS \sqsubseteq TS'$  (via  $f$ ).

From the definition,  $f$  is defined on  $[\iota]$  at least, and  $f([\iota]) \subseteq [\iota']$ . With respect to [17, 18] our definition is slightly more general since it uses partial functions instead of total ones. In the sequel, this will allow to start from an lts  $TS$  that is not necessarily totally reachable nor deterministic. It may be observed that  $\sqsubseteq$  is a partial order.

### 5.1. WMG-solvable Over-Approximation

Starting from any lts  $TS = (S, \rightarrow, T, \iota)$  with  $T$  finite, we show that a finite WMG-solvable lts  $TS'$  exists such that  $TS \sqsubseteq TS'$ . Indeed, let  $TS_{\max}(T)$  be the lts  $(\{\iota\}, \{(\iota, t, \iota) \mid t \in T\}, T, \iota)$  with a single state (the initial one) and a loop around it labelled by each label in  $T$ ; with  $f(s) = \iota$

for each  $s \in S$ , we trivially have  $TS \sqsubseteq TS_{\max}(T)$  and  $TS_{\max}(T)$  has a WMG-solution with isolated transitions  $t$  for each  $t \in T$ . In general, there are many WMG-solvable lts  $TS'$  with  $TS \sqsubseteq TS'$ , which we call *WMG-solvable over-approximations of  $TS$* .

The problem is then to find a *minimal* WMG-solvable over-approximation of  $TS$ , if possible. Each PN-solvable lts is totally reachable and deterministic. Suppose that the latter two properties are fulfilled by lts  $TS$  and  $TS'$ : if  $TS \sqsubseteq TS'$  and  $TS' \sqsubseteq TS$ , then they are isomorphic. This fact is important when searching a minimal over-approximation.

On the contrary, if  $TS$  is not totally reachable and  $TS'$  is obtained from  $TS$  by dropping states that are not reachable from  $\iota$  (together with the arcs to/from them), then trivially  $TS \sqsubseteq TS'$  and  $TS' \sqsubseteq TS$ . Hence, in the process of finding a minimal WMG-solvable over-approximation of a finite lts  $TS$ , we can always first build  $TS_0$  from  $TS$  by dropping all the states non-reachable from  $\iota$  (then  $TS \sqsubseteq TS_0$ ).

In the rest of this section, so as to show that every finite lts  $TS$  has a unique (up to isomorphism) minimal WMG-solvable over-approximation, we exploit (and detail a bit more) a construction investigated in [18] that forms a series of finite lts  $TS_i$  such that  $TS_i \sqsubseteq TS_{i+1}$  for each  $i \in \mathbb{N}$ , and if  $TS_i = TS_{i+1}$  (up to isomorphism) then  $TS_i$  is the desired minimal over-approximation. Here, since the considered lts do not necessarily satisfy all the conditions of **b**, we need to consider state separation problems in addition to event/state ones.

In Subsection 4.2, we defined WMG-regions, WMG-semi-regions and the separation problems involved in a general synthesis process. In the rest of this section, we exploit them to compute a minimal WMG-over-approximation of a given lts.

## 5.2. Fixed Point for WMG-Over-Approximations

To construct the approximation, we consider the following steps:

- starting from the given lts, we remove iteratively its WMG-unsolvable separation problems by merging states and adding new arcs to the lts, thus building a sequence of lts;
- from a previous result, we derive that this sequence leads to a fixed point, which is the minimal WMG-solvable over-approximation.

From the (finite) lts  $TS_0$  defined as the given finite  $TS$  without its non-reachable states, we define next a series of WMG-over-approximations  $\{TS_i \mid i \in \mathbb{N}\}$ .

First, let  $\widetilde{TS}_i = \text{Merge}_{WMG}(TS_i)$  be  $TS_i$  where all pairs of states  $\{s_1, s_2\} \in \text{SSP}_{\neg WMG}(TS_i)$  are merged. The result is fully forward and backward deterministic, as well as strongly cycle consistent; it is also finite and over-approximates  $TS_i$ . This may suppress event/state separation problems, but some may remain and new ones may occur, rendering the built lts non-WMG-solvable.

