

Resource-Constrained Workflow nets

Kees van Hee

Natalia Sidorova ^C

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@tue.nl, n.sidorova@tue.nl, m.voorhoeve@tue.nl

Abstract. We study concurrent processes modelled as workflow Petri nets extended with resource constraints. Resources are durable units that can be neither created nor destroyed: they are claimed during the handling procedure and then released again. Typical kinds of resources are manpower, machinery, computer memory. We define structural criteria based on traps and siphons for the correctness of workflow nets with resource constraints. We also extend the soundness notion for workflow nets to the workflow nets with resource constraints; extra conditions concern the durability of resources. We prove some properties of sound resource-constrained workflow nets.

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 workflows 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. [2]).

Workflow systems are modelled by so-called *Workflow Nets (WF-nets)*, 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

^CCorresponding author

of a single token on the initial place. The token on the final place with no tokens (garbage) 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.

Originally, WF-nets were intended to model the execution of a single case. In [8, 9] we considered WF-nets modelling the execution of batches of cases in WF-nets, where along with standard for WF-nets transitions/subnets that process one case per time there are transitions/subnets allowing a (faster, cheaper, etc.) processing of several cases at the same time. We defined the notion of generalised soundness: “States reachable after starting with k tokens on the initial place will be able to reach the state with only k tokens on the final place, for any natural number k ” and showed that the generalised soundness is decidable.

WF-nets are meant to emphasise the partial ordering of activities in the process while abstracting from *resources* (e.g. machines or personnel) which may further restrict the occurrence of activities. In this paper (which is an extended version of [10]) we consider the influence of *resources* on the processing of cases in Workflow Nets. Resources are durable, i.e. they are claimed and released during the execution, but they cannot be created or destroyed. We concentrate here on fundamental correctness requirements for Resource-Constrained Workflow nets (RCWF-nets): no redundancy in system design, resource conservation laws (every claimed resource is freed before the case terminates and no resource is created during processing), and the absence of deadlocks or livelocks that occur due to the lack of resources. We introduce some *structural* correctness criteria for RCWF-nets, extend the notions of soundness to RCWF-nets and give necessary conditions for soundness expressed in terms of net invariants.

The rest of the paper is organised as follows. In Section 2, we sketch basic definitions related to Petri nets and Workflow nets. In Section 3 we introduce the notion of Resource-Constrained Workflow Nets and consider some structural correctness criteria for them. In Section 4 we define and investigate the notion of soundness for RCWF-nets. We conclude in Section 5 by discussing obtained results, related work and directions for future work.

2. Preliminaries

\mathbb{N} denotes the set of natural numbers, \mathbb{Z} the set of integers and \mathbb{Q} the set of rational numbers.

Let P be a set. A *bag* (*multiset*) m over P is a mapping $m : P \rightarrow \mathbb{N}$. 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 $p \in m$ when $m(p) > 0$. We write e.g. $m = k[p] + \ell[q]$ for a bag m with $m(p) = k$, $m(q) = \ell$, and $m(x) = 0$ for all $x \notin \{p, q\}$. For a sum over the elements of a bag m we write $\sum_{p \in m} f(p)$ (assuming that every p appears in the sum $m(p)$ times) rather than $\sum_{p \in P} m(p) \cdot f(p)$. We write $m|_Q$ for the projection of bag m on $Q \subseteq P$; formally, $m|_Q$ is a bag over Q such that $m|_Q(p) = m(p)$ for every $p \in Q$.

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. The *Parikh vector* $\vec{\sigma}$ of a sequence σ maps every element $t \in T$ to the number of occurrences of t in σ , denoted by $\vec{\sigma}(t)$.

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 $\langle E, s_0 \rangle$ where E is a transition system and $s_0 \in S$ an initial state. We denote (s_1, a, s_2) from T as $s_1 \xrightarrow{a} s_2$, and

we say that a leads from s_1 to s_2 . For a sequence of transitions $\sigma = t_1 \dots t_n$ we write $s_1 \xrightarrow{\sigma} s_2$ when $s_1 = s^0 \xrightarrow{t_1} s^1 \xrightarrow{t_2} \dots \xrightarrow{t_n} s^n = s_2$, and $s_1 \xrightarrow{\sigma}$ when $s_1 \xrightarrow{\sigma} s_2$ for some 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 $s_1 \xrightarrow{\sigma} s_2$. To indicate that the step a is taken in the transition system E we write $s \xrightarrow{a}_E s'$.

Given two transition systems $N_1 = \langle S_1, Act, T_1 \rangle$ and $N_2 = \langle S_2, Act, T_2 \rangle$. A relation $R \subseteq S_1 \times S_2$ is a *simulation* iff for all $s_1, s'_1 \in S_1, s_2 \in S_2, s_1 R s_2$ and $s_1 \xrightarrow{a}_1 s'_1$ implies that there exists $s'_2 \in S_2$ such that $s'_1 R s'_2$ and $s_2 \xrightarrow{a}_2 s'_2$.

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; elements of the set $P \cup T$ are called the *nodes* of N ;
- F^+ and F^- are mappings $(P \times T) \rightarrow \mathbb{N}$ 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 depict nets by the usual graphical notation.

To project away some places of the net together with their in- and outgoing arcs we define the projection operation:

Definition 2.1. (projection)

Let $N = \langle P, T, F^+, F^- \rangle$ be a Petri net and let $P' \subseteq P$. The *projection* $N|_{P'}$ of N w.r.t. P' is the net $\langle P', T, F^+|_{(P' \times T)}, F^-|_{(P' \times T)} \rangle$.

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. To emphasize the fact that the preset/postset is considered within some net N , we write $\bullet_N a, a_N \bullet$. We overload this notation further and apply preset and postset operations to a set B of places: $\bullet B = \{t \mid \exists p \in B : t \in \bullet p\}$ and $B \bullet = \{t \mid \exists p \in B : t \in p \bullet\}$. Note that $\bullet B$ and $B \bullet$ are not bags but sets. We will say that node n is a *source* node iff $\bullet n = \emptyset$ and n is a *sink* node iff $n \bullet = \emptyset$. A *path* of a net is a sequence $x_0 \dots x_n$ of nodes such that $\forall i : 1 \leq i \leq n : x_{i-1} \in \bullet x_i$.

