Soundness of Resource-Constrained Workflow Nets

Kees van Hee, Alexander Serebrenik, Natalia Sidorova, and Marc Voorhoeve

Department of Mathematics and Computer Science
Eindhoven University of Technology
P.O. Box 513, 5600 MB Eindhoven, The Netherlands
{k.m.v.hee, a.serebrenik, n.sidorova, m.voorhoeve}@tue.nl

Abstract. We study concurrent processes modelled as workflow Petri nets extended with resource constraints. We define a behavioural correctness criterion called *soundness*: given a sufficient initial number of resources, all cases in the net are guaranteed to terminate successfully, no matter which schedule is used. We give a necessary and sufficient condition for soundness and an algorithm that checks it.
Keywords: Petri nets; concurrency; workflow; resources; verification.

1 Introduction

In systems engineering, coordination plays an important role on various levels. Workflow management systems coordinate the activities of human workers; the principles underlying them can also be applied to other software systems, like middleware and web services. Petri nets are well suited for modelling and verification of concurrent systems; for that reason they have proven to be a successful formalism for Workflow systems (see e.g. [1–4]).

Workflow systems are modelled by so-called *Workflow Nets (WF-nets)* [1], i.e. Petri nets with one initial and one final place and every place or transition being on a directed path from the initial to the final place. The execution of a *case* is represented as a firing sequence that starts from the initial marking consisting of a single token on the initial place. The token on the final place with no garbage (tokens) left on the other places indicates the *proper termination* of the case execution. A model is called *sound* iff every reachable marking can terminate properly.

WF-nets are models emphasising the partial ordering of activities in the process and abstracting from *resources*, e.g. machines, manpower, or money, which may further restrict the occurrence of activities. In this paper we consider the influence of *resources* on the processing of cases in Workflow Nets. We consider here only durable resources, i.e. resources that can be claimed and released during the execution, but not created or destroyed. We introduce the notion of the Resource-Constrained Workflow net (RCWF-net), which is a workflow net consisting of a production (sub)net — a workflow net where resources are abstracted away, and a number of resource places restricting the functionality of the production net.

The notion of soundness is adapted to the nature of RCWF-nets: we say that an RCWF-nets with a given number of resources is sound iff for an arbitrary number of cases processed by the net, all cases can terminate properly, whatever choices are made during the execution and all resources are returned to their places. We will say that an RCWF-net is sound iff there exists a resource marking such that the RCWF-net with this and any larger resource marking is sound. This definition is very natural, especially in the area of business processes, since we would like to have a system specification such that buying new machines or obtaining additional financial resources would not require to reconsider the specification, lest the system become unreliable.

In many practical applications, cases processed in the Workflow net are independent of each other, which can be modelled by introducing simple colours for the tokens going through the production net. We build a transition system corresponding to the work of the production net with a single initial token, extending this transition system with the information about consumptions and releases of resources for every transition in it. Then we represent this transition system as a state machine, that can be considered as a model of the production net where the colours of tokens can be removed without influencing the system behaviour, and finally we extend this state machine up to the RCWF-net by adding resource places according to information about resource consumptions/releases that we have for every transition of the net. Thus our task of checking the correctness of arbitrary RCWF-nets is reduced to checking the correctness of RCWF-nets whose production nets are state machines.

In this paper we consider only RCWF-nets with one resource type, which is sufficient for many practical applications. We give a necessary and sufficient condition of soundness for the nets of this class and give a decision algorithm with a polynomial complexity w.r.t. the size of the state machine describing the behaviour of the production net.

Related work The problem of the correct functioning of parallel processes that share resources is not new at all. The famous banker's algorithm of Dijkstra (cf. [8]) is one of the oldest papers on this topic. The problem of the banker's algorithm is different from ours, because in the bankers algorithm a schedule (i.e. an ordering of processes for granting their resource claims) is designed. It is a pessimistic approach because it assumes that each process will eventually claim its maximal need for resources, a number that has to be known in advance. In our situation a process does not have to claim the maximal possible number of resources it might claim, and therefore the pessimistic scheduling is too restrictive. Another important difference is that we do not consider a scheduling strategy at all: we look for conditions such that a workflow engine can execute tasks (i.e. fire transitions) as soon as all preliminary work has been done, if there are enough resources available. So the workflow engine may assign resources considering the local state only. This means that if the processes are designed properly, a standard workflow engine can be used to execute the process in a sound way.

The problem of resource sharing in flexible manufacturing systems have been studied extensively, specifically by modelling them as Petri nets (see [14, 13, 10, 6, 9] for an overview of works in this field). In these works the authors focus on extending a model that represents the production process with a scheduler in order to avoid deadlocks and to use resources in the most efficient way. As mentioned before, our goal is to allow the workflow engine to execute processes without further scheduling. Therefore we concentrate here on fundamental correctness requirements for RCWF-nets: resource conservation laws (every claimed resource is freed before the case terminates and no resource is created) and no deadlocks or livelocks that occur due to the lack of resources.

In [5] the authors consider structural analysis of Workflow nets with shared resources. Their definition of structural soundness corresponds approximately to the existence of k cases and R resource tokens such that the net is sound for this k and R. We consider systems where a number of cases with id's are going through the net and the number of available resources can vary, and so we require that the system should work correctly for any number of cases and resources. Therefore the results of [5] are not applicable to our case.

The rest of the paper is organised as follows. In Section 2, we sketch the basic definitions related to Petri nets and Workflow nets. In Section 3 we introduce the notion of Resource-Constrained Workflow Nets and define the notion of soundness for RCWF-nets. In Section 4 we give a necessary and sufficient condition of soundness and in Section 5 we give a decision algorithm for soundness. We conclude in Section 6 with discussion of the obtained results and directions for future work.

2 Preliminaries

$\mathbb{N}$ denotes the set of natural numbers and $\mathbb{Q}$ the set of rational numbers.

Let P be a set. A *bag (multiset)* m over P is a mapping $m : P \to \mathbb{N}$. The set of all bags over P is $\mathbb{N}^P$. We use $+$ and $-$ for the sum and the difference of two bags and $=, <, >, \leq, \geq$ for comparisons of bags, which are defined in a standard way. We overload the set notation, writing $\emptyset$ for the empty bag and $\in$ for the element inclusion. We write $m = 2[p] + [q]$ for a bag m with $m(p) = 2$, $m(q) = 1$, and $m(x) = 0$ for all $x \notin \{p, q\}$. As usual, $|m|$ stands for the number of elements in bag m.

For (finite) *sequences* of elements over a set T we use the following notation: The empty sequence is denoted with ϵ; a non-empty sequence can be given by listing its elements. A concatenation of sequences σ_1 and σ_2 is denoted with $\sigma_1\sigma_2$; $t\sigma$ and σt stand for the concatenation of t and sequence σ and vice versa, and σ^n for the concatenation of n sequences σ.

Transition Systems A *transition system* is a tuple $E = \langle S, Act, T\rangle$ where S is a set of *states*, Act is a finite set of *action names* and $T \subseteq S \times Act \times S$ is a *transition relation.* A *process* is a pair (E, s_0) where E is a transition system and $s_0 \in S$ an initial state. We denote $(s_1, a, s_2) \in T$ as $s_1 \xrightarrow{a}_E s_2$, and we say that a leads from s_1 to s_2 in E. We omit E and write $s \xrightarrow{a} s'$ whenever no ambiguity can arise. For a sequence of transitions $\sigma = (t_1, \ldots, t_n)$ we write $s_1 \xrightarrow{\sigma} s_2$ when $s_1 = s^0 \xrightarrow{t_1} s^1 \xrightarrow{t_2} \ldots \xrightarrow{t_n} s^n = s_2$. In this case we say that σ is a trace of E. Finally, $s_1 \xrightarrow{*} s_2$ means that there exists a sequence $\sigma \in T^*$ such that $s_1 \xrightarrow{\sigma} s_2$. We say that s_2 is *reachable* from s_2 iff $s_1 \xrightarrow{*} s_2$.

