Synthesis of Weighted Marked Graphs from Constrained Labelled Transition Systems

Raymond Devillers[1], Evgeny Erofeev*[2], and Thomas Hujsa(✉)[3]

[1] Département d'Informatique, Université Libre de Bruxelles, B-1050 Brussels, Belgium (rdevil@ulb.ac.be)
[2] Department of Computing Science, Carl von Ossietzky Universität Oldenburg, D-26111 Oldenburg, Germany (evgeny.erofeev@uni-oldenburg.de)
[3] ONERA/DTIS & Université Fédérale Toulouse Midi-Pyrénées, F-31055 Toulouse, France (thomas.hujsa@onera.fr)

Abstract. Recent studies investigated the problems of analyzing Petri nets and synthesizing them from labelled transition systems (LTS) with two letters (transitions) only. In this paper, we extend these works by providing new characterizations for the synthesis of two- and three-letter Weighted Marked Graphs (WMGs), a well-known and useful class of weighted Petri nets in which each place has at most one input and one output. In this study, we focus mainly on LTS forming a single circuit. Also, we develop a sufficient condition of WMG-solvability for an arbitrary number of letters. Finally, we show that this sufficient condition is not necessary in the case of LTS forming a single circuit with five letters.

Keywords: Weighted Petri net, marked graph, synthesis, labelled transition system, cycles, cyclic words, circular solvability.

1 Introduction

Petri nets form a highly expressive and intuitive operational model of discrete event systems, capturing the mechanisms of synchronization, conflict and concurrency. Many of their fundamental behavioral properties are decidable, allowing to model and analyze numerous artificial and natural systems. However, most interesting model checking problems are intractable, and the efficiency of synthesis algorithms varies widely depending on the constraints imposed on the desired solution. In this study, we focus on the Petri net synthesis problem from a labelled transition system (LTS), which consists in determining the existence of a Petri net whose reachability graph is isomorphic to the given LTS, and building such a Petri net solution when it exists.

In previous studies on analysis or synthesis, structural restrictions on nets encompassed *plain* nets (each weight equals 1; also called ordinary nets) [19], *homogeneous* nets (meaning that for each place p, all the output weights of p are

* Supported by DFG through grant Be 1267/16-1 ASYST

equal) [22, 18], *free-choice* nets (the net is homogeneous, and any two transitions sharing an input place have the same set of input places) [9, 22], choice-free nets (each place has at most one output transition) [21, 16], marked graphs (each place has at most one output transition and one input transition) [6, 20, 5, 12], join-free nets (each transition has at most one input place) [22, 8, 17, 18], etc.

More recently, another kind of restriction has been considered, limiting the number of different transition labels of the LTS [1, 2, 13, 14].

In this paper, we combine this restriction on the number of labels with the weighted marked graph (WMG) constraint. Moreover, we focus mainly on finite *circular LTS*, meaning strongly connected LTS that contain a unique *cycle*[4]. In this context, we investigate the *cyclic solvability* of a word w, meaning the existence of a Petri net solution to the finite circular LTS induced by the infinite *cyclic word* w^∞.

An important purpose of studying such constrained LTS is to better understand the relationship between LTS decompositions and their solvability by Petri nets. Indeed, the unsolvability of simple subgraphs of the given LTS, typically elementary paths (i.e. not containing any node twice) and cycles (i.e. closed paths, whose start and end states are equal), often induces simple conditions of unsolvability for the entire LTS, as highlighted in other works [1, 13, 3]. Moreover, cycles appear systematically in the reachability graph of live and/or reversible Petri nets [21], which are used to model various real-world applications, such as embedded systems [15].

Contributions. In this work, we study further the links between simple LTS structures and the reachability graph of WMG, as follows.
First, we provide a characterization of the 2-letter (i.e. binary) cyclic words solvable by a WMG (i.e. WMG-solvable), and extend the analysis to finite cyclic LTS. We also tackle the case of infinite cyclic LTS with 2 letters. Then, we provide a general sufficient condition for the WMG-solvability of a cyclic word with an arbitrary number of letters, using a decomposition into specific WMG-solvable binary cyclic subwords. We prove that the same sufficient condition becomes a characterization of WMG-solvability for a subclass of the 3-letter cyclic words. Finally, we show, with the help of a counter-example, that this sufficient condition is not necessary for cyclic words with 5 letters.

Organization of the paper. After recalling classical definitions, notations and properties in Section 2, we present the results of WMG-solvability for 2-letter words in Section 3. Then, in Section 4, we focus on circular LTS: we develop the

[4] A set A of k arcs in a LTS G defines a cycle of G if the elements of A can be ordered as a sequence $a_1 \dots a_k$ such that, for each $i \in \{1, \dots, k\}$, $a_i = (n_i, \ell_i, n_{i+1})$ and $n_{k+1} = n_1$, i.e. the i-th arc a_i goes from node n_i to node n_{i+1} until the first node n_1 is reached, closing the path. Cycles are also called circuits, circles and oriented cycles.

general sufficient condition of WMG-solvability for any number of letters; in the case of 3 letters, we provide the characterization for a subset of the circular LTS; and we exhibit the counter-example for 5 letters. Finally, Section 5 presents our conclusions and perspectives.

2 Classical Definitions, Notations and Properties

LTS, sequences and reachability. A *labelled transition system with initial state*, abbreviated *LTS*, is a quadruple $TS = (S, \to, T, \iota)$ where S is the set of *states*, T is the set of *labels*, $\to \subseteq (S \times T \times S)$ is the *transition relation*, and $\iota \in S$ is the *initial state*.
A label t is *enabled* at $s \in S$, written $s[t\rangle$, if $\exists s' \in S\colon (s, t, s') \in \to$, in which case s' is said to be *reachable* from s by the firing of t, and we write $s[t\rangle s'$. Generalizing to any (firing) sequences $\sigma \in T^*$, $s[\varepsilon\rangle$ and $s[\varepsilon\rangle s$ are always true; and $s[\sigma t\rangle s'$, i.e. σt is *enabled* from state s and leads to s', if there is some s'' with $s[\sigma\rangle s''$ and $s''[t\rangle s'$. A state s' is *reachable* from state s if $\exists \sigma \in T^*\colon s[\sigma\rangle s'$. The set of states reachable from s is denoted by $[s\rangle$.