A marking m of N is a bag over P ; 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 $m + \bullet t \xrightarrow{t} m + t \bullet$. The notion of reachability for Petri nets is inherited from transition systems. For a firing sequence σ in a net N , we define $\bullet \sigma$ and $\sigma \bullet$ respectively as $\sum_{t \in \sigma} \bullet t$ and $\sum_{t \in \sigma} t \bullet$, which are the sums of all tokens consumed/produced during the firings of σ . So if $m \xrightarrow{\sigma} m'$, then $m' = m - \bullet \sigma + \sigma \bullet$. We will use the well-known *Marking Equation Lemma*:

Lemma 2.1. (Marking Equation)

Given a finite firing sequence σ of a net N : $m \xrightarrow{\sigma} m'$, the following equation holds: $m' = m + F^+ \cdot \vec{\sigma} - F^- \cdot \vec{\sigma}$, or in other words, $m' = m + F \cdot \vec{\sigma}$.

Note that the reverse is not true: not every marking m' that is representable as a sum $m + F \cdot v$ for some $v \in \mathbb{N}^T$ is reachable from the marking m . We will write $F.X$ for the set of vectors $\{F \cdot x \mid x \in X\}$.

We denote the set of all markings reachable in net N from marking m as $\mathcal{R}(N, m)$. We omit N and write $\mathcal{R}(m)$ when no ambiguities can arise.

Traps and Siphons (see [5]) A set R of places is a *trap* if $R^\bullet \subseteq {}^\bullet R$. A set R of places is a *siphon* if ${}^\bullet R \subseteq R^\bullet$. The trap/siphon is a *proper trap/siphon* iff it is not empty. As follows from the definition, traps and siphons are dual by their nature. Important properties of traps and siphons are that *marked traps remain marked* and *unmarked siphons remain unmarked* whatever transition firings would happen.

Invariants (see [11]) A *place invariant* is a row vector $I : P \rightarrow \mathbb{Q}$ such that $I \cdot F = 0$. When talking about invariants, we consider markings as *vectors*, where we assume arbitrary but fixed orderings of places and transitions. We denote the set of all place invariants as \mathcal{I}_N , which is a linear subspace of \mathbb{Q}^P . The following properties follow directly from the definition of place invariants:

Lemma 2.2. \mathcal{I}_N is orthogonal to $F \cdot \mathbb{Q}^T$.

Lemma 2.3. Let N be a Petri net with places P , let $P' \subseteq P$ and I' be a place invariant of $N|_{P'}$. Then I defined as $I(p) = I'(p)$ if $p \in P'$ and $I(p) = 0$ otherwise, is a place invariant of N .

The main property of place invariants is that for any two markings m_1, m_2 such that $m_1 \xrightarrow{*} m_2$ and any place invariant I holds: $I \cdot m_1 = I \cdot m_2$.

A *transition invariant* is a column vector $J : P \rightarrow \mathbb{Q}$ such that $F \cdot J = 0$. For any markings m, m' and firing sequences σ, γ , if $m \xrightarrow{\sigma} m'$ and $m \xrightarrow{\gamma} m'$, then $\vec{\sigma} - \vec{\gamma}$ is a transition invariant. This also means that for any firing sequence σ such that $m \xrightarrow{\sigma} m$, $\vec{\sigma}$ is a transition invariant.

Workflow Petri nets In this paper we primarily focus upon *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 2.2. 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 introduced in [8]:

Definition 2.3. We say that a marking $m \in \mathcal{R}(k[i])$ in a WF-net N *terminates properly* iff $m \xrightarrow{*} k[f]$. N is *k-sound* for some $k \in \mathbb{N}$ iff for all $m \in \mathcal{R}(k[i])$, m terminates properly. N is *sound* iff it is *k-sound* for all $k \in \mathbb{N}$.

We will use terms *initial* and *final* markings for markings $k[i]$ and $k[f]$ respectively ($k \in \mathbb{N}$).

3. Resource-Constrained Workflow Nets

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

A resource belongs to a type; we have one place per type in the net, where the resources are located when they are free. The resources become part of case tokens when they are occupied. We assume that resources are durable, i.e. they can be neither 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 3.1. We 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 $(P_p \times T) \rightarrow \mathbb{N}$,
- F_r^+ and F_r^- are mappings $(P_r \times T) \rightarrow \mathbb{N}$, and
- $N_p = \langle P_p, T, F_p^+, F_p^- \rangle$ is a WF-net, which we call a *production net* of N .

Recall that according to our notation for bags we write $m|_{P_p}$ for the projection of $m \in \mathbb{N}^P$ on production places and $m|_{P_r}$ for the projection of m on resource places, and N_p is actually $N|_{P_p}$.

By projecting away resource places of an RCWF-net, we obtain an RCWF-net again:

Lemma 3.1. Let $N = \langle (P_p \cup P_r), T, F^+, F^- \rangle$ be an RCWF-net. Then for any $P'_r \subseteq P_r$, the net $N' = N|_{P'_r}$ is an RCWF net.

Proof:

All the conditions are met, with P'_r being the set of resource places. The production net of N' is the same as the one of N . \square

Note that additional resource places only limit the behaviour of the net:

Lemma 3.2. Let $N = \langle (P_p \cup P_r), T, F^+, F^- \rangle$ be an RCWF-net and $P'_r \subseteq P_r$. Then $R = \{ (m, m|_{(P_p \cup P'_r)}) \mid m \in \mathbb{N}^{P_p \cup P_r} \}$ is a simulation relation between N and $N|_{(P_p \cup P'_r)}$.