Petri nets A *Petri net* is a tuple $N = \langle P, T, F^+, F^- \rangle$, where:

- P and T are two disjoint non-empty finite sets of *places* and *transitions* respectively, the set $P \cup T$ are the *nodes* of N;
- F^+ and F^- are mappings $(P \times T) \to \mathbb{N}$ that are *flow functions* from transitions to places and from places to transitions respectively.

$F = F^+ - F^-$ is the *incidence matrix* of net N.

We present nets with the usual graphical notation.

Given a transition $t \in T$, the *preset* ${}^\bullet t$ and the *postset* $t^\bullet$ of t are the *bags* of places where every $p \in P$ occurs $F^-(p,t)$ times in ${}^\bullet t$ and $F^+(p,t)$ times in $t^\bullet$. Analogously we write ${}^\bullet p, p^\bullet$ for pre- and postsets of places. We will say that a node n is a *source* node iff ${}^\bullet n = \emptyset$ and n is a *sink* node iff $n^\bullet = \emptyset$.

A marking m of N is a bag over P; markings are states (configurations) of a net. A pair (N, m) is called a *marked* Petri net. A transition $t \in T$ is *enabled* in marking m iff ${}^\bullet t \leq m$. An enabled transition t may *fire*. This results in a new marking m' defined by $m' \stackrel{\text{def}}{=} m - {}^\bullet t + t^\bullet$. We interpret a Petri net N as a transition system/process where markings play the role of states and firings of the enabled transitions define the transition relation, namely $m + {}^\bullet t \xrightarrow{t} m + t^\bullet$, for any $m \in \mathbb{N}^P$. The notion of reachability for Petri nets is inherited from the transition systems. We denote the set of all markings reachable in net N from marking m as $\mathcal{R}(N, m)$. We will drop N and write $\mathcal{R}(m)$ when no ambiguities can arise.

Place invariants (see [12]) A *place invariant* is a row vector $I : P \to \mathbb{Q}$ such that $I \cdot F = 0$. When talking about invariants, we consider markings as *vectors*.

State machines A subclass of Petri nets that we will heavily use further on is *state machines*. State machines can represent conflicts by a place with several output transitions, but they can not represent concurrency and synchronisation. Formally: Let $N = \langle S, T, F \rangle$ be a Petri net. N is a *state machine* (SM) iff $\forall\, t \in T : |{}^\bullet t| = 1 \wedge |t^\bullet| = 1$.

Workflow Petri nets In this paper we primarily focus upon the *Workflow Petri nets (WF-nets)* [1]. As the name suggests, WF-nets are used to model the processing of tasks in workflow processes. The initial and final nodes indicate respectively the initial and final states of processed cases.

Definition 1. *A Petri net N is a* Workflow net (WF-net) *iff:*

1. *N has two special places: i and f. The initial place i is a source place, i.e. ${}^{\bullet}i = \emptyset$, and the final place f is a sink place, i.e. $f^{\bullet} = \emptyset$.*
2. *For any node $n \in (P \cup T)$ there exists a path from i to n and a path from n to f.*

We consider the processing of batches of tasks in Workflow nets, meaning that the initial place of a Workflow net may contain an arbitrary number of tokens. Our goal is to provide correctness criteria for the design of these nets. One natural correctness requirement is *proper termination*, which is called *soundness* in the WF-net theory. We will use the generalised notion of soundness for WF-nets introduced in [11]:

Definition 2. *N is k-*sound *for some $k \in \mathbb{N}$ iff for all $m \in \mathcal{R}(k[i])$, $m \xrightarrow{*} k[f]$. N is* sound *iff it is k-sound for all $k \in \mathbb{N}$.*

3 Resource-Constrained Workflow Nets

Workflow nets specify the handling of tasks within the organisation, factory, etc. without taking into account resources available there for the execution. We extend here the notion of WF-nets in order to include information about the use of resources into the model.

A resource belongs to a type; we have one place per resource type in the net where the resources are located when they are free. We assume that resources are durable, i.e. they can neither be created nor destroyed, they are claimed during the handling procedure and then released again. By abstracting from the resource places we obtain the WF-net that we call *production net*.

Definition 3. *We will say that a WF-net $N = \langle P_p \cup P_r, T, F_p^+ \cup F_r^+, F_p^- \cup F_r^- \rangle$ with initial and final places $i, f \in P_p$ is a* Resource-Constrained Workflow net (RCWF-net) *with the set P_p of* production places *and the set P_r of* resource places *iff*

- *$P_p \cap P_r = \emptyset$,*
- *F_p^+ and F_p^- are mappings $(P_p \times T) \to \mathbb{N}$,*
- *F_r^+ and F_r^- are mappings $(P_r \times T) \to \mathbb{N}$, and*
- *$N_p = \langle P_p, T, F_p^+, F_p^- \rangle$ is a WF-net, which we call the* production net *of N.*

Workflow nets with id-tokens Cases processed in the Workflow net are often independent of each other, i.e. tokens related to different cases cannot interfere with each other. This can be modelled by assigning a *unique id-colour* to each case, and allowing firings only on the tokens of the same colour. Colouring does not concern the resource tokens: resources are shared by all cases processed in the net and are colourless.

Therefore, we extend the semantics of Petri nets by introducing *id-tokens*. Our RCWF-nets will have tokens of two types: coloured tokens on production places, which are pairs (p, a), where p is a place and $a \in Id$ is an identifier, and uncoloured tokens on resource places. We assume Id to be a countable set. We will write x_p for the projection of $x \in \mathbb{N}^P$ on production places (coloured part of the marking) and x_r for the projection of x on resource places (uncoloured part). A transition $t \in T$ is *enabled* in m iff $({}^\bullet t)_r \leq m$ and there exists $a \in Id$ such that m_p contains tokens on $({}^\bullet t)_p$ with identifier a. A firing of t results in consuming these tokens and producing tokens with identifier a to $(t^\bullet)_p$ and uncoloured tokens to $(t^\bullet)_r$. Later on, we will use the extended semantics when working with id-tokens, and the standard semantics for classical tokens.

Though being a very simple sort of coloured nets, WF-nets with id-tokens are often expressive enough to reflect the essence of a modelled process, separating different cases which are processed in the net concurrently.

Soundness of RCWF-nets Soundness in WF-nets is the property that says that every marking reachable from an initial marking with k tokens on the initial place terminates properly, i.e. it can reach a marking with k tokens on the final place, for an arbitrary natural number k. In the RCWF-net, the initial marking of the net is a marking with some tokens on the initial place and a number of resource tokens on the resource places. With the proper termination for RCWF-nets we mean that the resource tokens are back to their resource places and all tasks are processed correctly, i.e. all the places of N_p except for f are empty. Moreover, we want the net to work properly not only with some fixed amount of resources but also with any greater amount: we want the verified system to work correctly also when more money, manpower, or machinery is available. On the other hand, it is clear that there is some minimal amount of resources needed to guarantee that the system can work at all.

Another correctness requirement that should be reflected by the definition of soundness is that resource tokens cannot be created during the processing, i.e. at any moment of time the number of available resources does not exceed the number of initially given resources. The extended definition of soundness reads thus as follows:

Definition 4 (soundness of RCWF-nets). *Let N be an RCWF-net.*
N is (k, R)-sound *for some $k \in \mathbb{N}, R \in \mathbb{N}^{P_r}$ iff for all $m \in \mathcal{R}(\sum_{a \in Id}[i, a] + R)$ with $|Id| = k$ holds: $m_r \leq R$ and $m \xrightarrow{*} (\sum_{a \in Id}[f, a] + R)$.*
N is k-sound *iff there exists $R \in \mathbb{N}^{P_r}$ such that it is (k, R')-sound for all $R' \geq R$.*
N is sound *iff there exists $R \in \mathbb{N}^{P_r}$ such that it is (k, R')-sound for all $k \in \mathbb{N}, R' \geq R$.*

The soundness problem is a *parameterised* problem formulated on a *coloured* Petri net. We will first use the nature of the colouring to reduce this problem to a problem on an uncoloured net.

