“Alexandru Ioan Cuza” University of Iași

Faculty of Computer Science

Workflow Nets with Resource, Priority, and Time Constraints

Supervisor:
Prof. Dr. Ferucio Laurențiu Țiplea

Ph.D. Student:
Corina C. Dima (căs. Bocăneală)

March 2013

Workflow Nets with Resource, Priority, and Time Constraints

Corina C. Dima (căs. Bocăneală)
Department of Mathematics and Computer Science
"Dunarea de Jos" University of Galati
Galaţi, România

Ph.D. Thesis Summary

March 2013

Corina C. Dima (căs. Bocăneală)
Department of Mathematics and Computer Science
"Dunarea de Jos" University of Galati
Galaţi, România
E-mail: `Corina.Bocaneala@ugal.ro`.

Thesis Committee

Lect. Dr. Adrian Iftene, Chair ("Al. I. Cuza" University of Iaşi)
Prof. Dr. Ferucio Laurenţiu Ţiplea, Supervisor ("Al. I. Cuza" University of Iaşi)
Prof. Dr. Simona Iuliana Caramihai (University "POLITEHNICA" of Bucharest)
Prof. Dr. Adrian Atanasiu (University of Bucharest)
Prof. Dr. Marius Brezovan (University of Craiova)

Contents

List of Published Papers

My Ph.D. thesis contains personal contributions which have been announced in the following publications:

1. **C. Bocăneală**, *On Coverability Structures for Nested Petri Nets*, Proceedings of The Third International Conference on Mathematical Sciences, ICM2008, 3-6 march 2008, Al-Ain, United Arab Emirates, Volume 1, 234–243.

2. **C. Bocăneală**, *n-Levels Petri Nets*, The Annals of the University "Dunărea de Jos" of Galaţi, Fascicle II, Mathematics, Physics, Chemistry, Informatics, vol. XXXI, 2008, 105–114.

3. **C. Bocăneală**, *Termination Problem for Petri Nets with multi Levels*, The Annals of the University "Dunărea de Jos" of Galaţi, Fascicle II, Mathematics, Physics, Chemistry, Informatics, vol. XXXI, 2008, 118–125.

4. **C. Bocăneală**, *Modeling Intelligent Systems with Level Petri Nets*, The Annals of the University "Dunărea de Jos" of Galaţi, Fascicle III, Electrotechnics, Electronics, Automatic Control and Informatics, vol. 2, 2008, 31–36.

5. **C. Bocăneală**, *Place Invariants for Nested Petri Nets*, The Annals of the University "Dunărea de Jos" of Galaţi, Fascicle II, Mathematics, Physics, Theoretical Mechanics, vol. XXXII, 2009, 209–222.

6. **C. Bocăneală**, *T-Invariants for Nested Petri Nets*, The Annals of the University "Dunărea de Jos" of Galaţi, Fascicle II, Mathematics,

Physics, Theoretical Mechanics, Facicle II, Year II (XXXIV), 2010, 345–353.

7. F.L. Ţiplea, **C. Bocăneală**, *Decidability Results for Soundness Criteria of Resource-Constrained Workflow Nets*, IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans, vol. 42(1), 2012, 238–249, influence score 2011: 1.52137.

8. F.L. Ţiplea, **C. Bocăneală**, *Priority Workflow Nets*, IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 43(2), 2013, 402–415, influence score 2011: 1.52137.

9. F.L. Ţiplea, **C. Bocăneală**, *Resource Relocation in Workflow Nets with Time, Resource, and Task Priority Constraints*, IEEE Transactions on Systems, Man, and Cybernetics: Systems, to appear, influence score 2011: 1.52137.

Introduction

In the last twenty years the economy started to change its traditional industrial foundation to an informational base strategy. The necessity of the automation of the business processes led to the development of a large number of software tools called *workflow management systems (WFMSs)*. A WFMS is a generic information system which supports the modeling, the execution, and the control of the flow of work in an organization [93]. WFMSs have applications in different domains, such as: industry, banking, insurance, health-care, e-commerce, and so on. This was the reason why the researchers tried to define, model, analyze, and manage workflows [4, 131].