Petri nets, reachability and languages . A (finite, place-transition) *weighted Petri net*, or *weighted net*, is a tuple $N = (P, T, W)$ where 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 $W\colon ((P \times T) \cup (T \times P)) \to \mathbb{N}$ giving the weight of each arc. A *Petri net system*, or *system*, is a tuple $\mathcal{S} = (N, M_0)$ where N is a net and M_0 is the *initial marking*, which is a mapping $M_0\colon P \to \mathbb{N}$ (hence a member of $\mathbb{N}^P$) indicating the initial number of *tokens* in each place. The *incidence matrix* C of the net is the integer $P \times T$-matrix with components $C(p, t) = W(t, p) - W(p, t)$.

A place $p \in P$ is *enabled by* a marking M if $M(p) \geq W(p, t)$ for every output transition t of p. A transition $t \in T$ is *enabled by* a marking M, denoted by $M[t\rangle$, if for all places $p \in P$, $M(p) \geq W(p, t)$. If t is enabled at M, then t can *occur* (or *fire*) in M, leading to the marking M' defined by $M'(p) = M(p) - W(p, t) + W(t, p)$; we note $M[t\rangle M'$. A marking M' is *reachable* from M if there is a sequence of firings leading from M to M'. The set of markings reachable from M is denoted by $[M\rangle$. The *reachability graph of* $\mathcal{S}$ is the labelled transition system $RG(\mathcal{S})$ with the set of vertices $[M_0\rangle$, the set of labels T, initial state M_0 and transitions $\{(M, t, M') \mid M, M' \in [M_0\rangle \land M[t\rangle M'\}$. A system $\mathcal{S}$ is *bounded* if $RG(\mathcal{S})$ is finite.

The language of a Petri net system $\mathcal{S}$ is the set $\mathcal{L}(\mathcal{S}) = \{\sigma \in T^* \mid M_0[\sigma\rangle\}$. These languages are prefix-closed, i.e., if $\sigma = \sigma'\sigma'' \in \mathcal{L}(\mathcal{S})$, then $\sigma' \in \mathcal{L}(\mathcal{S})$. For any language $L \subseteq T^*$, we denote by $PREF(L)$ the language formed by its prefixes.

Vectors. 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 com-

ponents is one (i.e. its components do not have a common non-unit factor). A *T-semiflow* ν of the net is a non-null T-vector 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 [21].
The *Parikh vector* $\mathbf{P}(\sigma)$ of a finite sequence σ of transitions is a T-vector counting the number of occurrences of each transition in σ, and the *support* of σ is the support of its Parikh vector, i.e. $supp(\sigma) = supp(\mathbf{P}(\sigma)) = \{t \in T \mid \mathbf{P}(\sigma)(t) > 0\}$.

Strong connectedness and cycles in LTS. The LTS is said *reversible* if, $\forall s \in [\iota\rangle$, we have $\iota \in [s\rangle$, i.e., it is always possible to go back to the initial state; reversibility implies the strong connectedness of the LTS.
A sequence $s[\sigma\rangle s'$ is called a *cycle*, or more precisely a *cycle at (or around) state* s, if $s = s'$. A non-empty cycle $s[\sigma\rangle s$ is called *small* if there is no non-empty cycle $s'[\sigma'\rangle s'$ in TS with $\mathbf{P}(\sigma') \lneqq \mathbf{P}(\sigma)$ (the definition of Parikh vectors extending readily to sequences over the set of labels T of the LTS).
A *circular LTS* is a finite, strongly connected LTS that contains a unique cycle; hence, it has the shape of an oriented circle. The circular LTS *induced by* a word $w = w_1 \ldots w_k$ is the LTS with initial state s_0 defined as $s_0[w_1\rangle s_1[w_2\rangle s_2 \ldots [w_k\rangle s_0$.

All notions defined for labelled transition systems apply to Petri nets through their reachability graphs.

Petri net subclasses. A net N is *plain* if no arc weight exceeds 1; *pure* if $\forall p \in P\colon (p^\bullet \cap {}^\bullet p) = \emptyset$, where $p^\bullet = \{t \in T \mid W(p,t)>0\}$ and ${}^\bullet p = \{t \in T \mid W(t,p)>0\}$; CF (*choice-free* [7,21]) or ON (place-output-nonbranching [3]) if $\forall p \in P\colon |p^\bullet| \leq 1$; a WMG (*weighted marked graph* [20]) if $|p^\bullet| \leq 1$ and $|{}^\bullet p| \leq 1$ for all places $p \in P$. The latter form a subclass of the choice-free nets; other subclasses are *marked graphs* [6], which are plain with $|p^\bullet| = 1$ and $|{}^\bullet p| = 1$ for each place $p \in P$, and *T-systems* [9], which are plain with $|p^\bullet| \leq 1$ and $|{}^\bullet p| \leq 1$ for each place $p \in P$.

Isomorphism and solvability. Two LTS $TS_1 = (S_1, \to_1, T, s_{01})$ and $TS_2 = (S_2, \to_2, T, s_{02})$ are isomorphic if there is a bijection $\zeta\colon S_1 \to S_2$ with $\zeta(s_{01}) = s_{02}$ and $(s,t,s') \in \to_1 \Leftrightarrow (\zeta(s), t, \zeta(s')) \in \to_2$, for all $s, s' \in S_1$.
If an LTS TS is isomorphic to $RG(\mathcal{S})$ where $\mathcal{S}$ is a system, we say that $\mathcal{S}$ *solves* TS. Solving a word $w = \ell_1 \ldots \ell_k$ amounts to solve the acyclic LTS defined by the single path $\iota[\ell_1\rangle s_1 \ldots [\ell_k\rangle s_k$. A finite word w is *cyclically solvable* if the circular LTS induced by w is solvable. A LTS is WMG-solvable if a WMG solves it.