Lemma 5. *The production net of a sound RCWF-net is 1-sound.*

Proof. Since we want to prove 1-soundness, we only have to consider the processing of a single case in the net, and therefore all production tokens have the same colour, which we abstract from. Let N be a sound RCWF-net and assume that N_p is not 1-sound. Then there exist a firing sequence σ and a production marking m_p such that $[i] \xrightarrow{\sigma}_{N_p} m_p$ and $m_p \not\xrightarrow{*}_{N_p} [f]$. Take enough resources to enable σ in N, then $m_p + m_r$ is reachable in $(N, [i] + (m_0)_r)$ but $m_p + m_r \not\xrightarrow{*}_N [f] + (m_0)_r$, which contradicts the soundness of the RCWF-net. $\square$

1-soundness of the production net is thus a necessary condition of the soundness of the RCWF-net. 1-soundness of a WF-net can be checked by checking that the closure[1] of the WF-net is live and bounded [1]. In the rest of the paper we assume that the check of 1-soundness of the production net has been done and its result is positive.

Corollary 6. *For any sound RCWF-net N, $\mathcal{R}(N_p, [i, a])$ is finite for any $a \in Id$.*

Proof. All production tokens in $(N_p, [i, a])$ will have colour a and thus the colour does not influence the behaviour of the net and we abstract from it. Assume $\mathcal{R}(N_p, [i])$ is infinite. Then there are $m_1, m_2 \in \mathcal{R}(N_p, [i])$ such that $m_2 = m_1 + \delta$ for some $\delta > \mathbf{0}$. Since N is sound, N_p is 1-sound and $m_1 \xrightarrow{*}_{N_p} [f]$ and thus $m_1 + \delta \xrightarrow{*}_{N_p} [f] + \delta$. Thus $[f] + \delta \in \mathcal{R}(N_p, m_2) \subseteq \mathcal{R}(N_p, [i])$ and $[f] + \delta \xrightarrow{*}_{N_p} [f]$, which is impossible since f is a sink place and any transition of N_p has at least one output place. $\square$

Given an RCWF-net N with one resource type we construct a resource-constrained state machine WF-net with the same behaviour as N as follows. First, let T be a transition system corresponding to $(N_p, [i])$ extended with the information about resource consumption and production for every transition of T. Then we build a resource-constrained state machine workflow net N' by creating a place for every state of T and a transition with the corresponding resource consumption/production for every transition of T. Observe that due to the use of id-tokens, N' is sound iff N is. Hence, we can check soundness of an RCWF-net by checking soundness of the corresponding state machine workflow net.

In this paper we restrict our attention to Resource-Constrained Workflow nets with one type of resources. This is a typical situation in various practical applications with memory, money or manpower being the considered resource. Therefore, in the remainder of the paper we consider only state machine workflow nets with one resource type (SM1WF-nets):

Definition 7. *An RCWF-net $N = \langle P_p \cup P_r, T, F_p^+ \cup F_r^+, F_p^- \cup F_r^- \rangle$ is called a* state machine workflow net with one resource type (SM1WF-net) *if $P_r = \{r\}$ and the production net N_p of N is a state machine.*

Note that a production token in the SM1WF-net represents a part of a production marking of the original RCWF-net related to one case (one id-colour).

[1] The closure of a WF-net N is the net obtained by adding to N a transition with f as the input place and i as the output place.

Thus all production tokens in the SM1WF-net have different id-colours. Note that every firing in an SM1WF-net requires only one production token (and a number of resource tokens) and results in the production of a single production token (and a number of resource tokens). Therefore we can abstract from colours when considering soundness of SM1WF-nets.

For SM1WF-nets we will write $^\circ t$ and t° for the input/output place of t in the production net.

4 Soundness Check for SM1WF-nets

In this section we will give a necessary and sufficient condition for the soundness of SM1WF-nets. We start by introducing a notion of *path* that we will use here. Unlike a trace, a path does not deal with the processing of multiple production tokens. Formally, given an SM1WF-net N, a *path* is a sequence $(t_1 \dots t_n)$ of transitions in T such that $\forall\, k : 1 \leq k < n : t_k^\circ = {}^\circ t_{k+1}$. We will write $^\circ\sigma$ and σ° for the input and the output place of a nonempty path $\sigma = (t_1 \dots t_n)$, i.e. for $^\circ t_1, t_n^\circ$ respectively. A path σ is called a *successor* of a path ρ (and ρ a *predecessor* of σ) if $\rho^\circ = {}^\circ\sigma$. Their juxtaposition $\rho\sigma$ then is again a path of N.

With every path we associate three numbers: its resource production, consumption and effect.

Definition 8. *Let N be an SM1WF-net. The resource effect $\mathcal{E}$, production $\mathcal{P}$ and consumption $\mathcal{C}$ are defined as follows:*

- *for the empty path ϵ, $\mathcal{E}(\epsilon) = \mathcal{P}(\epsilon) = \mathcal{C}(\epsilon) = 0$;*
- *for a path t, $t \in T$, $\mathcal{E}(t) = t^\bullet(r) - {}^\bullet t(r)$, $\mathcal{P}(t) = t^\bullet(r)$, and $\mathcal{C}(t) = {}^\bullet t(r)$;*
- *for a path σt, $\mathcal{E}(\sigma t) = \mathcal{E}(\sigma) + \mathcal{E}(t)$, $\mathcal{P}(\sigma t) = \max(\mathcal{P}(t), \mathcal{P}(\sigma) + \mathcal{E}(t))$ and for a path $t\sigma$, $\mathcal{C}(t\sigma) = \max(\mathcal{C}(t), \mathcal{C}(\sigma) - \mathcal{E}(t))$.*

The notion of effect allows us to distinguish three kinds of paths. A path σ is called a *C-path (consumption path)* if $\mathcal{E}(\sigma) < 0$, an *E-path (equality path)* if $\mathcal{E}(\sigma) = 0$, and a *P-path (production path)* if $\mathcal{E}(\sigma) > 0$.

Example 9. Now we will illustrate the intuitive meaning of $\mathcal{E}, \mathcal{P}$ and C on an example and in the rest of the section we will prove that $\mathcal{E}, \mathcal{P}$ and C confirm this intuition indeed. Consider paths tu and vx of SM1WF-net N in Fig. 1.[2] The resource effect of these paths $\mathcal{E}(tu) = 1 - 4 + 5 - 2 = 0$ and $\mathcal{E}(vx) = 3 - 1 + 3 - 2 = 3$, which corresponds to the change of the number of resource tokens due to the firing of the transitions of the corresponding path. $\mathcal{P}(tu) = \max(\mathcal{P}(u), \mathcal{P}(t) + \mathcal{E}(u)) = \max(5, 1 + 3) = 5$ and $\mathcal{P}(vx) = \max(\mathcal{P}(x), \mathcal{P}(v) + \mathcal{E}(x)) = \max(3, 3 + 1) = 4$. Note that $\mathcal{P}(tu), \mathcal{P}(vx)$ correspond to the minimal number of resource tokens we are guaranteed to have immediately after the firing

[2] Instead of drawing a resource place and its in- and outgoing arcs, we put the weights of the arcs from and to the resource place under the corresponding transitions. So $(4, 1)$ for transition t means that t consumes 4 resource tokens and then releases 1 resource token.

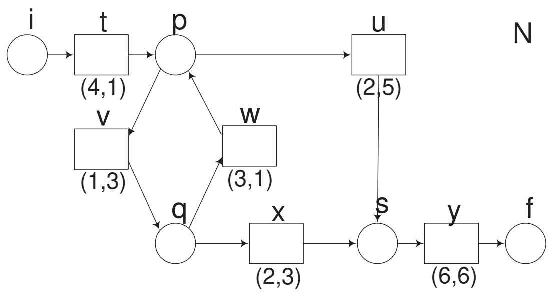

Fig. 1. Example of an SM1WF-net

of tu/vx respectively. $\mathcal{C}(tu) = \max(\mathcal{C}(t), \mathcal{C}(u) - \mathcal{E}(t)) = \max(4, 2+3) = 5$ and $\mathcal{C}(vx) = \max(\mathcal{C}(v), \mathcal{C}(x) - \mathcal{E}(v)) = \max(1, 2-2) = 1$. $\mathcal{C}(tu)$ and $\mathcal{C}(vx)$ correspond to the minimal number of resource tokens needed to make the firings of tu/vx possible.