The are two main modeling formalisms for workflows. The first one is based on workflow graphs [104, 106, 107] and the second one is based on Petri nets [4, 131].

In this thesis we use the Petri nets theory for modeling workflows. A Petri net model of a workflow is called *workflow net (WF net)*.

Three main reasons justify the benefits of using Petri nets for workflow modeling and specification [2]:

- Petri nets offer a formal semantics and an intuitive graphical representation;
- Petri nets can clearly model states, tasks, and can distinguish between the enabling and execution of a task;
- there are many theoretically analysis techniques.

A workflow should satisfy some correctness criteria. Soundness [4] is a correctness criterion which assures the proper termination of the workflow's execution without no abnormality (a deadlock or a livelock). Generalized soundness guarantees that the system runs correctly for any number of cases.

This thesis focuses on the modeling and verification of the resource constrained workflows. [16, 17, 19] proposes *workflow nets with resources (WFR nets)*. Almost similar to these are *resource constrained workflow nets (RCWF nets)* [60, 61, 120]. The main difference between these classes is that in WFR nets each place has associated an invariant which ensures the resource preservation along the execution of the system.

The main problem we studied is soundness for resource constrained workflow nets [120]. In most practical applications activities should be executed in some order. This problem can be solved by associating priorities to tasks to ensure their execution in the correct succession. We defined *priority (resource-constrained) workflow nets* [121]. The complex workflows usually combine resource, priority, and time constraints [122]. Sometimes is necessary to "compose" workflows. This is the reason why we defined *multi-level workflow nets with resource constraints (mlRCWF nets)*. We enrich the mlRCWF net model by adding priorities between tasks and time durations associated to tasks. We also proposed a *resource relocation* policy between tasks.

Structure of the Thesis

This thesis is organized as follows:

The first chapter presents the basic definitions regarding Petri nets theory. We remember the main decision problems for Petri nets and their decidability status. We also present some classes of Petri nets which are relevant for the rest of the paper. The Petri nets path logic is necessary to prove the decidability of the CBhC and SBhC conditions in Chapter 4. Deterministic counter machines are described because the undecidability of the halting problem is essential to prove the undecidability of soundness for priority workflow nets.

In Chapter 2 we review the main results about workflow nets. We focus on the soundness problem. This problem was proved to be decidable in [59, 116]. We also present some workflow net classes for which soundness can be decided efficiently.

Chapter 3 analyzes two Petri net models for resource constrained workflows. The first model is WFR nets [16, 17, 19]. Another similar model is *RCWF nets* [60, 61]. The main matter we approach is the decidability of soundness for RCWF nets. We refine the soundness criteria for RCWF nets and we group them into three classes: soundness criteria under specified resources, soundness criteria under unspecified resources, and structural soundness. We prove that soundness criteria under specified resources are decidable. In the case of the soundness criteria under unspecified resources we show that k-soundness (i.e. k-soundness with respect to a minimal resource marking) is decidable. Structural R-soundness is also decidable. Most results in this chapter have been published in [120].

In Chapter 4 we define *priority (resource-constrained) workflow nets* and we formulate the soundness criteria for them. We illustrate that priority Petri nets can simulate deterministic counter machines (DCMs). Because the halting problem for DCMs is undecidable, it follows that the soundness is undecidable for priority (resource-constrained) workflow nets. We prove that soundness is decidable for the priority workflow nets which satisfy the EQUAL-conflict or the conflict-freeness conditions. We also generalize these two conditions to the *CBhC*, respectively the *SBhC conditions* and we show that soundness is decidable for the priority (resource-constrained) workflow nets which satisfy these new conditions. Using Petri net path logic we demonstrate the decidability for the CBhC and SBhC conditions. These results appear in [121].