Proof:

Let $P = P_p \cup P_r$, $P' = P_p \cup P'_r$ and $N' = N|_{(P_p \cup P'_r)}$. Suppose $m_1 \xrightarrow{t}_N m_2$ for some $m_1, m_2 \in \mathbb{N}^P$, thus $m_1 \geq \bullet_N t$ and $m_2 = m_1 - \bullet_N t + t_N$. Then also $m_1|_{P'} \geq \bullet_{N'} t$ and $m_2|_{P'} = m_1|_{P'} - \bullet_{N'} t + t_{N'}$. \square

By taking $P'_r = \emptyset$ we trivially obtain that $R = \{ (m, m|_{P_p}) \mid m \in \mathbb{N}^{(P_p \cup P_r)} \}$ is a simulation relation for N and N_p .

Next we will discuss *structural correctness criteria* for WF-nets based on traps and siphons and then show how these criteria can be adapted to the RCWF-nets.

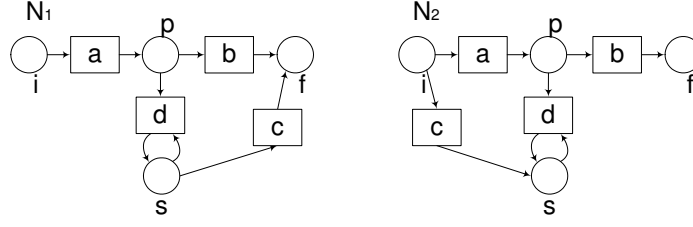


Figure 1. Redundant and persistent places

3.1. Redundant and Persistent Places

In [9] we introduced notions of redundant and persistent places in WF-nets and showed how to find them with the use of siphons and traps. Here we extend the results from [9] and use the notions of redundancy and persistency to analyse structural correctness of RCWF-nets.

A natural requirement for the correct design of an RCWF-net is *non-redundancy* of the *production net*, namely: every transition of the net can potentially fire and every place of the production net can potentially obtain tokens, provided that there are enough tokens on the initial place i and enough resource tokens. Production net N_1 in Fig. 1 does not satisfy this requirement because transition d can never fire and place s can never get tokens. So d and s are *redundant*. The *resource places* are, contrary to the production places, *redundant* by their nature, since resource tokens cannot be created by the production net and should be present in the initial marking of the RCWF-net.

On the other hand, it should be possible for all places of the *production net* (except for f) to become unmarked again—otherwise the net is guaranteed to leave garbage after processing, as e.g. production net N_2 in Fig. 1—place s can obtain tokens but it can never become unmarked after that, i.e. this place is *persistent*. Similarly, there should be no persistent transitions in the production net, i.e. transitions producing a token to a non-final place of the production net which cannot be “moved” to a final place later on. The *resource places*, on the other hand, are *persistent*, since every claimed resource should be released before the production process is completed. In formal terms:

Definition 3.2. Let $N = \langle P, T, F \rangle$ be a WF-net.

A place $p \in P$ is *non-redundant* iff there exist $k \in \mathbb{N}$ and $m \in \mathbb{N}^P$ such that $k[i] \xrightarrow{*} m \wedge p \in m$.

A place $p \in P$ is *non-persistent* iff there exist $k \in \mathbb{N}$ and $m \in \mathbb{N}^P$ such that $p \in m \wedge m \xrightarrow{*} k[f]$.

A transition t is *non-redundant* iff there exist $k \in \mathbb{N}$ and $m \in \mathbb{N}^P$ such that $k[i] \xrightarrow{*} m \xrightarrow{t}$.

A transition t is *non-persistent* iff there exist $k \in \mathbb{N}$ and $m, m' \in \mathbb{N}^P$ such that $m \xrightarrow{t} m' \xrightarrow{*} k[f]$.

The following lemma presents these desirable behavioural properties in more general terms:

Lemma 3.3. (1) A WF-net N has no redundant places iff every marking is majorated by a marking reachable from some initial marking $k[i]$, i.e.

$$\forall m \in \mathbb{N}^P : \exists k \in \mathbb{N}, m' \in \mathcal{R}(k[i]) : m' \geq m.$$

(2) A WF-net N has no persistent places iff every marking is majorated by a marking from which some final marking $k[f]$ is reachable, i.e.

$$\forall m \in \mathbb{N}^P : \exists k \in \mathbb{N}, m' \in \mathbb{N}^P : m' \xrightarrow{*} k[f] \wedge m' \geq m.$$

Proof:

(1) If every marking can be majorated by a marking reachable from some $k[i]$, then every marking $[p]$, $p \in P$, can be majorated and p is non-redundant. In the opposite direction: suppose that for every p there exist k_p, m_p , such that $k_p[i] \xrightarrow{*} m_p$ where $m_p \geq [p]$. Then we can majorate a given marking m by a marking $m' = \sum_{p \in m} m_p$ reachable from $(\sum_{p \in m} k_p)[i]$.

(2) can be proved similarly. \square

As an immediate consequence we obtain the following property:

Lemma 3.4. (1) A WF-net N has no redundant places iff it has no redundant transitions. (2) A WF-net N has no persistent places iff it has no persistent transitions.

Proof:

(1) Let N have no redundant places. Consider an arbitrary transition $t \in T$. By applying property (1) of Lemma 3.3 to $\bullet t$ we obtain that t can become enabled, and hence it is non-redundant.

Now assume that N has no redundant transitions. Consider an arbitrary place $p \in P \setminus \{i\}$. Since N is a WF-net, $\bullet p \neq \emptyset$, and since all transitions are non-redundant, transitions from $\bullet p$ can fire and so p can get marked. Thus p is non-redundant.

(2) can be proved similarly. \square

3.2. Structural Correctness Requirements for RCWF-Nets

Non-redundancy and non-persistency are behavioural properties. They imply though the following restrictions on the structure of the net: all proper siphons of the net should contain i and all proper traps should contain f . If N contained a proper siphon without i , the transitions consuming tokens from places of that siphon would be dead, no matter how many tokens are put into i . Similarly, if N contained a trap without f , the net could not terminate properly. It is not surprising that the absence of traps and siphons is a necessary condition for the correctness of the design. What is more interesting is that the absence of such siphons and traps is a *sufficient* condition for the absence of redundant and persistent places respectively: if a net has a redundant place, there exists a proper siphon without i , and if a net has a persistent place, there exists a proper trap without f , i.e. these behavioural and structural characteristics are equivalent for WF-nets [9]:

Theorem 3.1. Let $N = \langle P, T, F \rangle$ be a WF-net. Then the following holds:

(1) p is a redundant place iff it belongs to a siphon $X \subseteq (P \setminus \{i\})$.