Next, let  $TS_{i+1} = \text{Expand}_{WMG}(\widetilde{TS}_i)$  be  $\widetilde{TS}_i$  where one adds a fresh state  $s'$  for each  $(s, t) \in \text{ESSP}_{\neg WMG}(\widetilde{TS}_i)$ , with an arc  $(s, t, s')$ . This wipes out the cases where an arc is missing while over-approximating  $\widetilde{TS}_i$ , but may introduce new WMG-unsolvable state separation problems.

We then specialise<sup>4</sup> Theorem 5.4.4 of [18] (see also [17]):

<sup>4</sup>Unfortunately, this does not apply to choice-free synthesis, since the result requires that the considered Petri net subclass is stable with respect to *complement* regions (details on complement regions can be found in [30, 17, 18]). This works for WMGs, but not for CF nets. In the latter case, the chains may be infinite, leading to unbounded nets.

**Theorem 5.2. (Minimal WMG-solvable over-approximation)**

Given a finite Its  $TS$ , the chain of Its thus defined reaches a fixed point which is the unique-up-to-isomorphism finite minimal WMG-solvable over-approximation of  $TS$ .

**Example.** Let us consider the Its  $TS$  in Figure 11. State  $s_5$  is not reachable from  $\iota$ , hence it is dropped in  $TS_0$ . In  $TS_0$ ,  $s_6$  cannot be separated from  $s_2$ , hence they are merged in  $\widetilde{TS}_0$ . In  $\widetilde{TS}_0$ ,  $b$  cannot be WMG-separated from  $s_1$ , as well as  $a$  from  $s_4$ : this yields the expansion  $TS_1$ . There is no unsolvable WMG state separation problem in  $TS_1$ , which is then the same as  $\widetilde{TS}_1$ . In the latter,  $b$  cannot be WMG-separated from  $s_8$ , as well as  $a$  from  $s_7$ : this yields the expansion  $TS_2$ . In  $TS_2$ , two new unsolvable state separations arise, leading to merge  $s_9$  with  $s_3$  and  $s_{10}$  with  $s_1$ . No more (state or event/state) separation problems arise from now on, so that we reach the wanted fixed point  $TS_3$ . The corresponding WMG-solution is shown on the bottom right.