4.1 Properties of the resource-effect function

Lemma 10. *Let N be a sound SM1WF-net. Then for any place $p \in P_p$ and any two paths σ and ρ such that ${}^\circ\sigma = {}^\circ\rho = i$ and $\sigma^\circ = \rho^\circ = p$ holds $\mathcal{E}(\sigma) = \mathcal{E}(\rho) \leq 0$.*

Proof. Since N is sound, N_p is sound as well and there exists a firing sequence γ such that $[p] \xrightarrow{\gamma} [f]$. Take R large enough to make both $\sigma\gamma$ and $\rho\gamma$ firable from $[i] + R[r]$. Thus $[i] + R[r] \xrightarrow{\sigma} [p] + (R + \mathcal{E}(\sigma))[r] \xrightarrow{\gamma} [f] + R[r]$ and $[i] + R[r] \xrightarrow{\rho} [p] + (R + \mathcal{E}(\rho))[r] \xrightarrow{\gamma} [f] + R[r]$, which implies that $\mathcal{E}(\sigma) = \mathcal{E}(\rho)$. Moreover, since N is sound and thus no resource creation happens, $R + \mathcal{E}(\sigma) \leq R$, i.e. $\mathcal{E}(\sigma) \leq 0$. $\square$

Thus, in a sound SM1WF-net, each production place p has a unique weight defined as $-\mathcal{E}(\sigma)$ for some σ such that ${}^\circ\sigma = i$ and $\sigma^\circ = p$, showing how many resources possesses a production token on place p. (Clearly, the weight can be equivalently defined as $\mathcal{E}(\rho)$ where ρ is some sequence with ${}^\circ\rho = p$ and $\rho^\circ = f$.) This observation leads to the following place invariant property for sound SM1WF-nets:

Lemma 11. *Let N be a sound SM1WF-net with the initial place i, the final place f, and the resource place r. Then, there exists a unique place invariant W such that $W(i) = W(f) = 0$, $W(r) = 1$. Moreover, for every place $p \in P_p$, $W(p) = -\mathcal{E}(\sigma)$ for any σ with ${}^\circ\sigma = i$ and $\sigma^\circ = p$, and hence $W(p) \geq 0$ for all $p \in P_p$.*

Proof. The proof is done in a constructive way. Since N is sound, we have a unique mapping $W : P \to \mathbb{N}$ such that for every place $p \in P_p$ $W(p) = -\mathcal{E}(\sigma)$

where σ is some path with ${}^\circ\sigma = i$ and $\sigma^\circ = p$, and $W(r) = 1$. By construction, for any sound net $W(i) = W(f) = 0$ and $W(p) \geq 0$, for all $p \in P_p$.

Now we will show that W is a place invariant, i.e. $W \cdot F = 0$. Since N_p is a state machine, a column of F corresponding to a transition t has -1 in the cell ${}^\circ t$, 1 in t° and $t^\bullet(r) - {}^\bullet t(r)$ in the resource place r. Hence, the product of W and the t-column of F can be written as $-W({}^\circ t) + W(t^\circ) + (t^\bullet(r) - {}^\bullet t(r)) \cdot W(r) = \mathcal{E}(\sigma) - \mathcal{E}(\sigma t) + t^\bullet(r) - {}^\bullet t(r) = 0$ (σ is some path with ${}^\circ\sigma = i$ and $\sigma^\circ = {}^\circ t$). Since the same reasoning can be applied to any transition t, we have $W \cdot F = 0$.

By induction on the length of σ with ${}^\circ\sigma = i, \sigma^\circ = p$, it is easy to show that W is unique, i.e. for any invariant W' such that $W'(i) = W'(f) = 0$ and $W'(r) = 1$ we have $W(p) = -\mathcal{E}(\sigma)$. □

Thus the existence of such an invariant is a *necessary condition* of soundness. This condition can be easily checked by standard algebraic procedures. For net N from Fig. 1 the invariant is $r + 3p + q$, i.e. the weights of places are $W(p) = 3$, $W(q) = 1$ and $W(s) = 0$. We assume further on that N is an SM1WF-net with a unique place invariant W satisfying $W(i) = W(f) = 0$ and $W(r) = 1$, and moreover, we have $W(p) \geq 0$.

4.2 Properties of the consumption and production functions

The following lemma states that at least $\mathcal{C}(\sigma)$ resources are needed to execute σ and at least $\mathcal{P}(\sigma)$ resources become available after the execution of σ.

Lemma 12. *Let σ be a path in N. Then*

1. *Let $M \xrightarrow{\sigma} M'$ for some markings M, M' and path σ. Then $M'(r) \geq \mathcal{P}(\sigma)$ and $M(r) \geq \mathcal{C}(\sigma)$.*
2. *$[{}^\circ\sigma] + \mathcal{C}(\sigma)[r] \xrightarrow{\sigma} [\sigma^\circ] + \mathcal{P}(\sigma)[r]$ for any $\sigma \neq \epsilon$.*

Proof. We prove Part 1 by induction on the length of σ. If $\sigma = \epsilon$, the lemma holds. We prove the $\mathcal{P}$-part by setting $\sigma = \rho t$. Let M'' be such that $M \xrightarrow{\rho} M'' \xrightarrow{t} M'$. By the induction hypothesis, $M''(r) \geq \mathcal{P}(\rho)$ and thus, $M'(r) \geq \max(\mathcal{P}(t), \mathcal{P}(\rho) + \mathcal{E}(t))$, i.e., $M'(r) \geq \mathcal{P}(\sigma)$, completing the proof of (1). We omit the proof of the $\mathcal{C}$-part since it can be obtained analogously by taking $\sigma = t\rho$.

Part 2 follows from the existence of markings M and M' such that

$$M \xrightarrow{\sigma} [\sigma^\circ] + \mathcal{P}(\sigma)[r] \quad \text{and} \quad [{}^\circ\sigma] + \mathcal{C}(\sigma) \xrightarrow{\sigma} M'. \tag{1}$$

We prove (1) by induction on the length of σ. The case $\sigma = t$, where $t \in T$, is clear. For the $\mathcal{P}$-part, let $\sigma = \rho t$, with $\rho \neq \epsilon$. By the induction hypothesis, there exists M'' such that $M'' \xrightarrow{\rho} [\rho^\circ] + \mathcal{P}(\rho)[r]$. Note that $\mathcal{P}(\sigma) = \mathcal{P}(\rho t) = \max(\mathcal{P}(t), \mathcal{P}(\rho) + \mathcal{E}(t))$. We distinguish between two cases:

- If $\mathcal{P}(\sigma) = \mathcal{P}(\rho) + \mathcal{E}(t)$, then $\mathcal{P}(\rho) + \mathcal{E}(t) \geq \mathcal{P}(t)$, i.e., $\mathcal{P}(\rho) \geq \mathcal{P}(t) - \mathcal{E}(t) = {}^\bullet t(r)$. Hence, $\mathcal{P}(\rho) \geq \mathcal{C}(t)$ and $[\rho^\circ] + \mathcal{P}(\rho)[r] \xrightarrow{t} [\sigma^\circ] + (\mathcal{P}(\rho) + \mathcal{E}(t))[r]$. Recall that $\mathcal{P}(\rho) + \mathcal{E}(t) = \mathcal{P}(\sigma)$, i.e., $[\rho^\circ] + \mathcal{P}(\rho)[r] \xrightarrow{t} [\sigma^\circ] + \mathcal{P}(\sigma)$, so we take $M = M''$ and have $M \xrightarrow{\sigma} [\sigma^\circ] + \mathcal{P}(\sigma)[r]$.