Other classical notions. An LTS $TS = (S, \to, T, \iota)$ is *fully reachable* if $S = [\iota\rangle$. It is *forward deterministic* if $s[t\rangle s' \wedge s[t\rangle s'' \Rightarrow s' = s''$, and *backward deterministic* if $s'[t\rangle s \wedge s''[t\rangle s \Rightarrow s' = s''$.
A system $\mathcal{S}$ is *forward persistent* if, for any reachable markings M, M_1, M_2, $(M[a\rangle M_1 \wedge M[b\rangle M_2 \wedge a \neq b) \Rightarrow M_1[b\rangle M' \wedge M_2[a\rangle M'$ for a reachable marking M'; it is *backward persistent* if, for any reachable markings M, M_1, M_2,

$(M_1[a\rangle M \wedge M_2[b\rangle M \wedge a \neq b) \Rightarrow M'[b\rangle M_1 \wedge M'[a\rangle M_2$ for a reachable marking M'.

Next, we recall classical properties of Petri net reachability graphs.

Proposition 1 (Classical Petri net properties). *If $\mathcal{S}$ is a Petri net system:*
– *$RG(\mathcal{S})$ is a fully reachable LTS.*
– *$RG(\mathcal{S})$ is forward deterministic and backward deterministic.*

For the subclass of WMGs, we have the following dedicated properties, extracted from Proposition 4, Lemma 1, Theorem 2 and Lemma 2 in [12].

Proposition 2 (Properties of WMG). *If $\mathcal{S} = (N, M_0)$ is a WMG system:*
– *It is forward persistent and backward persistent.*
– *If N is connected and has a T-semiflow ν, then there is a unique minimal one π, with support T, and $\nu = k \cdot \pi$ for some positive integer k. Moreover, if there is a non-empty cycle in $RG(\mathcal{S})$, there is one with Parikh vector π in $RG(\mathcal{S})$ around each reachable marking and $RG(\mathcal{S})$ is reversible. If there is no cycle, all the paths starting from some state s and reaching some state s' have the same Parikh vector.*

To simplify our reasoning in the sequel, we introduce the following notation, which captures some of the behavioral properties satisfied by WMG (Propositions 1 and 2). We denote by

- **b** the set of properties: forward and backward deterministic, forward and backward persistent, totally reachable;
- **c** the property: there is a small cycle whose Parikh vector is prime with support T.

A synthesis procedure does not necessarily lead to a connected solution. However, the technique of decomposition into prime factors described in [10, 11] can always be applied first, so as to handle connected partial solutions and recombine them afterwards. Hence, in the following, we focus on connected WMG, without loss of generality. In the next section, we consider the synthesis problem of WMG with exactly two different labels.

3 Cyclic binary WMG Synthesis

In this section, we provide conditions for the WMG-solvability of 2-letters cyclic LTS. In Subsection 3.1, we investigate the WMG-solvability of a finite cyclic LTS, first when it is circular, then when it contains at least one cycle, which is the more general cyclic case. In Subsection 3.2, we investigate the WMG-solvability of an infinite cyclic binary LTS.

3.1 WMG-solvable finite cyclic binary systems

In this subsection, we first consider any circular LTS with only two different labels. Each such LTS is defined by a word $w \in \{a,b\}^*$, corresponding to the labels encountered by firing the circuit once from ι, leading back to ι. Changing the initial state in this LTS amounts to rotate w. Clearly, each such LTS satisfies property **b**, but is not always WMG (or even Petri net) solvable.

The next results consider circuit Petri nets as represented in Figure 1, where places are named following the direction of the arcs, e.g. $p_{a,b}$ is the output place of a and the input place of b.

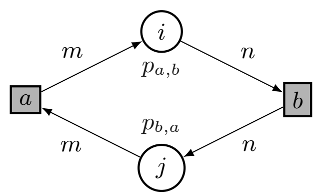

Fig. 1. A generic WMG solving a finite circular LTS induced by a word w over the alphabet $\{a,b\}$, whose initial marking (i,j) depends on the given solvable LTS. We assume that $\mathbf{P}(w) = (n,m)$ is prime.

Theorem 1 (Cyclically solvable binary words). *Consider a finite binary word w over*[5] *the alphabet $\{a,b\}$, with $(n,m) = \mathbf{P}(w)$ and $n \leq m$ (the case $m \leq n$ is handled symmetrically). Then, w is cyclically solvable if and only if $gcd(n,m) = 1$ and w is a rotation of the word $w' = ab^{m_0} \ldots ab^{m_{n-1}}$, where the sequence $m_0, \ldots, m_{n-1}$ is the sequence of quotients in the following system of equalities, with $r_0 = 0$:*

$$\begin{cases} r_0 + m = m_0.n + r_1, \text{ where } 0 \leq r_1 < n \\ r_1 + m = m_1.n + r_2, \text{ where } 0 \leq r_2 < n \\ \ldots \\ r_{n-1} + m = m_{n-1}.n. \end{cases}$$

Moreover, let $N = (P,T,W)$ be a circuit net as in Figure 1 with $P = \{p_{a,b}, p_{b,a}\}$, $T = \{a,b\}$, $W(p_{b,a},a) = W(a,p_{a,b}) = m$, and $W(p_{a,b},b) = W(b,p_{b,a}) = n$. Then, the system $\mathcal{S} = (N,M_0)$ with $M_0(p_{a,b}) = 0$ and $M_0(p_{b,a}) = m+n-1$, solves w cyclically.

Proof. From Proposition 2, for a connected WMG solution to exist, the Parikh vector of the word must be the minimal T-semiflow $\mu = (n,m)$ with support $T = \{a,b\}$, which is prime by definition, thus $gcd(m,n) = 1$.

[5] We consider only words that contain each letter of the alphabet.

A variant of this problem has been studied in [4] (section 6). Basing on this study, we derive the following. If a solution exists, then:
- there is a WMG solution as in Figure 1, implying that the sum $M_s(p_{a,b}) + M_s(p_{b,a})$, where $M_s(p)$ denotes the marking of place p when reaching state s, is the same for all states (since each firing preserves the number of tokens);
- all the markings reached in $p_{a,b}$ before reaching the initial state again are different, and similarly for $p_{b,a}$, since otherwise, two identical markings are reached at two different states, implying unsolvability (more precisely, the "separation property", which is necessary for synthesis, is not satisfied);
- w.l.o.g., there is a state s where $M_s(p_{a,b}) = 0$, and similarly for $p_{b,a}$ (otherwise, since the solution is pure, useless tokens can be removed);
- $M_s(p_{a,b})+M_s(p_{b,a}) = m+n-1$. Indeed, with more tokens, a reachable marking enables both a and b, which is not allowed by the given LTS; with fewer tokens, a deadlock is reached, i.e. a marking in which no transition is enabled;
- for each i, $m_i \in \{\lfloor m/n \rfloor, \lceil m/n \rceil\}$, there are $(m \bmod n)$ b-blocks of size $(\lfloor m/n \rfloor + 1)$, the other ones have size $\lfloor m/n \rfloor$.