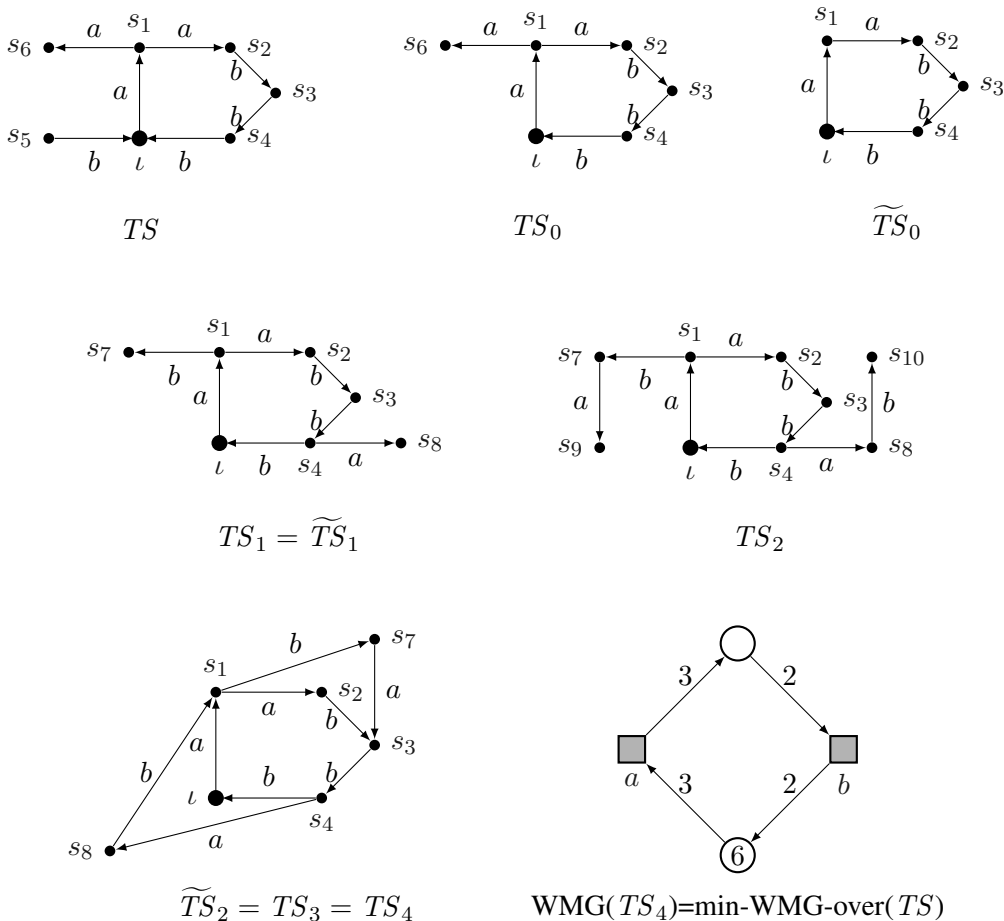


Figure 11: Construction of the minimal WMG-over-approximation  $TS_4$  of  $TS$  and the corresponding WMG-solution.

## 6. Conclusions and Perspectives

Weighted marked graphs (WMGs) form a well-known subclass of Petri nets with numerous real-life applications. These nets have been extensively studied in previous works, leading to strong theoretical results.

For this class, we obtained new behavioural properties of WMGs expressed on their reachability graph, including backward persistence and strong similarities between the sequences sharing the same starting state and the same destination state. We also delineated necessary structural conditions that must be fulfilled by a labelled transition system to be WMG-solvable. We showed however that these necessary conditions are not sufficient.

Using these properties, we designed a general and efficient synthesis procedure dedicated to the WMG class, specialising previous methods devised for choice-free nets.

Finally, in case no WMG solves the given lts, we showed the existence and described the construction of a minimal WMG-solvable over-approximation of the lts. This allows our general synthesis method to create a WMG whose reachability graph minimally includes the specification.

A perspective is to develop additional properties of WMGs in order to enhance the pre-synthesis phase, allowing to discard more non-solvable systems promptly. Ideally, such properties should characterise the WMG-solvable labelled transition systems in a purely structural way, in the spirit of the methods designed for plain marked graphs and T-systems in [19, 20].

## Acknowledgements

The authors thank the anonymous referees for their useful remarks and suggestions.

## References

- [1] Desel J, Esparza J. Free Choice Petri Nets, volume 40 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, New York, USA, 1995.
- [2] Hujsa T, Delosme JM, Munier-Kordon A. On Liveness and Reversibility of Equal-Conflict Petri Nets. *Fundamenta Informaticae*, 2016. **146**(1):83–119. doi:10.3233/FI-2016-1376.
- [3] Hujsa T, Devillers R. On Liveness and Deadlockability in Subclasses of Weighted Petri Nets. *Application and Theory of Petri Nets and Concurrency: 38th International Conference, Petri NETS 2017, Zaragoza, Spain, Proceedings*, 2017. pp. 267–287. doi:10.1007/978-3-319-57861-3\_16.
- [4] Silva M, Teruel E, Colom JM. Linear algebraic and linear programming techniques for the analysis of place/transition net systems. *Lectures on Petri Nets I: Basic Models, LNCS 1491*, 1998. pp. 309–373. doi:10.1007/3-540-65306-6\_19.
- [5] Teruel E, Silva M. Structure theory of Equal Conflict systems. *Theoretical Computer Science*, 1996. **153**(1&2):271–300. doi:10.1016/0304-3975(95)00124-7.
- [6] Lee EA, Messerschmitt DG. Synchronous Data Flow. *Proceedings of the IEEE*, 1987. **75**(9):1235–1245.