– If $\mathcal{P}(\sigma) = \mathcal{P}(t)$, then $\mathcal{P}(\rho) + \mathcal{E}(t) \leq \mathcal{P}(t)$, i.e., $\mathcal{P}(\rho) \leq \mathcal{P}(t) - \mathcal{E}(t)$. Therefore, $\mathcal{P}(\rho) \leq \mathcal{C}(t)$ and we take $M = M'' + (\mathcal{C}(t) - \mathcal{P}(\rho))[r]$. Thus, $M \xrightarrow{\rho} [\rho^\circ] + \mathcal{C}(t)[t] \xrightarrow{t} [\sigma^\circ] + \mathcal{P}(t)[t]$. Since $\sigma = \rho t$, $M \xrightarrow{\sigma} [\sigma^\circ] + \mathcal{P}(\sigma)[r]$.

The $\mathcal{C}$-part is analogous, using $\sigma = t\rho$. Due to Part 1 of the lemma, M and M' in (1) satisfy $M \geq [{}^\circ\sigma] + \mathcal{C}(\sigma)[r]$ and $M' \geq [\sigma^\circ] + \mathcal{P}(\sigma)[r]$. Hence $M \xrightarrow{\sigma} M' + \delta$ where $\delta = M - ([{}^\circ\sigma] + \mathcal{C}(\sigma)[r])$ and $M + \delta' \xrightarrow{\sigma} M'$ where $\delta' = M' - ([\sigma^\circ] + \mathcal{P}(\sigma)[r])$. Thus we conclude that $\delta = \delta' = \emptyset$ and $[{}^\circ\sigma] + \mathcal{C}(\sigma)[r] \xrightarrow{\sigma} [\sigma^\circ] + \mathcal{P}(\sigma)[r]$. □

Corollary 13. *$\mathcal{E}(\sigma) = \mathcal{P}(\sigma) - \mathcal{C}(\sigma)$ and $\mathcal{E}(\sigma) = W({}^\circ\sigma) - W(\sigma^\circ)$ for all σ.*

Proof. Follows directly from Lemma 12.(2) and the definition of W. □

Corollary 14. *Let $k > 0$ and σ be a path such that $\mathcal{E}(\sigma) \leq 0$. Then,*

$$k[{}^\circ\sigma] + (\mathcal{C}(\sigma) - (k-1) * \mathcal{E}(\sigma))[r] \xrightarrow{\sigma^k} k[\sigma^\circ] + \mathcal{P}(\sigma)[r]$$

Proof. The proof is done by induction on k with the use of Lemma 12(2) and Corollary 13. □

Next we show that under certain conditions two paths can be swapped.

Lemma 15 (Interchange Lemma). *Let M, M' be markings and σ, ρ be paths such that $\mathcal{E}(\sigma) \leq 0 \leq \mathcal{E}(\rho)$, and ρ is not a successor of σ. If $M \xrightarrow{\sigma\rho} M'$ then $M \xrightarrow{\rho\sigma} M'$.*

Proof. Let M_1 be a marking such that $M \xrightarrow{\sigma} M_1 \xrightarrow{\rho} M'$. Since $\sigma^\circ \neq {}^\circ\rho$, $M_1 \geq [\sigma^\circ] + [{}^\circ\rho] + \max(\mathcal{C}(\rho), \mathcal{P}(\sigma))[r]$. Hence, $M \geq [{}^\circ\sigma] + [{}^\circ\rho] + \max(\mathcal{C}(\rho) - \mathcal{E}(\sigma), \mathcal{C}(\sigma))[r]$. Since $\mathcal{E}(\sigma) \leq 0$, there exists a marking M_2 such that $M \xrightarrow{\rho} M_2$ and $M_2 \geq [{}^\circ\sigma] + [\rho^\circ] + \max(\mathcal{P}(\rho) - \mathcal{E}(\sigma), \mathcal{C}(\sigma) + \mathcal{E}(\rho))[r]$. Therefore, $M_2 \geq [{}^\circ\sigma] + [\rho^\circ] + (\mathcal{C}(\sigma) + \mathcal{E}(\rho))[r]$. Recall that $\mathcal{E}(\rho) \geq 0$, so $M_2 \geq [{}^\circ\sigma] + \mathcal{C}(\sigma)[r]$ and thus $M \xrightarrow{\rho\sigma} M'$. □

The next lemma gives implicit lower bounds for the number of resources in states reachable from the initial marking and states that reach the final marking.

Lemma 16. *Let $M, M' \in \mathbb{N}^P$ with $M(r) < M'(r)$.*
If $M' \xrightarrow{*} M$, there exists a C-path ρ such that $M \geq [\rho^\circ] + \mathcal{P}(\rho)[r]$.
If $M \xrightarrow{*} M'$, there exists a P-path σ such that $M \geq [{}^\circ\sigma] + \mathcal{C}(\sigma)[r]$.

Proof. Let $M' \xrightarrow{\alpha} M$. We normalise the trace α as follows. We write α as the concatenation of paths $\sigma_1 \ldots \sigma_n$, where no σ_{k+1} is a successor of σ_k. If α contains a C-path σ_k succeeded by a P-path or by an E-path σ_{k+1}, we swap them in α, obtaining α'. By the interchange lemma, $M' \xrightarrow{\alpha'} M$. We continue with normalizing α' further by using the same procedure. The normalisation

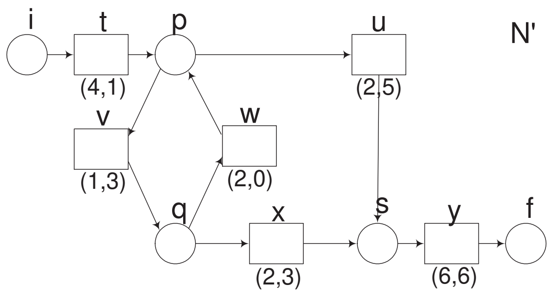

Fig. 2. Example of an unsound SM1WF-net

process terminates since every swap decreases the number of P- and E-paths following a C-path.

Thus, there exists a trace β such that $M' \xrightarrow{\beta} M$ and the division of β into paths consists of a number of P- and/or E-paths followed by C-paths. Since $M(r) < M'(r)$, β contains at least one C-path. Let ρ be the last path of β. Then ρ is a C-path, $M(\rho^\circ) > 0$ and by statement (1) of Lemma 12, $M(r) \geq \mathcal{P}(\rho)$.

Similarly, if $M \xrightarrow{\gamma} M'$, there exists a trace δ containing P-paths followed by C- and/or E-paths such that $M \xrightarrow{\delta} M'$. Since $M(r) < R$, δ contains at least one P-path. Let σ be the first P-path. Then by Lemma 121, $M(r) \geq \mathcal{C}(\sigma)$. $\square$

We will show that the $\mathcal{C}$-bound in Lemma 16 is sharp. (Sharpness of the $\mathcal{P}$-bound can be proved but is not needed here.)

Lemma 17. *Let $k_0 > 0$ and let σ be a C-path. Then there exist $k > k_0$ and $R \in \mathbb{N}$ such that $k[i] + R[r] \xrightarrow{*} k[\sigma^\circ] + \mathcal{P}(\sigma)[r]$.*

Proof. Let $p = {}^\circ\sigma, q = \sigma^\circ$. There exists a path ρ with ${}^\circ\rho = i, \rho^\circ = p$. Since we assume the existence of the place invariant as described in Lemma 11, $\mathcal{E}(\rho) \leq 0$. So by Corollary 14, $k[i] + (\mathcal{C}(\rho) - (k-1) * \mathcal{E}(\rho))[r] \xrightarrow{\rho^k} k[p] + \mathcal{P}(\rho)[r]$ for all $k > 0$. Since $\mathcal{E}(\sigma) < 0$, there exists $k > k_0$ such that $\mathcal{C}(\sigma) - (k-1)\mathcal{E}(\sigma) \geq \mathcal{P}(\rho)$. By taking $R = \mathcal{C}(\rho) - (k-1) * \mathcal{E}(\rho) + (\mathcal{C}(\sigma) - (k-1)\mathcal{E}(\sigma) - \mathcal{P}(\rho))$, we obtain by Corollary 14: $k[i] + R(k)[r] \xrightarrow{\rho^k} k[p] + (\mathcal{C}(\sigma) - (k-1)\mathcal{E}(\sigma))[r] \xrightarrow{\sigma^k} k[q] + \mathcal{P}(\sigma)[r]$. $\square$

The construction described in the proof of Lemma 17 will be later on used for giving a meaningful verification feedback on unsound nets, namely we will construct an example of a deadlock/livelock in an unsound net.