In Chapter 5 we "compose" workflow nets, we introduce *multi level workflow nets with resource constraints* (mlRCWF nets), and we define soundness for mlRCWF nets. mlRCWF nets are enriched with priority and time durations associated to tasks. The timed priority mlRCWF net model is investigated with respect to the soundness property. The soundness is reduced to the soundness of the untimed priority mlRCWF net model. The complexity of this reduction is linear in the size of the original model. We also analyze the case of the *resource relocation*. The soundness in this case can also be reduced to the soundness of the priority untimed model. The complexity of this procedure is quadratic in the size of the original model. The results in this chapter are synthesized in [122].

The thesis ends with a section of conclusions and future work.

Contributions of the Thesis

This thesis presents an overview about workflow nets with resource constraints. There are many attempts to incorporate the resource perspective in modeling workflows. We focus on *resource-constrained workflow nets* (RCWF nets) and their soundness properties. The decidability status of the k-soundness and generalized soundness problems for general resource-constrained workflow nets were open problems. We prove that k-soundness for RCWF nest is decidable [120]. To do that, we gradually refine the soundness criteria for RCWF nets considering the number of cases that the system is able to correctly process and the number of the available resources [120].

The soundness criteria were grouped into three classes:

1. For the case of *soundness criteria under specified resources* some resource marking is given and the problem is to decide soundness of an RCWF with respect to that marking. We showed that these soundness criteria are decidable. The proof is based on closure nets and instantiation nets.

2. The second class is the one of *soundness criteria under unspecified resources*. In this case, the main question is to decide whether there is a resource marking that makes the RCWF net sound. k-soundness is equivalent to k-soundness with respect to some minimal marking on the resource places. We establish that it is decidable if this minimal marking exists and, if it exists, it can be effectively computed. Therefore k-soundness is decidable for general RCWF nets. This characterization can not be extended to generalized soundness because there are RCFW nets for which such a minimal marking for the resource places does not exist. We also prove that the ($\leq k$)-soundness property is decidable, which may be sufficient for modeling many real workflows.

3. The third class of soundness criteria includes two cases of *structural soundness* of RCWF nets. Structural R-soundness is proved to be decidable, while the decision for the structural soundness “seems” as hard as the decision of soundness.

Adding priorities to tasks is a natural requirement for the real systems. As far as we know, no workflow model with priorities has been proposed until now. Our aim is to fill this gap. Starting from priority Petri nets we propose *priority (resource-constrained) workflow nets* (P(RC)WF nets) [121]. Priorities can be associated to the resources or to tasks, and we show how priorities associated to resources can be simulated by priorities associated to tasks. The main interest is also the investigation of the decidability status of the soundness property for P(RC)WF nets. It is well known that priority Petri nets can simulate deterministic counter machines. We extend this result to priority (resource-constrained) workflow nets. Because the halting problem is undecidable for deterministic counter machines, we obtain that soundness is undecidable for P(RC)WF nets [121].

If we impose some additional conditions, the soundness of a priority (resource-constrained) workflow net can be reduced to the soundness of its underlying (resource-constrained) workflow net. Such conditions are the well known EQUAL-conflict condition and the conflict-freeness condition. We propose two more conditions: the *CBhC condition* which generalizes the EQUAL-conflict condition and *SBhC condition* which generalizes the conflict-freeness condition. Using Yen's path logic for Petri nets we prove that the CBhC and SBhC conditions are decidable. A generalization of the CBhC and SBhC conditions is proposed in order to be able to model more types of workflow systems. The above results are verified for this case too [121].

Sometimes it is necessary that several workflows to use a set of shared resources. To manage such situations, we define *multi-level workflow nets with resource constraints* (mlRCWF nets) as a place compositions of standard workflows with resource constraints [122]. The soundness of this model is investigated, and we conclude that it cannot be reduced, in general, to the soundness property of standard workflows. $(\bar{k}, R)$-soundness is decidable for mlRCWF nets [122].

The complex workflows in real world usually combine resource, task priority, and time constraints. This led us to the idea to combine all these constraints into a suitable workflow model, and to check the soundness property for it. We enrich the mlRCWF model with task priorities and time. Time is added as time durations associated to tasks. We proved that soundness for the *timed priority mlRCWF nets* model can be reduced to soundness of the untimed priority mlRCWF nets model, and therefore it is undecidable. The reduction complexity is linear in the size of the original model [122].