(2) p is a persistent place iff it belongs to a trap $X \subseteq (P \setminus \{f\})$.

Proof:

(1) Let $X \subseteq P \setminus \{i\}$ be a proper siphon. Since an unmarked siphon stays unmarked, places from X are redundant.

In the opposite direction: Let $X \subseteq P \setminus \{i\}$ be the set of all redundant places of N . We will prove that X is a siphon. Consider some $t \notin X^\bullet$; $\bullet t$ contains no places from X and hence all places from $\bullet t$ are non-redundant. Then for every place p in $\bullet t$ there exists a marking $m_p \geq [p]$ reachable from some $k_p[i]$, $k_p \in \mathbb{N}$. Taking a sum of corresponding initial markings we obtain an initial marking from which a marking $m \geq \bullet t$ can be reached. Thus t can fire and all places from t^\bullet can obtain tokens, i.e. they are

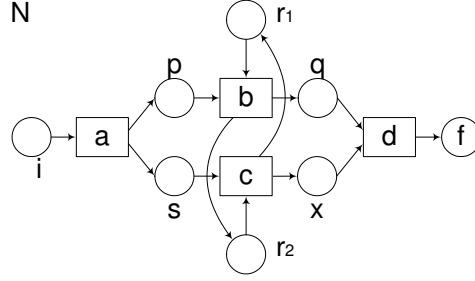


Figure 2. An RCWF-net with dependent resource places

non-redundant. Therefore, $t^\bullet \cap X = \emptyset$ and so $t \notin {}^\bullet X$. Hence $(T \setminus X^\bullet) \subseteq (T \setminus {}^\bullet X)$, and so X is a siphon. Thus all redundant places belong to a proper siphon in $P \setminus \{i\}$.

(2) can be proved analogously. \square

One can compute the largest siphon X in $P \setminus \{i\}$ in a standard manner [13]: initialize X with $P \setminus \{i\}$ and remove the places that belong to t^\bullet for some t such that $t \notin X^\bullet$ until the fixed point is reached. The largest trap not containing f can be computed with a similar algorithm.

Thus, let an RCWF-net N with an underlying production net N_p be given. To check the structural correctness requirements on N , we first check that the production net N_p has no redundant and persistent places, i.e. there is no siphon in $(P_p \setminus \{i\})$ and there is no trap in $(P_p \setminus \{f\})$. If redundant or persistent places are found, the error is reported to a designer. The production net without redundant and persistent places does not have redundant or persistent transitions either. Note that by projecting out redundant places and transition of the production net, we do not change the behaviour of the RCWF-net. Presence of persistent places and transitions in the production net indicates problems in the net design, since there are possible executions that will leave garbage on persistent places (if these places are not redundant at the same time as well). We will call the production net that has neither redundant nor persistent places *structurally correct*.

Next, we check that all resource places are redundant *and* persistent in net N . If this is not the case, there is an error in the design: resources can be created or destroyed during the processing. If the design is correct w.r.t. this criterion, we can proceed further with using different interpretations of “design correctness”, depending on whether the resources are supposed to be *independent* or not. From the modelling point of view, resource dependence means that resource items may render to resource items of another type during the processing; in many cases resource dependencies do not correspond to the real system behaviour and indicate design errors (manpower cannot be transformed into machinery and vice versa). We illustrate the notion of resource dependence with net N in Figure 2. A firing of transition b moves a resource from the resource place r_1 to the resource place r_2 while firing of transition c moves a resource from the resource place r_2 to the resource place r_1 . Thus, during the processing of a task in the net $(N, [i] + [r_1] + [r_2])$ resources r_1 and r_2 trade places. We call such resources *dependent*. Note that r_1 (as well as r_2) is redundant and persistent in net N , but it is neither redundant nor persistent in the net obtained from N by removing place r_2 (r_1 respectively) together with its in- and outgoing arcs.

Definition 3.3. We will say that a resource r is *independent* of other resources in an RCWF-net $N =$

$\langle P_p \cup P_r, T, F_p^+ \cup F_r^+, F_p^- \cup F_r^- \rangle$ with a structurally correct production net N_p iff $N|_{P_p \cup \{r\}}$ is an RCWF-net where place r is a resource place again, i.e. it is both redundant and persistent.

We expect the designer to indicate which resource places in the net are supposed to model independent resources; the check whether resources are independent indeed can be easily done by calculating traps and siphons. In the rest of the paper we suppose that all resources places in RCWF-nets are independent.

To summarize:

Definition 3.4. An RCWF-net N is *structurally correct* iff

- the production net $N|_{P_p}$ has no redundant or persistent places, and
- all resources $r \in P_r$ are independent of other resources in N .

4. Soundness of Resource-Constrained Workflow Nets

Soundness in WF-nets is the property 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 RCWF-nets, the initial marking of the net is a marking with some tokens on the initial place and resource places. Proper termination assumes that the resource tokens are back to their resource places and all tasks have been processed correctly, i.e. all the places of N_p except for f are empty and the number of tokens on f is the same as we had on i initially. Moreover, the net should behave properly not only with some fixed amount of resources but also with any greater amount: we want a sound system to work correctly also when more money, manpower, or machinery is available. On the other hand, it is clear that some minimal amount of resources is needed for the system to work at all.

Another correctness requirement that should be reflected by the definition of soundness is that resource tokens cannot be created during processing, i.e. the number of available resources does not exceed the number of initially given resources at any moment of time. This requirement is related to the requirement of resource independence: Consider a firing sequence $[i] + [r_1] + [r_2] \xrightarrow{ab} [s] + [q] + 2[r_2]$ in net N in Figure 2, the number of resources of type r_2 has increased in comparison to the initial number, which contradicts the assumption that resources cannot be created, also not from other resources.