Let us start from the state s such that $M_s(p_{a,b}) = 0$ and $M_s(p_{b,a}) = m + n - 1$, with $r_0 = 0$. We denote by r_i the number of tokens in $p_{a,b}$ at the $i+1$-th visited state that enables a. The value m_0 is the maximal number of b's that can be fired after the first a, and then r_1 tokens remain in $p_{a,b}$; hence, there are $m+n-1-r_1$ tokens in $p_{b,a}$ (which is at least m) before the second a. After the second a, we have $m + r_1$ tokens in $p_{a,b}$ and we fire m_1 b's. We iterate the process until the initial state is reached.

In the state enabling the $(i + 1)$-th a, there are $(i \cdot m) \mod n$ tokens in $p_{a,b}$, implying that r_n is again 0 when the initial state is reached. In between, we visited all the values from 0 to $n-1$ for the r_i's: indeed, if $(i \cdot m) \mod n = (j \cdot m) \mod n$ for $0 \leq i < j < n$, we have $((j-i) \cdot m) \mod n = 0$, or $((j-i) \cdot m = k \cdot n$ for some k; but then n must divide $j - i$ since m and n are relatively prime, which is only possible if $i = j$.

Finally, an adequate rotation of w' leads to w and to the corresponding value of r_0. □

An example is given in Figure 2, where the elements of the sequence $m_0, \ldots, m_{n-1}$ are put in bold in the system on the left.

Complexity. The number of operations to determine the sequence of m_i's is linear in the smallest weight n, i.e. also in the minimal number of occurrences of a letter. In comparison, the previous algorithm of [4] checks a quadratic number of subwords.

In Theorem 1, we provided a criterion for the cyclic solvability of a given word. In the next theorem, we abstract the word by a Parikh vector, which provides less accurate information on the behavior of the process. This result investigates the possible WMG-solvable LTS for this vector.

$$
\begin{aligned}
0 + 21 &= \mathbf{2}.8 + 5\\
5 + 21 &= \mathbf{3}.8 + 2\\
2 + 21 &= \mathbf{2}.8 + 7\\
7 + 21 &= \mathbf{3}.8 + 4\\
4 + 21 &= \mathbf{3}.8 + 1\\
1 + 21 &= \mathbf{2}.8 + 6\\
6 + 21 &= \mathbf{3}.8 + 3\\
3 + 21 &= \mathbf{3}.8 + 0.
\end{aligned}
$$

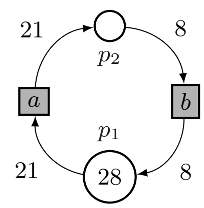

Fig. 2. This system solves the word $w = ab^2ab^3ab^2ab^3ab^3ab^2ab^3ab^3$ cyclically.

Theorem 2 (WMG-solvable cyclic binary systems). *Let us consider* $\mu = (n, m) \geq \mathbb{1}$ *such that* $\gcd(n, m) = 1$*, and a positive integer* k*. Up to isomorphism and the choice of the initial state, when* $k \geq n + m$*, there exists a single finite WMG-solvable LTS* $(S, \rightarrow, \{a, b\}, \iota)$ *that satisfies* **b**, **c** *and* $|S| = k$*, and that contains a small cycle whose Parikh vector is* μ*. No such WMG-solvable LTS exists when* $k < n+m$*. In the particular case of* $S = \{0, 1, \ldots, m+n-1\}$*, we have (up to isomorphism)* $\rightarrow = \{(i, a, i+m) | i, i+m \in S\} \cup \{(i, b, i-n) | i, i-n \in S\}$.

Proof. If a solution exists, it has the form of Fig. 1. If $k \geq n + m$, there are exactly $k - 1$ tokens in the system and the reachability graph is unique up to isomorphism. From the previous results of this section, if $M_0 = n + m - 1$, then the RG is circular and contains exactly $n + m$ distinct states: all the values for i between 0 and $n + m - 1$ are reached in some order. Moreover, if we identify the states to i, i.e., the marking of $p_{a,b}$, the arcs are $\{(i, a, i+m) | 0 \leq i, i+m < n + m \in S\} \cup \{(i, b, i-n) | 0 \leq i, i-n < n+m \in S\}$. As a consequence, if $|S| < n + m$, there aren't enough states to close the cycle, and there is no solution. The rest of the claim is deduced. □

3.2 WMG-solvability of infinite cyclic binary LTS

Let us now consider an infinite LTS satisfying **b** and **c** with only two different labels. From the discussion in the previous section, it cannot correspond to a net of the kind illustrated in Figure 1 since $i + j$ remains constant, hence yields finitely many states. On the other hand, a net of the kind illustrated in Figure 3 (or the variant obtained by switching the roles of a and b) yields infinitely many occurrences of transition a, leading to infinitely many different reachable markings. Besides, from any state, there may only be finitely many consecutive b's. Moreover, this is the only way to obtain infinitely many cycles with Parikh vector (n, m).

If $n = 1$, i is the maximum number of consecutive executions of b from ι; we can then verify if the given LTS corresponds to the constructed net. Otherwise, let k and l be the Bezout coefficients corresponding to the relatively prime numbers m and n, so that $k \cdot m + l \cdot n = 1$. If $l \geq 0 \geq k$, i is the maximum number of

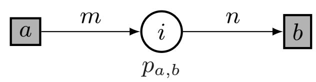

Fig. 3. A WMG solution for the infinite cyclic case.

times we may execute $a^{-k}b^l$ consecutively from ι, and we can check again if the given LTS corresponds to the constructed net (this is a direct generalization of the case $n = 1$). Otherwise, since $-n \cdot m + m \cdot n = 0$, by adding this relation enough times to the previous one, we get $k' \cdot m + l' \cdot n = 1$ with $l' \geq 0 \geq k'$, and we apply the same idea.

4 Finite words being cyclically solvable with a WMG

In this section, we provide several conditions of WMG-solvability for LTS formed of a single circuit, based on the WMG-solvability of specific binary subwords.