Example 18. Consider the consumption path $\sigma = w$ in net N' from Fig. 2 (which differs from net N from Fig. 1 only in the resource consumption/production of transition w); ${}^\circ\sigma = q$, $\sigma^\circ = p, \mathcal{E}(\sigma) = -2, \mathcal{C}(\sigma) = 2$. Take tv as ρ; $\mathcal{E}(\rho) = -1, \mathcal{C}(\rho) = 4, \mathcal{P}(\rho) = 3$. We choose R as $\mathcal{C}(\rho) - (k-1) * \mathcal{E}(\rho) + (\mathcal{C}(\sigma) - (k -$

$1)\mathcal{E}(\sigma) - \mathcal{P}(\rho)) = 4 + (k-1) + 2 + 2(k-1) - 3 = 3k$. Then $k[i] + 3k[r] \xrightarrow{(tv)^k} k[q] + 2k[r] \xrightarrow{w^k} k[p]$. Note that no resources are left and thus we obtained a deadlock since we need resources to proceed. We can get R larger than any given number just by taking a larger k.

Finally, we are ready to state the main theorem, giving a necessary and sufficient condition for the soundness of SM1WF nets.

Theorem 19. *An SM1WF net N is sound iff there exists a unique place invariant W such that $W(i) = W(f) = 0$, $W(r) = 1$, and moreover $W(p) \geq 0$ for all $p \in P_p$, and for each C-path ρ there is a successor P-path σ such that $\mathcal{P}(\rho) \geq \mathcal{C}(\sigma)$.*

Proof. ($\Rightarrow$): Assume there exists a C-path ρ such that all succeeding P-paths σ satisfy $\mathcal{P}(\rho) < \mathcal{C}(\sigma)$. By Lemma 17, there exist k and $R > \mathcal{P}(\rho)$ such that $k[i] + R[r] \xrightarrow{*} M = k[\rho^\circ] + \mathcal{P}(\rho)[r]$. If $M \xrightarrow{*} k[f] + R[r]$, by Lemma 16 there exists a P-path σ with $M(r) \geq \mathcal{C}(\sigma)$, contradicting the assumption. So $M \not\xrightarrow{*} k[f] + R[r]$ and the net is not sound.
($\Leftarrow$): Let R_0 be a maximal $\mathcal{C}(\rho)$ over all paths ρ of N with $\rho^\circ = f$. The choice of R_0 ensures that if at least R_0 resources are present, one token in the production net can be successfully transferred from any place to f.

Suppose that $R \geq R_0$ and $k[i] + R[r] \xrightarrow{*} M$. We prove by induction on $R - M(r)$ that there exists a marking M' with $M'(r) = R$ such that $M \xrightarrow{*} M'$, i.e., that for any reachable marking there is a way to continue and to return all the resources consumed so far. Note that $M(r) \leq R$ since the number of the resources consumed is always non-negative, i.e. no resources are created (due to the existence of the place invariant W). If $M(r) = R$, the statement clearly holds. If $M(r) < R$, by applying Lemma 16 to $k[i]+R[r] \xrightarrow{*} M$, we conclude that there exists a C-path ρ such that $M \geq [\rho^\circ]+\mathcal{P}(\rho)[r]$. By the condition of the theorem, there exists a P-path σ and a marking $M'' = M - [{}^\circ\sigma] + [\sigma^\circ] + \mathcal{E}(\sigma)[r]$ such that $M \xrightarrow{\sigma} M''$, so $M \xrightarrow{*} M''$. Since $M''(r) > M(r)$, the induction hypothesis is applicable to M'', i.e. finally we obtain that $M'' \xrightarrow{*} M'$ and $M'(r) = R$.

Let $p \in P_p$ be such that $M'(p) > 0$. Then since $R \geq R_0$ and by the choice of R_0, we have $[p] + R[r] \xrightarrow{*} [f] + R[r]$. So $M' \xrightarrow{*} M' - [p] + [f]$. We can repeat this procedure for all $p \neq f$ with $M(p) > 0$, reaching $k[f] + R[r]$. $\square$

Note that the net may be unsound if it contains a deadlock (a nonterminal marking where there are no enough resources to proceed any further even with one single step) or a livelock (there are always enough resources to make a following step, but all possible steps are not "progress"-steps, i.e. we cannot leave the cycle). With a slight modification of the condition in Theorem 19 we can diagnose whether the net has no deadlock: along with the invariant requirement we require that for each C-path ρ there is a successor path σ (no matter whether σ is a P-path or a C-path) such that $\mathcal{P}(\rho) \geq \mathcal{C}(\sigma)$. This reflects the requirement that there is always some next step possible. If the net has no deadlock but does not meet the requirements of Theorem 19, this net has a livelock.

5 Decision algorithm

The necessary and sufficient condition formulated in Theorem 19 allows to characterise soundness of SM1WF-nets. The condition as stated is however not directly verifiable, since infinitely many different paths should be taken into account. In this subsection we show that checking finitely many paths is sufficient. The decision algorithm we give here is polynomial in the size of the SM1WF-net.

We start by the following simple observation.

Lemma 20. *Let σ be a cyclic path (i.e. ${}^\circ\sigma = \sigma^\circ$). Then for any ρ_1, ρ_2 such that ${}^\circ\sigma = \rho_1^\circ$ and $\sigma^\circ = {}^\circ\rho_2$ we have $\mathcal{E}(\rho_1\sigma\rho_2) = \mathcal{E}(\rho_1\rho_2)$, $\mathcal{P}(\rho_1\sigma\rho_2) = \mathcal{P}(\rho_1\rho_2)$, and $\mathcal{C}(\rho_1\sigma\rho_2) = \mathcal{C}(\rho_1\rho_2)$.*

Proof. For $\mathcal{E}$ the lemma follows from Lemma 11. Results for $\mathcal{P}$ and $\mathcal{C}$ can be obtained analogously. $\square$

Hence, to check the condition of Theorem 19 it is sufficient to consider acyclic paths only. Since there are finitely many acyclic paths, soundness of SM1WF-nets is decidable. As we showed in Section 3, the soundness of RCWF-nets can be reduced to the soundness of SM1WF-nets, and thus we can conclude the following:

Corollary 21. *Soundness of RCWF-nets with one resource is decidable.*

Next we give an efficient decision algorithm for SM1WF-nets. The algorithm is based on the following property of paths.

Lemma 22. *Let ρ, σ be paths such $\sigma^\circ = {}^\circ\rho$. Then, $\mathcal{P}(\rho\sigma) = \max(\mathcal{P}(\rho) + \mathcal{E}(\sigma), \mathcal{P}(\sigma))$ and $\mathcal{C}(\rho\sigma) = \max(\mathcal{C}(\rho), \mathcal{C}(\sigma) - \mathcal{E}(\rho))$.*

Proof. Suppose $[{}^\circ\rho] + A[r] \xrightarrow{\rho} [\rho^\circ] + B[r] = [{}^\circ\sigma] + B[r] \xrightarrow{\sigma} [\sigma^\circ] + C[r]$. Then by Lemma 12.(1) (applied both to ρ and to σ) $A \geq \mathcal{C}(\rho)$ and $C \geq \mathcal{P}(\sigma)$. Since $B = A + \mathcal{E}(\rho) \geq \mathcal{C}(\rho) + \mathcal{E}(\rho) = \mathcal{P}(\rho)$ and $C = B + \mathcal{E}(\sigma)$, i.e. $B = C - \mathcal{E}(\sigma) \geq \mathcal{P}(\sigma) - \mathcal{E}(\sigma) = \mathcal{C}(\sigma)$, we deduce that $B \geq \max(\mathcal{P}(\rho), \mathcal{C}(\sigma))$. Thus $A \geq \max(\mathcal{C}(\rho), \mathcal{C}(\sigma) - \mathcal{E}(\rho)$ and $C \geq \max(\mathcal{P}(\sigma), \mathcal{P}(\rho) + \mathcal{E}(\sigma))$. By applying Lemma 12.(2) to ρ and σ, we deduce that $[{}^\circ\rho] + \max(\mathcal{C}(\rho), \mathcal{C}(\sigma) - \mathcal{E}(\rho))[r] \xrightarrow{\rho} [\rho^\circ] + \max(\mathcal{P}(\rho), \mathcal{C}(\sigma))[r] \xrightarrow{\sigma} [\sigma^\circ] + \max(\mathcal{P}(\rho) + \mathcal{E}(\sigma), \mathcal{P}(\sigma))[r]$ indeed. Finally, using Lemma 12.(1) and Lemma 12.(2) on $\rho\sigma$, we conclude that $\mathcal{P}(\rho\sigma) = \max(\mathcal{P}(\rho) + \mathcal{E}(\sigma), \mathcal{P}(\sigma))$ and $\mathcal{C}(\rho\sigma) = \max(\mathcal{C}(\rho), \mathcal{C}(\sigma) - \mathcal{E}(\rho))$. $\square$