Sometimes is necessary that a resource used by a task to be released in order to be used by another task with a higher priority. We call that *resource relocation*. We consider *timed priority mlRCWF nets with resource relocation* and we show that the soundness property of these nets can be reduces to the untimed priority model if the transitions can be indefinitely delayed. The reduction complexity is quadratic in the size of the original model [122].

Finally we present some conclusions and some future work ideas.

I would not have finish my work without the help of some special persons. First of all, I would like to thank very much my supervisor, Professor Dr. Ferucio Laurenţiu Ţiplea, for accepted me as a PhD student, for his constant guidance, help and encouragement that supported me during these years. The results presented here were obtained with him. I also want to thank my former supervisor, Prof. Dr. Toader Jucan, for guiding my first steps in the doctoral studies; all the committee members for carefully reading my thesis; my colleagues for theirs encouragement; and my family for its continuous support.

Acronyms

CBhC condition	Concurrent behavioral conflict condition
CE property	Concurrent enabling property
CF condition	Conflict-freeness condition
DCM	Deterministic counter machine
EqC condition	EQUAL-conflict condition
mlRCWF net	Multi-level resource-constrained workflow net
PPN	Priority Petri net
PRCWF net	Priority resource-constrained workflow net
PWF net	Priority workflow net
RCWF net	Resource-constrained workflow net
SBhC condition	Sequential behavioral conflict condition
SE property	Sequential enabling property
TCE property	Transfer of concurrent enabling property
TSE property	Transfer of sequential enabling property
WF net	Workflow net
WFMS	Workflow management system
WFR net	Workflow net with resources

Chapter 1

Petri Nets

In this section we review the basic terminology, concepts, notations, and results related to Petri nets. For more details, the reader is referred to [92, 99, 89, 100, 69, 56].

1.1 Basic Notations

In what follows we use the following notations:

The set of integers is denoted by $\mathbb{Z}$. $\mathbb{N}$ represents the set of natural numbers (non-negative integers).

A binary relation on a set T is any subset ρ of $T \times T$. ρ^{-1} stands for the inverse of ρ, $\bar{\rho}$ stands for the complement of ρ, and ι_T stands for $\{(t,t)|t \in T\}$. The *composition* of two binary relations ρ_1 and ρ_2 on T is denoted $\rho_1 \circ \rho_2$. A binary relation ρ on T is called *reflexive* if $\iota_T \subseteq \rho$, *irreflexive* if $\rho \cap \iota_T = \emptyset$, *symmetric* if $\rho^{-1} = \rho$, *asymmetric* if $\rho \cap \rho^{-1} = \emptyset$, *transitive* if $\rho \circ \rho \subseteq \rho$, and an *equivalence relation* if it is reflexive, symmetric, and transitive.

$\subseteq$ represents the set inclusion and $\subset$ represents the strict inclusion. $|A|$ stands for the cardinality of the set A.

The free monoid generated by an alphabet A under the concatenation operation is denoted by A^*. The elements of A^* are called *words* over A. λ is the unity of A^* (the *empty word*). A^+ stands for $A^* - \{\lambda\}$. $|w|$ denotes the length of the word w. A language over A is any subset of A^*.

A *permutation* of a word w is any word w' obtained by permuting w's letters. For instance, *baab* is a permutation of *abab*.

If $f : A \to B$ is a function and $C \subseteq A$, then $f|_C$ denotes the restriction of f to C (i.e., $f|_C : C \to B$ and $f|_C(a) = f(a)$, for any $a \in C$).

1.2 Basic on Petri Nets

Petri nets are a well-known formalism for modeling and analyzing concurrent systems. They are an useful tool for solving problems from many fields such as: industry, software engineering, business processes, social, and educational systems. This theory was introduced by C.A. Petri in 1962. The characteristics that recommend it are: the simplicity, the intuitive graphical notation, the formal semantics, and the expressiveness.