The extended definition of soundness thus reads as follows:

Definition 4.1. (soundness)

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 $m \in \mathcal{R}(k[i] + R), m \xrightarrow{*} (k[f] + R)$ and $m|_{P_r} \leq R$.

N is k -sound iff there exists $R_0 \in \mathbb{N}^{P_r}$ such that N is (k, R) -sound for all $R \geq R_0$.

N is sound iff there exists $R_0 \in \mathbb{N}^{P_r}$ such that N is (k, R) -sound for all $k \in \mathbb{N}, R \geq R_0$.

Soundness of WF-nets becomes thus a special case of soundness of RCWF-nets (by taking the empty set of resource places). Note that any (finite) firing sequence of the production net is possible in the RCWF-net if we take a sufficiently large resource marking. Since we require a sound RCWF-net to work properly for *all* “large” resource markings, all nets obtained by projecting out resource places have to be sound as well:

Theorem 4.1. Let N be an RCWF net and let $P'_r \subseteq P_r$. Then: (1) If N is k -sound, then $N \upharpoonright_{(P_p \cup P'_r)}$ is k -sound too; (2) If N is sound, then $N \upharpoonright_{(P_p \cup P'_r)}$ is sound too.

Proof:

We only prove (1), as (2) is fully analogous. Set $N' = N \upharpoonright_{(P_p \cup P'_r)}$. Since N is k -sound, there exists $R_0 \in \mathbb{N}^{P_r}$ such that N is (k, R) -sound for each $R \geq R_0$. Choose $R \geq R_0$, and let $R' = R \upharpoonright_{P'_r}$ and $k[i] + R' \xrightarrow{\sigma}_{N'} m'$. Then we can find $R_1 \in \mathbb{N}^{P_r \setminus P'_r}$, $R_1 \geq R - R'$, such that $k[i] + R' + R_1 \xrightarrow{\sigma}_N m' + R_2$ for some marking R_2 . Note that $R_2 \in \mathbb{N}^{P_r \setminus P'_r}$, since $\sigma_{N'} - \bullet_{N'} \sigma = (\sigma_N - \bullet_N \sigma) \upharpoonright_{(P_p \cup P'_r)}$. Since N is k -sound and $R' + R_1 \geq R \geq R_0$, we have $m' + R_2 \xrightarrow{*}_N k[f] + R' + R_1$. By Lemma 3.2, $m' \xrightarrow{*}_{N'} k[f] + R'$. Moreover, by soundness of N , $m' \upharpoonright_{P'_r} + R_2 \leq R' + R_1$. Since $R_1, R_2 \in \mathbb{N}^{P_r \setminus P'_r}$ and $m' \upharpoonright_{P'_r} \in P_r$, $m' \upharpoonright_{P'_r} \leq R'$. So N' is k -sound. \square

By considering $N \upharpoonright_{P_p}$ we obtain a necessary condition of soundness for RCWF-nets:

Corollary 4.1. (necessary condition 1)

If N is a sound RCWF-net, its production net $N_p = N \upharpoonright_{P_p}$ is sound, too.

Thus soundness of the underlying production net is a necessary condition for soundness of an RCWF-net. We do not discuss the decision procedure for soundness of WF-nets here but refer the interested reader to [9], where soundness of WF-nets is proved to be decidable and a decision procedure is given.

4.1. No-Resource-Creation Condition

Another requirement a sound RCWF-net should meet is that there is no resource token creation, i.e. there is a resource marking R_0 such that for any larger resource marking $R \geq R_0$, any reachable marking $m \in \mathcal{R}(k[i] + R)$ ($k \in \mathbb{N}$) has at most R resource tokens. It is however impossible to check this property directly since we have infinitely many markings in $\bigcup_{k \in \mathbb{N}} \mathcal{R}(k[i] + R)$, and sets of all reachable markings $\mathcal{R}(k[i] + R)$ do not have simple algebraic characteristics. Therefore we will try and prove that the required property holds on markings from $\bigcup_{k \in \mathbb{N}} \mathcal{R}(k[i] + R)$ iff it holds for all markings from a set with a simple algebraic structure. For this purpose we introduce a notion of extended reachability:

Definition 4.2. (extended reachability)

The *extended reachability relation* $\rightsquigarrow \subseteq \mathbb{N}^P \times \mathbb{N}^P$ between markings of an RCWF-net N is defined by $m \rightsquigarrow m' \Leftrightarrow \exists \ell \in \mathbb{N}, R \in \mathbb{N}^{P_r} : m + \ell[i] + R \xrightarrow{*} m' + \ell[f] + R$.

Note that $\xrightarrow{*} \subseteq \rightsquigarrow$ (take $\ell = 0$ and $m_r = \emptyset$). We illustrate the meaning of extended reachability on net N in Figure 3.¹ Consider markings $[i]$ and $[q] + [r]$; $[i] \not\xrightarrow{*} [q] + [r]$, however $[i] + [i] + 4[r] \xrightarrow{t^2 u} [q] + [r] + [f] + 4[r]$, and thus, according to Definition 4.2, $[i] \rightsquigarrow [q] + [r]$. Note that after the firing of $t^2 u$, the number of resource tokens exceeds the initial number of resource tokens, which shows that the RCWF-net is not sound. In \rightsquigarrow we observe an increase of the number of resource tokens as well.

For sound RCWF-nets, \rightsquigarrow has a regular algebraic structure: it turns out to be equality modulo the F -lattice:

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

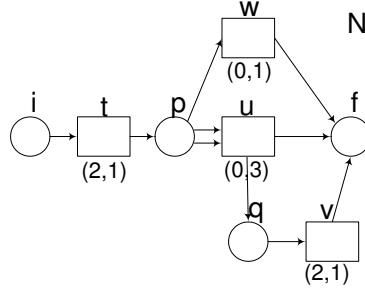


Figure 3. Extended reachability: $[i] \rightsquigarrow [q]$

Theorem 4.2. Let N be a sound RCWF-net without redundant places in its production net and let $m, m' \in \mathbb{N}^P$. Then $m \rightsquigarrow m'$ iff $m' - m \in F \cdot \mathbb{Z}^T$, where $F \cdot \mathbb{Z}^T \stackrel{\text{def}}{=} \{F \cdot x \mid x \in \mathbb{Z}^T\}$.