For $X = \emptyset$ we assume $\min X = \omega$. For $p, q \in P_p$, we define $\mu(p, q)$ as $\min \{\mathcal{P}(\sigma) + W(q) \mid {}^\circ\sigma = p \wedge \sigma^\circ = q\}$. If ${}^\circ\sigma = p$ and $\sigma^\circ = q$, then $\mathcal{C}(\sigma) + W(p) = \mathcal{P}(\sigma) + W(q)$, so $\mu(p, q)$ can alternatively be defined as $\min \{\mathcal{C}(\sigma) + W(p) \mid {}^\circ\sigma = p \wedge \sigma^\circ = q\}$. Then, the condition from Theorem 19 can be now reformulated in the following way, assuming the existence of the place invariant W:

Corollary 23. *N is sound if and only if*

$$\forall\, x \in P_p : \min\{\mu(y,x) \mid W(y) < W(x)\} \geq \min\{\mu(x,y) \mid W(y) < W(x)\}.$$

Analogously to Corollary 23, we can show that SM1WF-net has no deadlock iff

$$\forall\, x \in P_p : \min\{\mu(y,x) \mid W(y) < W(x)\} \geq \min\{\mu(x,y)\}.$$

With these conditions we can diagnose SM1WF-nets as sound, non-sound due to deadlock, or non-sound due to livelock.

Function μ has the following important property:

Lemma 24. *For all p and q in P_p we have $\mu(p,q) = \min\{\max(\mu(p,r), \mu(r,q)) \mid r \in P_p\}$.*

Proof. Recall that $\mu(p,q)$ is defined as $\min\{\mathcal{P}(\sigma) + W(q) \mid {}^\circ\sigma = p \wedge \sigma^\circ = q\}$. Every path from p to q can be seen as $\rho_1\rho_2$ for some paths ρ_1 from p to some r and ρ_2 from r to q. Hence, $\mathcal{P}(\sigma) + W(q) = \mathcal{P}(\rho_1\rho_2) + W(q)$, and by Lemma 22, $\mathcal{P}(\rho_1\rho_2) + W(q) = \max(\mathcal{P}(\rho_1) + \mathcal{E}(\rho_2), \mathcal{P}(\rho_2)) + W(q) = \max(\mathcal{P}(\rho_1) + \mathcal{E}(\rho_2) + W(q), \mathcal{P}(\rho_2) + W(q))$. Since ρ_2 is one of the possible paths from r to q, $\mathcal{P}(\rho_2) + W(q) \geq \mu(r,q)$. By Corollary 13, $\mathcal{E}(\rho_2) + W(q) = W(r)$. Therefore, $\mathcal{P}(\rho_1) + \mathcal{E}(\rho_2) + W(q) = \mathcal{P}(\rho_1) + W(r)$ and $\mathcal{P}(\rho_1) + \mathcal{E}(\rho_2) + W(q) \geq \mu(p,r)$. Summarizing these two parts obtain $\mathcal{P}(\sigma) + W(q) = \max(\mathcal{P}(\rho_2) + W(q), \mathcal{P}(\rho_1) + \mathcal{E}(\rho_2) + W(q)) \geq \max(\mu(r,q), \mu(p,r))$. Thus, $\mu(p,q) \geq \min\{\max(\mu(p,r), \mu(r,q)) \mid r \in P_p\}$.

Let s be such that $\min\{\max(\mu(p,r), \mu(r,q)) \mid r \in P_p\} = \max(\mu(p,s), \mu(s,q))$, i.e. the minimum is reached on s, and let $\mu(p,s) = \mathcal{P}(\sigma) + W(s)$ for some σ with ${}^\circ\sigma = p$ and $\sigma^\circ = s$ and $\mu(s,q) = \mathcal{P}(\gamma) + W(q)$ for some γ with ${}^\circ\gamma = s$ and $\gamma^\circ = q$. Then, $\sigma\gamma$ is a path from p to q and it should be taken into account while computing the minimum for $\mu(p,q)$. Hence, $\mu(p,q) \leq \mathcal{P}(\sigma\gamma) + W(q) = \max(\mathcal{P}(\sigma) + \mathcal{E}(\gamma), \mathcal{P}(\gamma)) + W(q) = \max(\mathcal{P}(\sigma) + \mathcal{E}(\gamma) + W(q), \mathcal{P}(\gamma) + W(q)) = \max(\mathcal{P}(\sigma) + W(s), \mu(s,q)) = \max(\mu(p,s), \mu(s,q)$. It implies that $\mu(p,q) \leq \min\{\max(\mu(p,r), \mu(r,q)) \mid r \in P_p\}$.

Therefore, $\mu(p,q) = \min\{\max(\mu(p,r), \mu(r,q)) \mid r \in P_p\}$. ◻

Lemma 24 leads to the following efficient algorithm for computing μ. For two matrices $A, B : P_p \times P_p \to \mathbb{N}$, $A = (a(p,q))$, $B = (b(p,q))$, we define $A \circ B = (c(p,q))$ where $c(p,q) = \min\{\max(a(p,r), b(r,q)) \mid r \in P_p\}$. The matrix $\mu(p,q)$ is computed by initializing the matrix $A = (a(p,q))$ by $a(p,p) = 0$ and $a(p,q) = \min\{\mathcal{P}(t) + W(q) \mid t \in T \wedge {}^\circ t = p \wedge t^\circ = q\}$. We then compute the subsequent powers of A with respect to $\circ$. The computation eventually reaches the fixpoint since the values in the matrix can be changed only to strictly lesser ones with respect to a well-founded ordering on $\mathbb{N} \cup \{\omega\}$. Moreover, A^k takes into account all paths of length k. Therefore, the process terminates after no more steps than the length of the longest acyclic path in the net. Upon termination the matrix becomes $(\mu(p,q))$.

Example 25. In our example net from Fig. 1, we have only one transition t leading from i to p and $W(i) = 0$, $W(p) = 3$, $\mathcal{C}(t) = 4$, $\mathcal{P}(t) = 1$, giving $a(i,p) = 4$ initially. Our full initial matrix A and its iterations become

$$A = \begin{array}{c|ccccc} & i & p & q & s & f \\ \hline i & 0 & 4 & \omega & \omega & \omega \\ p & \omega & 0 & 4 & 5 & \omega \\ q & \omega & 4 & 0 & 3 & \omega \\ s & \omega & \omega & \omega & 0 & 6 \\ f & \omega & \omega & \omega & \omega & 0 \end{array} \quad A^2 = \begin{array}{c|ccccc} & i & p & q & s & f \\ \hline i & 0 & 4 & 4 & 5 & \omega \\ p & \omega & 0 & 4 & 4 & 6 \\ q & \omega & 4 & 0 & 3 & 6 \\ s & \omega & \omega & \omega & 0 & 6 \\ f & \omega & \omega & \omega & \omega & 0 \end{array} \quad A^3 = \begin{array}{c|ccccc} & i & p & q & s & f \\ \hline i & 0 & 4 & 4 & 4 & 6 \\ p & \omega & 0 & 4 & 4 & 6 \\ q & \omega & 4 & 0 & 3 & 6 \\ s & \omega & \omega & \omega & 0 & 6 \\ f & \omega & \omega & \omega & \omega & 0 \end{array}$$

We find $A^4 = A^3$, so A^3 gives the desired $\mu(x, y)$. We now check our condition: $\forall\, x \in P_p : \min\{\mu(y, x) \mid W(y) < W(x)\} \geq \min\{\mu(x, y) \mid W(y) < W(x)\}$. Now $\min\{\mu(y, x) \mid W(y) < W(x)\} = \min\{\mu(x, y) \mid W(y) < W(x)\} = \omega$ for $x \in \{i, r, f\}$, since $W(i) = W(r) = W(f) = 0$ and no place has a smaller weight. Since $W(p) = 3$ and all other places have smaller weight, we have $\min\{\mu(y, p) \mid W(y) < W(p)\} = 4$ and $\min\{\mu(x, y) \mid W(y) < W(p)\} = 4$. Finally, for $x = q$ we have $\min\{\mu(y, q) \mid W(y) < W(q)\} = 4$ and $\min\{\mu(x, y) \mid W(y) < W(q)\} = 3$. Our condition holds, so the net is sound.