Definition 1.1 *[56] A (finite)* Petri net *is a 4-tuple* $\Sigma = (S, T, F, W)$*, where:*

- S *and* T *are two finite sets (of* places *and* transitions*, respectively),* $S \cap T = \emptyset$*,*
- $F \subseteq (S \times T) \cup (T \times S)$ *is the* flow relation*, and*
- $W : (S \times T) \cup (T \times S) \to \mathbb{N}$ *is the* weight function *of* Σ *verifying* $W(x, y) = 0$ *if and only if* $(x, y) \notin F$*.*

Given $x \in S \cup T$, we denote: ${}^\bullet x = \{y|(y, x) \in F\}$ the *pre-set* of x and $x^\bullet = \{y|(x, y) \in F\}$ the *post-set* of x.

A *marking* of Σ is any function $M \in \mathbb{N}^S$ from S into $\mathbb{N}$, usually denoted as an S-indexed vector. Given two markings M and M', we have $M \leq M'$ if and only if $M(s) \leq M'(s)$, for any $s \in S$. Moreover, if $M(s) < M'(s)$ for some place s, then we have $M < M'$. For a marking M of Σ and $S' \subseteq S$, $M|_{S'}$ is a submarking for M.

A *marked Petri net* is a pair $\gamma = (\Sigma, M_0)$, where Σ is a Petri net and M_0 is the initial marking of Σ.

We represent graphically a Petri net as follows: places are symbolized by circles, transitions are symbolized by boxes, the flow relation is symbolized by an arc between x and y, for $(x, y) \in F$, and the weight function labels the arcs whenever $W(x, y) \geq 1$. A marking M of a Petri net is represented by drawing $M(s)$ black tokens into the circle representing the place s, for all $s \in S$.

A transition t is *enabled* at a marking M, denoted $M[t\rangle_\Sigma$, if $M(s) \geq W(s,t)$, for all $s \in S$. Denote by $T(M)$ the set of all the transitions enabled to the marking M. If t is enabled at M, then it can *fire* yielding a new marking M' given by $M'(s) = M(s) - W(s,t) + W(t,s)$, for all $s \in S$; we denote this by $M[t\rangle_\Sigma M'$. We denote by $[M\rangle_\Sigma$ the set of all reachable markings (from M) in Σ. When no confusion may arise we simplify the notation $[\cdot\rangle_\Sigma$ to $[\cdot\rangle$.

The incidence matrix of Σ is a matrix indexed by $S \times T$, defined by $C(s,t) = W(t,s) - W(s,t)$, for all $(s,t) \in S \times T$.

An integer vector I, $I \neq 0$ indexed by S is a S-invariant if $I^t \cdot C = 0$.

$\|I\| = \{s \in S | I(s) \neq 0\}$ is the support of I. We denote by $\|I\|^+ = \{s \in S | I(s) > 0\}$ and by $\|I\|^- = \{s \in S | I(s) < 0\}$.

Definition 1.2 *Let M_0 be a marking of a Petri net Σ. We say that:*

- Σ *is* bounded with respect to M_0 *if $[M_0\rangle$ is finite (i.e. there exists an integer $n \geq 1$ such that $M(s) \leq n$ for all reachable markings M and all places s);*

- Σ *is* M'-bounded on S' *with respect to M_0, where S' is a subset of places and M' is a marking on S', if $M|_{S'} \leq M'$, for all $M \in [M_0\rangle$;*

- *a transition t of Σ is* quasi-live with respect to M_0 *if there exists $M \in [M_0\rangle$ such that $M[t\rangle$. If t is not quasi-live we will say that t is* dead with respect to M_0*;*

- *a transition t of Σ is* live with respect to M_0 *if for any $M \in [M_0\rangle$ there exists $M' \in [M\rangle$ such that $M'[t\rangle$. Σ is* live with respect to M_0 *if all its transitions are live with respect to M_0;*

- *a marking M is a* home marking of Σ with respect to M_0 *if $M \in [M'\rangle$, for all $M' \in [M_0\rangle$.*

Chapter 2

Workflow Nets Theory

In this chapter we review some basic concepts on classical workflow nets. Our main interest is the decidability of the soundness property.