Proof:

(\Rightarrow): Suppose $m, m' \in \mathbb{N}^P$ and $m \rightsquigarrow m'$. By Definition 4.2, there exist $\ell \in \mathbb{N}, R \in \mathbb{N}^{P_r}$ such that $m + \ell[i] + R \xrightarrow{\sigma} m' + \ell[f] + R$ for some firing sequence σ . By Lemma 2.1, $m' + \ell[f] + R = m + \ell[i] + R + F \cdot \vec{\sigma}$. Since N is sound, $\ell[f] + R' = \ell[i] + R' + F \cdot x$ for some $x \in \mathbb{N}^T, R' \in \mathbb{N}^{P_r}$ (Lemma 2.1). Hence, $m' = m + F \cdot (\vec{\sigma} - x)$ and thus $m' - m \in F \cdot \mathbb{Z}^T$.

(\Leftarrow): Suppose $(m' - m) \in F \cdot \mathbb{Z}^T$, so there exist $x, y \in \mathbb{N}^T$ such that $m' - m = (F^+ - F^-) \cdot (x - y)$. Thus, $m + F^+ \cdot x + F^- \cdot y = m' + F^- \cdot x + F^+ \cdot y$.

Since N_p has no redundant places, by Lemma 3.3 we can find $\ell > 0, m_1 \in \mathbb{N}^{P_p}$ such that $\ell[i] \xrightarrow{*}_{N_p} F_p^- \cdot (x + y) + m_1$. By taking enough resources $R \in \mathbb{N}^{P_r}$, we obtain $\ell[i] + R \xrightarrow{*}_N F^- \cdot (x + y) + m_1 + m_2$ for some $m_2 \in \mathbb{N}^{P_r}$. Note that every firing sequence σ with Parikh vector y is enabled in $F^- \cdot (x + y) + m_1 + m_2$, and $F^- \cdot (x + y) + m_1 + m_2 \xrightarrow{\sigma}_N F^- \cdot x + F^+ \cdot y + m_1 + m_2$. Since N is sound and $\ell[i] + R \xrightarrow{*}_N F^- \cdot x + F^+ \cdot y + m_1 + m_2$, we deduce $F^- \cdot x + F^+ \cdot y + m_1 + m_2 \xrightarrow{*}_N \ell[f] + R$.

On the other hand, for a firing sequence γ with $\vec{\gamma} = x$, we have $m + \ell[i] + R \xrightarrow{*}_N m + F^- \cdot (x + y) + m_1 + m_2 \xrightarrow{\gamma}_N m + F^+ \cdot x + F^- \cdot y + m_1 + m_2 = m' + F^- \cdot x + F^+ \cdot y + m_1 + m_2$. Since $F^- \cdot x + F^+ \cdot y + m_1 + m_2 \xrightarrow{*}_N \ell[f] + R$, we obtain $m + \ell[i] + R \xrightarrow{*}_N m' + \ell[f] + R$, i.e. $m \rightsquigarrow m'$. \square

Going back to net N in Figure 3, we choose the ordering (i, p, q, f, r) of places and (t, u, v, w) of transitions for the vector/matrix representation. It is easy to check that for $[i] \rightsquigarrow [q] + [r]$ we have indeed:

$$\begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} -1 & 0 & 0 & 0 \\ 1 & -2 & 0 & -1 \\ 0 & 1 & -1 & 0 \\ 0 & 1 & 1 & 1 \\ -1 & 3 & -1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 0 \\ -1 \\ 1 \end{pmatrix}$$

Having shown that the set of markings related by \rightsquigarrow to a given marking enjoys a regular algebraic structure, we will show that for sound RCWF-nets the number of resource tokens does not exceed the initial number of tokens not only for all the markings m such that $k[i] + R \xrightarrow{*} m$ ($k \in \mathbb{N}, R \geq R_0$), but also for the markings m' such that $k[i] + R \rightsquigarrow m'$, which will give us a necessary condition for the “non-creation” of resources in RCWF-nets.

Theorem 4.3. Let N be a sound RCWF-net without redundant places in its production net and $R_0 \in \mathbb{N}^{P_r}$ be a minimal resource marking such that N is (k, R) -sound for any $k \in \mathbb{N}, R \geq R_0$. Then for any marking m such that $k[i] + R \rightsquigarrow m$, where $k \in \mathbb{N}, R \geq R_0$, we have $m|_{P_r} \leq R$.

Proof:

Let m be a marking such that $k[i] + R \rightsquigarrow m$ for some $k \in \mathbb{N}, R \geq R_0$. By Definition 4.2, there exist $\ell \in \mathbb{N}, R' \in \mathbb{N}^{P_r}$ such that $k[i] + R + \ell[i] + R' \xrightarrow{*} m + \ell[f] + R'$. Since N is sound, $(m + \ell[f] + R')|_{P_r} \leq R + R'$, which implies that $m|_{P_r} \leq R$. \square

Now we can formulate another necessary condition of soundness:

Theorem 4.4. (necessary condition 2)

Let N be a sound RCWF-net without redundant places in its production net. Then for all $x \in \mathbb{Z}^T$, $(F \cdot x)|_{(P_p \setminus \{i\})} \geq 0$ implies that $(F \cdot x)|_{P_r} \leq 0$.