Now consider net N' from Fig. 2. Then the $(\mu(x, y))$ is computed as follows:

$$A = \begin{array}{c|ccccc} & i & p & q & s & f \\ \hline i & 0 & 4 & \omega & \omega & \omega \\ p & \omega & 0 & 4 & 5 & \omega \\ q & \omega & 3 & 0 & 3 & \omega \\ s & \omega & \omega & \omega & 0 & 6 \\ f & \omega & \omega & \omega & \omega & 0 \end{array} \quad A^2 = \begin{array}{c|ccccc} & i & p & q & s & f \\ \hline i & 0 & 4 & 4 & 5 & \omega \\ p & \omega & 0 & 4 & 4 & 6 \\ q & \omega & 3 & 0 & 3 & 6 \\ s & \omega & \omega & \omega & 0 & 6 \\ f & \omega & \omega & \omega & \omega & 0 \end{array} \quad A^3 = \begin{array}{c|ccccc} & i & p & q & s & f \\ \hline i & 0 & 4 & 4 & 4 & 6 \\ p & \omega & 0 & 4 & 4 & 6 \\ q & \omega & 3 & 0 & 3 & 6 \\ s & \omega & \omega & \omega & 0 & 6 \\ f & \omega & \omega & \omega & \omega & 0 \end{array}$$

A^3 is the fixpoint. Now, $\min\{\mu(y, p) \mid W(y) < W(p)\} = 3$ and $\min\{\mu(x, y) \mid W(y) < W(p)\} = 4$, so the net is not sound. Moreover, $\min\{\mu(x, y)\} = 4$, and thus we can use the construction from the proof of Lemma 17 to reproduce a deadlock from this net (see Example 18).

Observe that the computation proposed strongly resembles the All-Pairs Shortest Paths problem (Floyd-Warshal algorithm, see [7]; for a more efficient algorithm see [15]; also see [16] for a survey). Hence, our computation can benefit from efficient matrix multiplication algorithms. Moreover, to decrease the number of multiplication steps we can repeatedly square the result of the previous step, i.e., instead of $A, A^2, A^3, \ldots$ we compute $A, A^2, A^4, \ldots$. The number of multiplication steps is logarithmic in the length of the longest acyclic path of the net.

Corollary 26. *For an SM1WF-net with P places and T transitions the soundness decision algorithm presented above has complexity of $O(P^3 \log P + T)$.*

6 Conclusion

We have introduced an extension of Workflow nets: *Resource-Constrained Workflow nets* and defined a notion of soundness on this class of nets, which is an extension of the soundness notion for WF-nets. In addition to the soundness requirements for WF-nets, soundness for RCWF-nets states that no resources are created during the processing and all resources are returned to their resource place when the processing is completed; moreover, no deadlock or livelock can arise due to the lack of resources. We showed how to reduce the problem of soundness for a general class of RCWF-nets with one resource type to the problem of soundness for SM1WF-nets and gave a necessary and sufficient condition of soundness for SM1WF-nets. The decision algorithm we described has a polynomial complexity w.r.t. the size of the production net of the SM1WF-net.

Future work We have considered here the problem of soundness for RCWF-nets with one resource type. Finding a necessary and sufficient condition of soundness for RCWF-nets with multiple resource types is left for the future work. Another direction for future research is to find a method to transform a given unsound RCWF-net into a sound one by applying modifications of one type only: transitions may claim and release more resources than in the original situation.

Future work includes also the integration of our algorithm into tools working with this class of nets.

References

1. W. M. P. van der Aalst. Verification of workflow nets. In P. Azéma and G. Balbo, editors, *Application and Theory of Petri Nets 1997, ICATPN'1997*, volume 1248 of *Lecture Notes in Computer Science*. Springer-Verlag, 1997.
2. W. M. P. van der Aalst. The Application of Petri Nets to Workflow Management. *The Journal of Circuits, Systems and Computers*, 8(1):21–66, 1998.
3. W. M. P. van der Aalst. Workflow verification: Finding control-flow errors using Petri-net-based techniques. In W. M. P. van der Aalst, J. Desel, and A. Oberweis, editors, *Business Process Management: Models, Techniques, and Empirical Studies*, volume 1806 of *Lecture Notes in Computer Science*, pages 161–183. Springer-Verlag, 1999.
4. W. M. P. van der Aalst and K. M. van Hee. *Workflow Management: Models, Methods, and Systems*. MIT Press, 2002.
5. K. Barkaoui and L. Petrucci. Structural analysis of workflow nets with shared resources. In *Workflow management: Net-based Concepts, Models, Techniques and Tools (WFM'98)*, volume 98/7 of *Computing science reports*, pages 82–95. Eindhoven University of Technology, 1998.
6. J. Colom. The resource allocation problem in flexible manufacturing systems. In W. van der Aalst and E. Best, editors, *Application and Theory of Petri Nets 2003, ICATPN'2003*, volume 2679 of *Lecture Notes in Computer Science*, pages 23–35. Springer-Verlag, 2003.
7. T. H. Cormen, C. E. Leiserson, and R. L. Rivest. *Introduction to Algorithms*. MIT Press, 1990.
8. E. W. Dijkstra. Ewd 623. *Selected writings on computing: a personal perspective*, 1982.
9. J. Ezpeleta. Flexible manufacturing systems. In C. Girault and R. Valk, editors, *Petri nets for systems engineering*. Springer-Verlag, 2003.
10. J. Ezpeleta, J. M. Colom, and J. Martínez. A Petri net based deadlock prevention policy for flexible manufacturing systems. *IEEE Transactions on Robotics and Automation*, 11(2):173–184, 1995.

11. K. van Hee, N. Sidorova, and M. Voorhoeve. Soundness and separability of workflow nets in the stepwise refinement approach. In W. van der Aalst and E. Best, editors, *Application and Theory of Petri Nets 2003, ICATPN'2003*, volume 2679 of *Lecture Notes in Computer Science*, pages 337–356. Springer-Verlag, 2003.
12. K. Lautenbach. *Liveness in Petri Nets.* Internal Report of the Gesellschaft für Mathematik und Datenverarbeitung, Bonn, Germany, ISF/75-02-1, 1975.
13. M. Silva and E. Teruel. Petri nets for the design and operation of manufacturing systems. *European Journal of Control*, 3(3):182–199, 1997.
14. M. Silva and R. Valette. Petri nets and flexible manufacturing. In G. Rozenberg, editor, *Applications and Theory of Petri Nets*, volume 424 of *Lecture Notes in Computer Science*, pages 374–417. Springer, 1990.
15. T. Takaoka. Subcubic cost algorithms for the all pairs shortest path problem. *Algorithmica*, 3(20):309–318, 1998.
16. U. Zwick. Exact and approximate distances in graphs – a survey. In F. Meyer auf der Heide, editor, *Algorithms – ESA 2001, 9th Annual European Symposium, Aarhus, Denmark, August 28-31, 2001, Proceedings*, Lecture Notes in Computer Science, pages 33–48. Springer Verlag, 2001.