- [7] Lee E, Messerschmidt D. Static scheduling of synchronous data flow programs for digital signal processing. *IEEE Transaction on Computers*, C-36(1), 1987. pp. 24–35. doi:10.1109/TC.1987.5009446.
- [8] Pino JL, Bhattacharyya SS, Lee EA. A hierarchical multiprocessor scheduling framework for synchronous dataflow graphs. Technical report, University of California, Berkeley, 1995.
- [9] Sriram S, Bhattacharyya SS. Embedded multiprocessors: scheduling and synchronization. Signal Processing and Communications. CRC Press, 2009.
- [10] Teruel E, Chrzastowski-Wachtel P, Colom JM, Silva M. On weighted T-systems. *13th International Conference on Application and Theory of Petri Nets and Concurrency (ICATPN), LNCS 616*, 1992. pp. 348–367. doi:10.1007/3-540-55676-1\_20.
- [11] Marchetti O, Munier-Kordon A. A sufficient condition for the liveness of Weighted Event Graphs. *European Journal of Operational Research*, 2009. **197**(2):532–540. doi:10.1016/j.ejor.2008.07.037.
- [12] Hujsa T. Contribution to the study of weighted Petri nets. Ph.D. thesis, Pierre and Marie Curie University, Paris, France, 2014. URL <https://tel.archives-ouvertes.fr/tel-01127406>.
- [13] Hujsa T, Delosme J, Kordon AM. On the Reversibility of Well-Behaved Weighted Choice-Free Systems. *Application and Theory of Petri Nets and Concurrency - 35th International Conference, PETRI NETS 2014, Tunis, Tunisia*, 2014. pp. 334–353. doi:10.1007/978-3-319-07734-5\_18.
- [14] Best E, Devillers R. Synthesis of Bounded Choice-Free Petri Nets. *26th International Conference on Concurrency Theory, CONCUR 2015, Madrid, Spain*, 2015. pp. 128–141. doi:10.4230/LIPIcs.CONCUR.2015.128.
- [15] Best E, Devillers R, Schlachter U. Bounded Choice-Free Petri Net Synthesis: Algorithmic Issues. *Acta Informaticae*, 2018. **55**(7):575–611. doi:10.1007/s00236-017-0310-9.
- [16] Devillers R, Erofeev E, Hujsa T. Synthesis of Weighted Marked Graphs from Constrained Labelled Transition Systems. *International Workshop on Algorithms & Theories for the Analysis of Event Data, Bratislava, Slovakia, Proceedings*, 2018. pp. 75–90.
- [17] Schlachter U. Over-Approximative Petri Net Synthesis for Restricted Subclasses of Nets. *LATA 2018, LNCS 10792*, 2018. pp. 296–307.
- [18] Schlachter U. Petri Net Synthesis and Modal Specifications. Ph.D. thesis, University of Oldenburg, Germany, 2018. URL <http://oops.uni-oldenburg.de/3755>.
- [19] Best E, Devillers R. Characterisation of the State Spaces of Live and Bounded Marked Graph Petri Nets. In: 8th International Conference on Language and Automata Theory and Applications (LATA 2014). 2014 pp. 161–172. doi:10.1007/978-3-319-04921-2\_13.
- [20] Best E, Devillers R. State space axioms for T-systems. *Acta Informaticae*, 2015. **52**(2-3):133–152. doi:10.1007/s00236-015-0219-0.
- [21] Best E, Devillers R. Characterisation of the State Spaces of Marked Graph Petri Nets. *Information and Computation*, 2017. **253**(3):399–410. doi:10.1016/j.ic.2016.06.006.
- [22] Devillers R, Hujsa T. Analysis and Synthesis of Weighted Marked Graph Petri Nets. *Application and Theory of Petri Nets and Concurrency - 39th International Conference, PETRI NETS 2018, Bratislava, Slovakia, Proceedings*, 2018. pp. 19–39. doi:10.1007/978-3-319-91268-4\_2.