Proof:

Consider a vector $x \in \mathbb{Z}^T$ such that $(F \cdot x)|_{(P_p \setminus \{i\})} \geq 0$. Then there exist $k \in \mathbb{N}, R \in \mathbb{N}^{P_r}$ such that $k[i] + R + F \cdot x \geq 0$, i.e. $k[i] + R + F \cdot x$ is a marking. Then by Theorem 4.2, $k[i] + R \rightsquigarrow k[i] + R + F \cdot x$. By Theorem 4.3, $(k[i] + R + (F \cdot x))|_{P_r} \leq R$, which immediately implies that $(F \cdot x)|_{P_r} \leq 0$. \square

Necessary condition 2 is in fact a requirement that all solutions of one inequality (forming a convex polyhedral cone) are also solutions of another inequality, or in other words, one cone is a subset of another cone, which can be checked by standard algebraic techniques.

If necessary condition 2 holds for some RCWF-net N , it guarantees that the soundness condition of “no resource token creation” holds for N :

Theorem 4.5. (necessary condition 2 \Rightarrow no resource token creation)

Let N be an RCWF-net such that for all $x \in \mathbb{Z}^T$, $(F \cdot x)|_{(P_p \setminus \{i\})} \geq 0$ implies that $(F \cdot x)|_{P_r} \leq 0$. Then for any $k \in \mathbb{N}, R \in \mathbb{N}^{P_r}$ and any $m \in \mathcal{R}(k[i] + R)$ we have $m|_{P_r} \leq R$.

Proof:

Consider a marking $m \in \mathcal{R}(k[i] + R)$ where $k \in \mathbb{N}, R \in \mathbb{N}^{P_r}$. By Lemma 2.1, $m = k[i] + R + F \cdot x$ for some $x \in \mathbb{N}^T$. Since $m \geq 0$, $(F \cdot x)|_{(P_p \setminus \{i\})} \geq 0$, which implies that $(F \cdot x)|_{P_r} \leq 0$. Thus $m|_{P_r} = R + F \cdot x|_{P_r} \leq R$. \square

4.2. Proper Termination Condition

A consequence of the requirement to work correctly for all “large” markings is that any transition invariant of the closure of the production net is a transition invariant of the the closure of the RCWF-net N , where the *closure* of a WF-net is the net obtained by adding a closing transition t_c such that $\bullet t_c = [f]$ and $t_c^\bullet = [i]$ RCWF-net.

Theorem 4.6. (necessary condition 3)

Let N be a *sound* RCWF-net such that its production net N_p has no redundant transitions, and \overline{N} and \overline{N}_p be their respective closures. Then for any vector $x \in \mathbb{Z}^T$ holds:

$$\overline{F}_p \cdot x = 0 \Leftrightarrow \overline{F} \cdot x = 0. \quad (1)$$

Proof:

Note that $\bar{F} \cdot x = 0$ implies $\bar{F}_p \cdot x = 0$. So we only have to show the \Rightarrow implication. Suppose $\bar{F}_p \cdot x = 0$ for some $x \in \mathbb{N}^T$. Then $\bar{F} \cdot x \in \mathbb{Z}^{P_r}$ and we can find $m_1, m_2 \in \mathbb{N}^{P_r}$ such that $\bar{F} \cdot x = m_1 - m_2$. By Theorem 4.2, we have $m_1 \rightsquigarrow m_2$, thus there exist $\ell \in \mathbb{N}, R \in \mathbb{N}^{P_r}$ such that $m_1 + \ell[i] + R \xrightarrow{*} m_2 + \ell[f] + R$. Note that we can choose $R \geq R_0$ where R_0 is the minimal resource marking according to the definition of soundness. By soundness of N , also $m_1 + \ell[i] + R \xrightarrow{*} m_1 + \ell[f] + R$. Since m_1 and m_2 are resource markings, $m_1 = m_2$. Thus, $\bar{F} \cdot x = 0$. \square

Note that $\bar{F} \cdot x = 0 \Leftrightarrow (\bar{F}_p \cdot x = 0 \wedge \bar{F}_r \cdot x = 0)$. Thus, for any *sound* RCWF-net, the solution space of the equation $\bar{F}_p \cdot x = 0$ is a subset of the solution space of the equation $\bar{F}_r \cdot x = 0$. On the other hand, for any RCWF-net, if $\bar{F}_p \cdot x = 0 \Leftrightarrow \bar{F} \cdot x = 0$ holds we can conclude that if no deadlock or livelock caused by the lack of resources occurs, then the net terminates properly, i.e. all resources are returned to their places:

Theorem 4.7. (necessary condition 3 \Rightarrow resources are returned to their places upon termination)

Let N be an RCWF-net such that its production net N_p has no redundant transitions, and for the closure nets \bar{N} and \bar{N}_p holds that for any vector $x \in \mathbb{Z}^T$, $\bar{F}_p \cdot x = 0 \Leftrightarrow \bar{F} \cdot x = 0$. Then for any $k \in \mathbb{N}, R \in \mathbb{N}^{P_r}, m \in \mathbb{N}^P$, $k[i] + R \xrightarrow{*} k[f] + m$ implies $m = R$.

Proof:

Follows directly from Theorem 4.6 and Theorem 4.1. \square

4.3. Place invariant conditions

Now we are going to show that sound RCWF-nets possess “production” and “resource” place invariants, which is another necessary condition of soundness. Recall that \mathcal{I}_N denotes the set of all place invariants.

Lemma 4.1. If N is a sound RCWF-net, then each $I \in \mathcal{I}_N$ satisfies $I(i) = I(f)$.

Proof:

By choosing $R \in \mathbb{N}^{P_r}$ large enough, we have $[i] + R \xrightarrow{*} [f] + R$, so for any $I \in \mathcal{I}_N$ we have $I \cdot ([i] + R) = I \cdot ([f] + R)$ and hence $I(i) = I(f)$. \square

Lemma 4.2. Let N be a sound RCWF-net and $r \in P_r$. Then for every place $p \in P_p \cup P_r$ there is a place invariant I satisfying $I(p) = 1$.