We consider first the case of an arbitrary number of letters, providing a general sufficient condition of solvability by decomposition. Then, we develop a characterization of WMG-solvability for a subclass of the cyclic 3-letter words. Finally, a counter-example to this characterization is obtained for 5 letters.

4.1 Sufficient condition for the solvability of cyclic words

The next theorem provides a sufficient condition for cyclic solvability of k-ary words, for any positive integer k. This condition uses binary subwords obtained by projection[6] and containing occurrences of two different labels that are contiguous in the k-ary word. The other binary subwords are not needed since they lack this contiguity and do not capture the direct causality.

Theorem 3. *Consider any word w over[7] any finite alphabet T such that $\mathbf{P}(w)$ is prime. Suppose the following: $\forall u = w_{|t_1 t_2}$ (i.e., the projection of w on $\{t_1, t_2\}$) for some t_1, t_2 such that $t_1 \neq t_2 \in T$, and $w = (w_1 t_1 t_2 w_2)$ or $w = (t_2 w_3 t_1)$, $u = v^\ell$ for some positive integer ℓ, $\mathbf{P}(v)$ is prime, and v is cyclically solvable by a circuit. Then, w is cyclically solvable with a WMG.*

Proof. Consider, for every such pair (t_i, t_j), $t_i \neq t_j$, a circuit solution $C_{i,j}$ of v for the subword $v^l = u_{i,j} = w_{|t_i t_j}$, obtained as in the construction of Theorem

[6] The projection of a word $w \in A^*$ on a set $A' \subseteq A$ of letters is the maximum subword of w whose letters belong to A', noted $w_{|A'}$. For example, the projection of the word $w = \ell_1\, \ell_2\, \ell_3\, \ell_2$ on the set $\{\ell_1, \ell_2\}$ is the word $\ell_1\, \ell_2\, \ell_2$.

[7] We consider only words that contain each letter of the alphabet.

1. Assuming all these nets are place-disjoint, consider the transition-merging[8] of all these marked circuits $C = \{C_1, \ldots C_q\}$, $C_i = ((P_i, T_i, W_i), M_i)$ for $i = 1 \ldots q$, through a WMG $\mathcal{S}'$, i.e. $\mathcal{S}' = (N', M_0')$ such that $N' = (P', T, W')$ with $P' = \uplus_{i=1\ldots q} P_i$, $T = \cup_{i=1\ldots q} T_i$, $W' = \cup_{i=1\ldots q} W_i$, and M_0' is the concatenation $M_1 \ldots M_q$.

Let w be of the form aw'. We prove that a is the only transition enabled in $\mathcal{S}'$.

All the subwords of the form $w_{|a,t}$ necessarily start with a. All the input places of the transition a belong to the binary circuits defined by these subwords. Since these subwords are solvable by marked circuits which we merged together, all the input places of a are initially enabled. Now, let us suppose that another transition d is also initially enabled in $\mathcal{S}'$. Since d is not the first letter of w, another letter q appears in w just before the first occurrence of d. In the solution of $w_{|d,q}$, d is not initially enabled since q must occur before; hence it is not enabled in the merging either. We deduce that a is the only transition that is enabled in S'.

Now, the same arguments apply to $w'' = w'a$ whose relevant subwords are solvable by the circuits in the same way, and we deduce that the WMG $\mathcal{S}'$ has the language $PREF(w^*)$.

Note that we did not use explicitely above the special form of u. Simply, the latter is necessary to build a circuit system $C_{i,j}$ with the language $PREF(u^*) = PREF(v^*)$. $C_{i,j}$ is a circular solution for v, but not for u unless $\ell = 1$. The fact that the merging $\mathcal{S}'$ of all the $C_{i,j}$'s yields not only a system with the adequate language $PREF(w^*)$ but a circular solution of w arises from the fact that $\mathbf{P}(w)$ is prime (by Proposition 2). We thus deduce that the WMG $\mathcal{S}'$ solves w cyclically. □

4.2 WMG-solvability for a subclass of ternary cyclic words

Next, we prove the other direction of Theorem 3, leading to a characterization of WMG-solvability for a special subclass of the ternary cyclic words.

The proof exploits a WMG with 3 transitions and 6 places, connecting 2 places to each pair of transitions, as illustrated in Figure 4. In some cases, a smaller number of places can solve the same LTS, but we do not aim here at minimizing the number of nodes in a solution.

Theorem 4 (Cyclic solvability of ternary words). *Consider a ternary word w over[9] the alphabet T with Parikh vector (x, x, y) such that $gcd(x, y) = 1$. Then, w is cyclically solvable with a WMG if and only if $\forall u = w_{|t_1 t_2}$ such that $t_1 \neq t_2 \in T$, and $w = (w_1 t_1 t_2 w_2)$ or $w = (t_2 w_3 t_1)$, $u = v^\ell$ for some positive integer ℓ, $\mathbf{P}(v)$ is prime, and v is cyclically solvable by a circuit [10].*

[8] Also called sometimes the synchronization on transitions.
[9] We consider only words that contain each letter of the alphabet.
[10] also called circular Petri net.

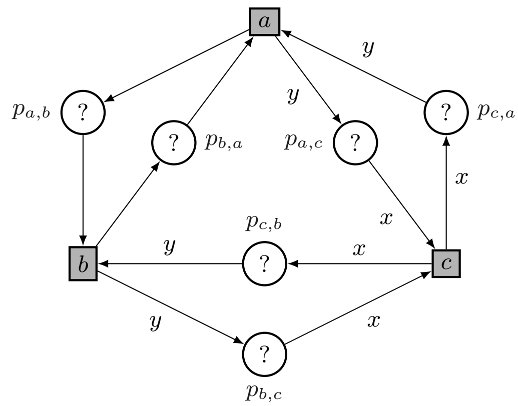

Fig. 4. A WMG with three transitions (letters), with minimal T-semiflow (x, x, y) and $\gcd(x, y) = 1$.

Proof. The right-to-left direction of the equivalence, assuming the properties on the projections, is true by Theorem 3, for the particular case that $|T| = 3$. We thus deduce the cyclic solvability.

In the rest of this proof, we consider the other direction, assuming circular solvability. If $x = y = 1$, the claim is trivially obtained. Let us assume that $x \neq y$.