- [23] Hujsa T, Delosme JM, Munier-Kordon A. Polynomial Sufficient Conditions of Well-Behavedness and Home Markings in Subclasses of Weighted Petri Nets. *ACM Trans. Embed. Comput. Syst.*, 2014. **13**(4s):141:1–141:25. doi:10.1145/2627349.
- [24] Delosme J, Hujsa T, Munier-Kordon A. Polynomial Sufficient Conditions of Well-Behavedness for Weighted Join-Free and Choice-Free Systems. In: 13th International Conference on Application of Concurrency to System Design (ACSD 2013). 2013 pp. 90–99. doi:10.1109/ACSD.2013.12.
- [25] Crespi-Reghizzi S, Mandrioli D. A Decidability Theorem for a Class of Vector-Addition Systems. *Inf. Process. Lett.*, 1975. **3**(3):78–80. doi:10.1016/0020-0190(75)90020-4.
- [26] Teruel E, Colom JM, Silva M. Choice-Free Petri Nets: a Model for Deterministic Concurrent Systems with Bulk Services and Arrivals. *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, 1997. **27**(1):73–83. doi:10.1109/3468.553226.
- [27] Best E, Devillers R. Synthesis and reengineering of persistent systems. *Acta Informaticae*, 2015. **52**(1):35–60. doi:10.1007/s00236-014-0209-7.
- [28] Commoner F, Holt A, Even S, Pnueli A. Marked Directed Graphs. *J. Comput. Syst. Sci*, 1971. **5**(5):511–523. doi:10.1016/S0022-0000(71)80013-2.
- [29] Keller RM. A Fundamental Theorem of Asynchronous Parallel Computation. *Sagamore Computer Conference 1974, LNCS 24*, 1975. pp. 102–112. doi:10.1007/3-540-07135-0\_113.
- [30] Badouel E, Bernardinello L, Darondeau P. Petri Net Synthesis. Springer-Verlag, 2015. ISBN 978-3-662-47966-7. doi:10.1007/978-3-662-47967-4.
- [31] Best E, Devillers R. Synthesis of Persistent Systems. *Application and Theory of Petri Nets and Concurrency - 35th International Conference, PETRI NETS 2014, Tunis, Tunisia, 2014. Proceedings*, 2014. pp. 111–129. doi:10.1007/978-3-319-07734-5\_7.
- [32] Best E, Devillers R. Synthesis of live and bounded persistent systems. *Fundamenta Informaticae*, 2015. **140**:39–59. doi:10.3233/FI-2015-1244.
- [33] Devillers R. Products of Transition Systems and Additions of Petri Nets. *Proc. 16th International Conference on Application of Concurrency to System Design (ACSD 2016) J. Desel and A. Yakovlev (eds)*, 2016. pp. 65–73. doi:10.1109/ACSD.2016.10.
- [34] Devillers R. Factorisation of transition systems. *Acta Informaticae*, 2018. **55**(4):339–362. doi:10.1007/s00236-017-0300-y.
- [35] Devillers R, Schlachter U. Factorisation of Petri Net Solvable Transition Systems. *Application and Theory of Petri Nets and Concurrency - 39th International Conference, PETRI NETS 2018, Bratislava, Slovakia, Proceedings*, 2018. pp. 82–98. doi:10.1007/978-3-319-91268-4\_5.
- [36] Ehrenfeucht A, Rozenberg G. Partial (Set) 2-Structures. Part I: Basic Notions and the Representation Problem. *Acta Informaticae*, 1990. **27**(4):315–342. doi:10.1007/BF00264611.
- [37] Ehrenfeucht A, Rozenberg G. A Characterization of Set Representable Labeled Partial 2-Structures Through Decompositions. *Acta Informaticae*, 1990. **28**(1):83–94. doi:10.1007/BF02983375.
- [38] Darondeau P. Equality of languages coincides with isomorphism of reachable state graphs for bounded and persistent Petri nets. *Inf. Process. Lett.*, 2005. **94**(6):241–245. doi:10.1016/j.ipl.2005.03.002.

- [39] Best E, Darondeau P. A decomposition theorem for finite persistent transition systems. *Acta Inf.*, 2009. **46**(3):237–254. doi:10.1007/s00236-009-0095-6.