Proof:

Let N be sound. Suppose all place invariants $I \in \mathcal{I}_N$ satisfy $I(p) = 0$. Then $[p]$ is orthogonal to \mathcal{I}_N , and by Lemma 2.2, $[p] \in F \cdot \mathbb{Q}^T$. Thus there exists $k \in \mathbb{Z}, k \neq 0$, such that $k[p] \in F \cdot \mathbb{Z}^T$. By Theorem 4.2, $\emptyset \rightsquigarrow k[p]$, therefore there exist ℓ, R such that $\ell[i] + R \xrightarrow{*} k[p] + \ell[f] + R$, which is only possible if $k = 0$. This is a contradiction, and so there exists a place invariant I such that $I(p) \neq 0$. By scalar multiplication, we obtain I with $I(p) = 1$. \square

Theorem 4.8. (necessary condition 4)

If N is a sound RCWF-net, there exists a place invariant I_p such that $I_p(i) = I_p(f) = 1$ and $I_p(r) = 0$ for each $r \in P_r$, which we call a *production place invariant*, and place invariants I_r for each $r \in P_r$ satisfying $I_r(i) = I_r(f) = 0$, $I_r(r) = 1$ and $\forall r' \in P_r \setminus \{r\} : I_r(r') = 0$, which we call *resource place invariants*.

Proof:

Let N be a sound RCWF-net. By Corollary 4.1, the production net N_p is sound too, so N_p has an invariant I'_p satisfying $I'_p(i) = I'_p(f) = 1$ (Lemmas 4.2 and 4.1). By Lemma 2.3, there is an invariant I_p of N such that $I(p) = I'(p)$ if $p \in P_p$ and $I(p) = 0$ if $p \in P_r$.

The projection $N' = N \setminus_{P_p \cup \{r\}}$, being sound (Theorem 4.1), has an invariant J satisfying $J(r) = 1$. Since the production net of N' equals N_p , N' has an invariant I''_p corresponding to I'_p , so it satisfies $I''_p(i) = 1$ and $I''_p(r) = 0$. Let $J' = J - J(i) \cdot I''_p$, then $J'(i) = J'(f) = 0$ and $J'(r) = 1$. By Lemma 2.3, J' this corresponds to an invariant I_r with the stated properties. \square

Note that the existence of resource place invariants is a requirement for S^4PR nets [4], which links sound RCWF-nets to S^4PR nets.

5. Conclusion

We have introduced an extension of Workflow nets: *Resource-Constrained Workflow nets (RCWF-nets)* and we have given a number of necessary conditions for design correctness of these nets. One correctness criterion is structural correctness that guarantees the absence of redundant and persistent places and transitions; structural correctness can be checked by using traps and siphons. We also defined resource dependencies and discussed how to discover them in a model. Another correctness criterion is soundness, which is an extension of the soundness notion for WF-nets to the case of RCWF-nets. Additional requirements concern the durability of resources.

We prove a number of properties of sound RCWF-nets, which form a set of necessary conditions of soundness. One natural condition is that the production net of a sound RCWF-net is sound. Soundness of the production net can be checked using the procedure described in [9]. The second condition is formulated in terms of linear inequalities on incidence matrices of an RCWF-net and its production net that can be checked by standard algebraic techniques and it guarantees that no resource token creation in the RCWF-net is possible during processing. The third condition postulates that the transition invariants of the closure of a sound RCWF-net and of its underlying production net are the same, which guarantees resource conservation upon termination. And finally, in the fourth condition we showed that soundness implies the existence of a resource place invariant for all resource places, which relates sound RCWF-nets to S^4PR nets.

Related work Modelling the use of resources by Petri nets and analyzing these models is an active research field. We mention research on flexible manufacturing systems (FMS) (see [7, 4, 6, 12]), where the construction of appropriate *schedules* for such models is the key issue. Our approach emphasises the construction of robust nets with self-scheduling that are free of deadlocks irrespective of the number of resources available beyond a certain minimum.

In [3] the authors consider structural analysis of Workflow nets with shared resources. Their definition of structural soundness corresponds approximately to the existence of k, m_r such that the net is

(k, m_r) -sound. Since we consider systems where the number of cases going through the net and the number of resources can vary, and the system should work correctly for any number of cases and resources, the results of [3] are not applicable to our case.

Future work The RCWF-nets satisfying the correctness criteria given in this paper can be unsound only if they contain a deadlock or a livelock due to a lack of resources during the production process. Soundness of RCWF-nets with a fixed number of resources is decidable by using techniques from [9] but it is still an open question whether soundness is decidable for general RCWF-nets. Another research question is finding structural patterns for building sound-by-construction RCWF-nets.

References

- [1] W. 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 and K. M. van Hee. *Workflow Management: Models, Methods, and Systems*. MIT Press, 2002.
- [3] 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.
- [4] 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.
- [5] F. Commoner. *Deadlocks in Petri Nets*. Applied Data Research, Inc., Wakefield, Massachusetts, Report CA-7206-2311, 1972.
- [6] J. Ezpeleta. Flexible manufacturing systems. In C. Girault and R. Valk, editors, *Petri nets for systems engineering*. Springer-Verlag, 2003.
- [7] 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.
- [8] 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.
- [9] K. van Hee, N. Sidorova, and M. Voorhoeve. Generalised soundness of workflow nets is decidable. In J. Cortadella and W. Reisig, editors, *Application and Theory of Petri Nets 2004, ICATPN'2004*, volume 3099 of *Lecture Notes in Computer Science*, pages 197–216. Springer-Verlag, 2004.
- [10] K. van Hee, N. Sidorova, and M. Voorhoeve. Resource-constrained workflow nets. In G. Lindemann, editor, *Proc. of Concurrency Specification and Programming, CS&P'2004*, Informatik-Bericht Nr. 170, pages 166–177. Humboldt-Universität zu Berlin, 2004.
- [11] K. Lautenbach. *Liveness in Petri Nets*. Internal Report of the Gesellschaft für Mathematik und Datenverarbeitung, Bonn, Germany, ISF/75-02-1, 1975.
- [12] M. Silva and E. Turuel. Petri nets for the design and operation of manufacturing systems. *European Journal of Control*, 3(3):182–199, 1997.
- [13] P. Starke. *Analyse von Petri-Netz-Modellen*. Teubner, 1990.