Let us write $T = \{a, b, c\}$. The general form of a solution has 3 transitions and 6 places (one for each ordered pair of transitions). Additional places are never necessary in the presence of a T-semiflow. Indeed, let $p_{u,v}$ be a place between transitions u and v, W_u the weight on the arc to this place and W_v the one from this place. Due to the presence of the T-semiflow $\mathbf{P}(w)$, we have $\mathbf{P}(w)(u) \cdot W_u = \mathbf{P}(w)(v) \cdot W_v$, and we may choose $W_u = \mathbf{P}(w)(v)$ as well as $W_v = \mathbf{P}(w)(u)$. We may also divide the weights around each place by their gcd. In our case, this leads to the configuration illustrated by Figure 4. We denote by RG the reachability graph of a solution based on this net.

We show first that the projection $w_{\restriction ab}$ of w on $\{a, b\}$ is of the form $(ab)^k$ or $(ba)^k$ for some positive integer k.

There is no pattern aab in w^2 (which allows to consider sequences on the border of two consecutive w's) because, if $M_1[a\rangle M_2[a\rangle M_3[b\rangle$, $M_1(p_{c,b}) = M_2(p_{c,b}) = M_3(p_{c,b}) \geq y$ and $M_2(p_{a,b}) \geq 1$, which would also allow to perform b after the first a and RG is not circular.

If there is a pattern aac and $M_1[a\rangle M_2[a\rangle M_3[c\rangle M_4$, $M_2(p_{a,c}) \geq y$ and $M_1(p_{b,c}) = M_2(p_{b,c}) = M_3(p_{b,c}) \geq x$, hence $y < x$ otherwise M_2 already enables c and RG is not circular. Then $M_4(p_{a,b}) \geq 2$, $M_4(p_{c,b}) \geq x > y$ and $M_4(p_{c,a}) \geq x > y$, so that $M_4[b\rangle$; hence $M_4(p_{b,a}) = 0$ since otherwise we also have $M_4[a\rangle$ and RG is not circular. We thus have $M_4[ba\rangle M_5$ for some marking M_5, with $M_5(p_{b,a}) = 0$, so that M_5 does not enable a; M_5 does not enable b either since otherwise we could also do $M_4[bb\rangle$ and again RG is not circular. Thus, we have $M_4[bac\rangle$, and then we are in a situation similar to the one after the first c. As a consequence, we must have a sequence $M_1[aa(cba)^\omega\rangle$, and RG is not circular.

Hence, in w^2 we cannot have a sequence aa, nor bb by symmetry.

Let us now assume that a pattern $ac^k a$ exists in w^2 for some $k \geq 1$. Since the first firing of a puts a token in $p_{a,b}$ and the next firing of c does not enable b, we must have $x < y$. Let us assume in the circular RG that $M_1[ac^k a\rangle M_2[\sigma\rangle M_1$. σ is not empty since it must contain x times b. It cannot end with an a, since otherwise we have a sequence aa, which we already excluded. It cannot end with a b either, since $M_1(p_{b,a}) \geq 2$ (in order to fire a twice without a b in between), so that if $M_3[b\rangle M_1[a\rangle$, $M_3(p_{b,a}) \geq 1$, we must also have $M_3[a\rangle$, and RG is not circular. Hence, σ ends with a c and for some markings M_2', M_3 we have $M_3[c\rangle M_1[ac^k\rangle M_2'[a\rangle M_2$.

We deduce that $M_2'(p_{a,b}) \geq 1$, $M_2'(p_{c,b}) \geq (k+1) \cdot x$; hence $(k+1) \cdot x < y$ otherwise M_2' also enables b and RG is not circular. Also, $M_3(p_{b,a}) \geq 2$ and $M_3(p_{c,a}) \geq 2 \cdot y - (k+1) \cdot x > y$, so that M_3 also enables a and again RG is not circular. As a consequence, we cannot have a pattern $ac^k a$, nor $bc^k b$ by symmetry, and $w_{|ab} = (ab)^k$ or $w_{|ab} = (ba)^k$ for the positive integer $k = x$. With $v = ab$ or $v = ba$, we have the adequate solvability property for $w_{|ab}$, and we can assume in the following that the sum of the tokens present in places $p_{a,b}$ and $p_{b,a}$ is 1 for all reachable markings.

Let us now suppose that we have a WMG $\mathcal{S}$ solving w cyclically, whose underlying net is pictured in Figure 4. From the previous results, we can assume that $M_0(p_{a,b}) + M_0(p_{b,a}) = 1$ in $\mathcal{S}$, this equality being preserved by all reachable markings. To show that $u = w_{|ac}$ has the adequate form (the case for $w_{|bc}$ is symmetrical), let us consider the circuit C_{ac}, restriction of $\mathcal{S}$ to $p_{a,c}, c, p_{c,a}, a$.

Let us assume in the following that u cannot be written under the form $u = v^\ell$ for some positive integer ℓ, where $\mathbf{P}(v)$ is prime and cyclically solvable. Since $gcd(x, y) = gcd(\mathbf{P}(w)(a), \mathbf{P}(w)(c)) = gcd(\mathbf{P}(u)(a), \mathbf{P}(u)(c)) = 1$, $\mathbf{P}(u)$ is prime and $u = v$ with $\ell = 1$, hence u is not cyclically solvable. For the net N considered, this implies the existence of some prefix σ_{ac} of u such that, for every initial marking of C_{ac} that enables the sequence u in this circuit, the marking reached by firing σ_{ac} necessarily enables both places $p_{a,c}$ and $p_{c,a}$. Indeed, Theorem 1 specifies the finite set of all possible minimal markings that allow cyclic solvability, and each such marking enables exactly one place of the circuit. Every other non-circular reachability graph is defined by some larger initial marking and contains a marking that enables both places.
Thus, for any initial marking M_0 that makes the system $\mathcal{S} = (N, M_0)$ solve w cyclically, the smallest prefix of w whose projection on $\{a, c\}$ equals σ_{ac} leads to a marking M in the WMG that enables $p_{a,c}$ and $p_{c,a}$.

Hereafter, we consider all the cases in which either a or c is enabled from M. In each case, we describe the shape of the LTS and deduce from it a reachable marking that enables two transitions, hence a contradiction.

Case $x > y$: In this case, in $\mathcal{S}$, we cannot have two consecutive c's.

– Subcase in which M enables the place $p_{a,c}$ as well as the transition a in the WMG, hence its input places $p_{b,a}$ and $p_{c,a}$. Since $M[a\rangle$, transition c is not enabled at M, implying that $p_{b,c}$ is not enabled by M. We deduce: $M(p_{a,c}) > M(p_{b,c})$. Since $p_{a,c}$ is enabled by M, the last occurrence of a transition before the next firing of c is necessarily b, implying: $M[(ab)^k c\rangle M_1$ for some integer $k \geq 1$ and some marking M_1. The inequality mentioned above is still valid at M_1, i.e. $M_1(p_{a,c}) > M_1(p_{b,c})$, and we iterate the same arguments from M_1 to deduce that the rotation w_M of w starting at M is of the form $(ab)^{k_1}c \ldots (ab)^{k_y}c$ with $\sum_{i=1..y} k_i = x$ and each k_i is positive.

– Subcase in which M enables the place $p_{c,a}$ as well as the transition c in the WMG, hence its input places $p_{a,c}$ and $p_{b,c}$. Thus, the firing of c from M cannot enable a, implying that $M(p_{c,b}) < M(p_{c,a})$ and that $M[c(ba)^k c\rangle M_1$ for some positive integer k and a marking M_1. The inequality is still valid at M_1, i.e. $M_1(p_{c,b}) < M_1(p_{c,a})$, from which we deduce that the rotation w_M of w starting at M is of the form $c(ba)^{k_1} \ldots c(ba)^{k_y}$ with $\sum_{i=1..y} k_i = x$ and each k_i is positive.

Case $x \leq y$:

– Subcase in which M enables the place $p_{a,c}$ as well as the transition a in the WMG, hence its input places $p_{c,a}$ and $p_{b,a}$. Thus, the firing of a from M cannot enable c, implying that $M(p_{b,c}) < M(p_{a,c})$ and that $M[abc^k\rangle M_1$ for some positive integer k and a marking M_1, at which the same inequality is still valid. We deduce that the rotation w_M of w starting at M is of the form $abc^{k_1} \ldots abc^{k_x}$ with $\sum_{i=1..x} k_i = y$ and each k_i is positive.

– Subcase in which M enables the place $p_{c,a}$ as well as the transition c in the WMG, hence its input places $p_{a,c}$ and $p_{b,c}$. Thus, firing one or several c's from M does not enable a, and $M(p_{c,b}) < M(p_{c,a})$, implying that $M[c^k ba\rangle M_1$ for some positive integer k and a marking M_1, at which the same inequality is still valid. We deduce that the rotation w_M of w starting at M is of the form $c^{k_1}ba \ldots c^{k_x}ba$ with $\sum_{i=1..x} k_i = y$ and each k_i is positive.

In each of the four cases developed above, we observe that each sequence of ab or ba could be seen as an atomic firing, and $w_{M\,|b,c}$ is obtained from $w_{M\,|a,c}$ by renaming each a into one b. This implies that the deletion of the initial useless tokens (also known as frozen tokens, i.e. never used by any firing) yields a system in which some reachable marking distributes the tokens in the same way in the places between c and a as in the places between c and b. This is for example the case of the marking M if it does not contain useless tokens.
We deduce that M (with or without useless tokens) enables all four places $p_{a,c}$, $p_{c,a}$, $p_{b,c}$ and $p_{c,b}$, thus enabling two transitions of the WMG at least. This contradicts the cyclic solvability of w, implying that $v = u$ is cyclically solvable by a circuit. Hence the claim. $\square$

4.3 A counter-example for 5 letters

In Theorem 4, we provided a characterization of cyclic WMG-solvability for ternary words w such that $\mathbf{P}(w)$ is prime and has only two values. However, this results does not apply to words w over 5 letters such that $\mathbf{P}(w)$ is prime and has only two values, as pictured in Figure 5. This WMG cyclically solves the word $w = aacbbeabd$ with $\mathbf{P}(w) = (3,3,1,1,1)$, which is prime, while its projection $u = aabbab$ on $\{a,b\}$ leads to $v = u$, and $\mathbf{P}(v) = (3,3)$ is not prime, hence is not cyclically solvable by a WMG.

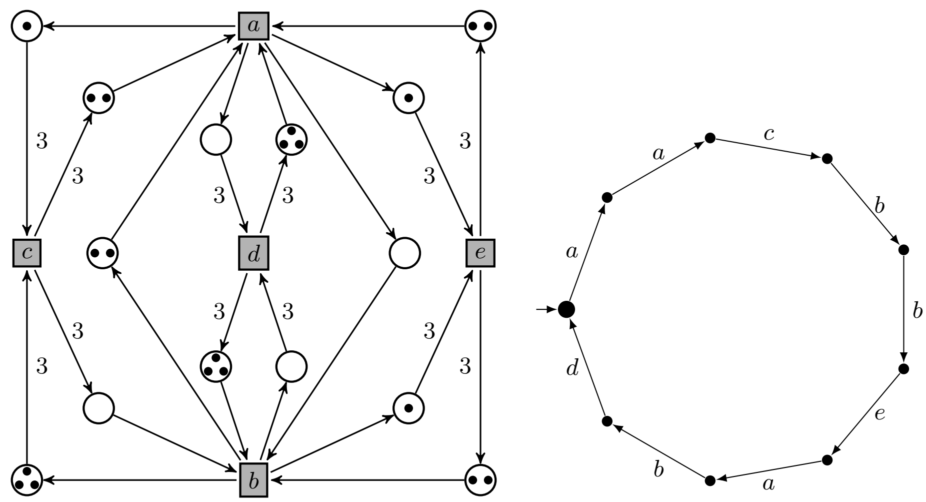

Fig. 5. *aacbbeabd* is cyclically WMG-solvable.

5 Conclusions and Perspectives

In this work, we generalized previous methods dedicated to the analysis and synthesis of weighted marked graphs, a well-known and useful subclass of weighted Petri nets allowing to model various real-world applications.

By restricting the size of the alphabet to 2 letters, we provided a characterization of the WMG-solvable labelled transition systems formed of a single circuit. We also extended this investigation to infinite LTS.

For the case of an LTS formed of a single circuit with an arbitrary number of letters, we proposed a sufficient condition of WMG-solvability.

Then, for the case of an LTS formed of a single circuit with 3 letters, we obtained a characterization of WMG-solvability for a subset of the possible Parikh vectors.

Finally, we proved that this condition for 3 letters does not extend to circular LTS using a 5-letter alphabet.

A general perspective is to characterize all the WMG-solvable LTS, which are not necessarily formed of a single circuit. To this end, developing new conditions based on decompositions into subwords, by weakening the results presented in this paper, would be worth investigating.

Acknowledgements

We would like to thank the anonymous referees for their useful suggestions.

References

1. Barylska, K., Best, E., Erofeev, E., Mikulski, L., Piatkowski, M.: On binary words being Petri net solvable. In: Proceedings of the International Workshop on Algorithms & Theories for the Analysis of Event Data, ATAED 2015, Brussels, Belgium. pp. 1–15 (2015), http://ceur-ws.org/Vol-1371/paper01.pdf
2. Barylska, K., Best, E., Erofeev, E., Mikulski, L., Piatkowski, M.: Conditions for Petri net solvable binary words. T. Petri Nets and Other Models of Concurrency 11, 137–159 (2016)
3. Best, E., Devillers, R.: Synthesis and reengineering of persistent systems. Acta Inf. 52(1), 35–60 (2015), http://dx.doi.org/10.1007/s00236-014-0209-7
4. Best, E., Erofeev, E., Schlachter, U., Wimmel, H.: Characterising petri net solvable binary words. In: Kordon, F., Moldt, D. (eds.) Application and Theory of Petri Nets and Concurrency. pp. 39–58. Springer International Publishing, Cham (2016)
5. Best, E., Hujsa, T., Wimmel, H.: Sufficient conditions for the marked graph realisability of labelled transition systems. Theoretical Computer Science (2017), http://www.sciencedirect.com/science/article/pii/S0304397517307181
6. Commoner, F., Holt, A., Even, S., Pnueli, A.: Marked directed graphs. J. Comput. Syst. Sci 5(5), 511–523 (1971), https://doi.org/10.1016/S0022-0000(71)80013-2
7. Crespi-Reghizzi, S., Mandrioli, D.: A decidability theorem for a class of vector-addition systems. Inf. Process. Lett. 3(3), 78–80 (1975), http://dx.doi.org/10.1016/0020-0190(75)90020-4
8. Delosme, J.M., 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. pp. 90–99 (July 2013)
9. Desel, J., Esparza, J.: Free Choice Petri Nets, Cambridge Tracts in Theoretical Computer Science, vol. 40. Cambridge University Press, New York, USA (1995)
10. Devillers, R.: Products of Transition Systems and Additions of Petri Nets. In: Proc. 16th International Conference on Application of Concurrency to System Design (ACSD 2016) J. Desel and A. Yakovlev (eds). pp. 65–73 (2016), https://doi.org/10.1109/ACSD.2016.10
11. Devillers, R.: Factorisation of transition systems. Acta Informatica (2017), https://doi.org/10.1007/s00236-017-0300-y

12. Devillers, R., Hujsa, T.: Analysis and synthesis of weighted marked graph Petri nets. In: Khomenko, V., Roux, O.H. (eds.) Application and Theory of Petri Nets and Concurrency: 39th International Conference, PETRI NETS 2018, Bratislava, Slovakia, 2018, Proceedings. pp. 1–21. Springer International Publishing (2018)
13. Erofeev, E., Barylska, K., Mikulski, L., Piatkowski, M.: Generating all minimal Petri net unsolvable binary words. In: Proceedings of the Prague Stringology Conference 2016, Prague, Czech Republic. pp. 33–46 (2016), http://www.stringology.org/event/2016/p04.html
14. Erofeev, E., Wimmel, H.: Reachability graphs of two-transition Petri nets. In: Proceedings of the International Workshop on Algorithms & Theories for the Analysis of Event Data 2017, Zaragoza, Spain. pp. 39–54 (2017), http://ceur-ws.org/Vol-1847/paper03.pdf
15. Hujsa, T.: Contribution to the study of weighted Petri nets. Ph.D. thesis, Pierre and Marie Curie University, Paris, France (2014), https://tel.archives-ouvertes.fr/tel-01127406
16. Hujsa, T., Delosme, J.M., Munier-Kordon, A.: On the reversibility of well-behaved weighted choice-free systems. In: Ciardo, G., Kindler, E. (eds.) Application and Theory of Petri Nets and Concurrency. pp. 334–353. Springer International Publishing, Cham (2014)
17. Hujsa, T., Delosme, J.M., Munier-Kordon, A.: Polynomial sufficient conditions of well-behavedness and home markings in subclasses of weighted Petri nets. ACM Trans. Embed. Comput. Syst. 13(4s), 141:1–141:25 (Jul 2014), http://doi.acm.org/10.1145/2627349
18. Hujsa, T., Devillers, R.: On liveness and deadlockability in subclasses of weighted Petri nets. In: van der Aalst, W., Best, E. (eds.) Application and Theory of Petri Nets and Concurrency: 38th International Conference, PETRI NETS 2017, Zaragoza, Spain, June 25–30, 2017, Proceedings. pp. 267–287. Springer International Publishing, Cham (2017), https://doi.org/10.1007/978-3-319-57861-3_16
19. Murata, T.: Petri nets: properties, analysis and applications. Proceedings of the IEEE 77(4), 541–580 (April 1989)
20. Teruel, E., Chrzastowski-Wachtel, P., Colom, J.M., Silva, M.: On weighted T-systems. In: Jensen, K. (ed.) 13th International Conference on Application and Theory of Petri Nets and Concurrency (ICATPN), LNCS. vol. 616, pp. 348–367. Springer, Berlin, Heidelberg (1992), https://doi.org//10.1007/3-540-55676-1_20
21. Teruel, E., Colom, J.M., 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 27(1), 73–83 (1997), http://dx.doi.org/10.1109/3468.553226
22. Teruel, E., Silva, M.: Structure theory of Equal Conflict systems. Theoretical Computer Science 153(1&2), 271–300 (1996), https://doi.org/10.1016/0304-3975(95)00124-7