To function properly, a workflow net must satisfy some behavioral correctness criteria. One of the most important correctness criteria is soundness [4]. This criterion ensures the proper termination of a workflow execution without no anomaly, such as deadlock or livelock. Generalized soundness guarantees that the system runs correctly for any number of cases, and it was proved decidable in [59, 116].

2.1 Workflow Petri Nets

Definition 2.1 *[2] A* workflow net *(WF net) is a Petri net Σ with the following two properties:*

1. *Σ has two special places i and o called the* input *and, respectively, the* output place *of Σ. They satisfy ${}^\bullet i = \emptyset$, and $o^\bullet = \emptyset$;*

2. *Any node $x \in S \cup T$ in the graph of Σ is on a path from i to o.*

Given a WF net Σ, a place s of it, and an integer $k \geq 1$, we denote by M_{ks} the marking given by $M_{ks}(s) = k$ and $M_{ks}(s') = 0$, for all $s' \neq s$. When $k = 1$ the notation is simplified to M_s.

2.2 The Soundness Property

A WF net should satisfy some "behavioral correctness criteria".

In this paper we will use the soundness criteria in a simplified form as it was defined in [60]. Thus, we say that a workflow net Σ is *k-sound*, where $k \geq 1$, if $M_{ko} \in [M\rangle$, for all $M \in [M_{ki}\rangle$. Σ is called *sound* if it is k-sound, for all $k \geq 1$.

Definition 2.2 *Let Σ be a WF-net and $k \geq 1$ an integer.*

1. *Σ is called* k-sound *if for any $M \in [M_{ki}\rangle$ we have $M_{ko} \in [M\rangle$.*

2. *Σ is called* sound *if it is k-sound, for all $k \geq 1$.*

3. *Σ is called* structurally sound *if it is k-sound for some $k \geq 1$.*

Definition 2.3 *A Petri net is called the* k-closure *of a WF-net Σ, $k \geq 1$ if it is obtained from Σ by adding a new transition t^* and two arcs (o, t^*), and (t^*, i), with the weight $W(t^*, i) = W(o, t^*) = k$.*

The k-closure is of a WF-net is unique up to the renaming of t^*. It will be denoted by $\Sigma(k)$. $\Sigma(1)$ will be called *the closure* of Σ.

Proposition 2.1 *[1] Let Σ be a WF net and k an integer. Then the following properties hold:*

1. *Σ is k-sound if and only if its k-closure $\Sigma(k)$ is bounded w.r.t. M_{ki}, and t^* is live w.r.t. M_{ki}.*

2. *Σ is k-sound if and only if M_{ko} is a home marking of Σ w.r.t M_{ki}.*

Proposition 2.2 *The k-soundness problem is decidable.*

Proposition 2.3 *The generalized soundness problem is decidable.*

Chapter 3

Resource Constrained Workflow Nets

WF-nets were initially meant to coordinate the execution of activities abstracting of time or resource constraints. In this chapter we emphasize the influence of resources in workflow management systems. Resources are modeled by places. A resource places is a plain place having a specified type and connected with the transitions according with the system necessities. The number of tokens in a resource place gives the number of the available resources of a certain type. Resources are durable. They can not be created or destroyed; they are used in the process execution, and then released.

The main problem we focus is soundness, a correctness criterion which assures the proper termination of the workflow's execution, and no anomaly has occurred. This criterion was formulated for WFR nets [16, 17], and for RCWF nets [60, 61, 62].

The generalized soundness was proved to be decidable in [108]. The technique used is based on the home space property.

3.1 Workflow Nets with Resources

In [16] are introduced workflow nets with resources (WFR nets). All the results in this section are based on [16, 17, 19]. WFR nets are similar to RCWF nets; the difference consists in the existence of a place invariant for each resource, assuring the resource preservation. Moreover, the underlying WF net is required to be bounded.

WFR nets are defined in the following manner [16]:

Definition 3.1 *A* WFR-net Σ^r *is a 4-tuple*

$$\Sigma^r = (S \cup S^r, T, F \cup F^r, W \cup W^r),$$

where:

1. $\Sigma = (S, T, F, W)$ *is a bounded WF-net,*
2. $S \cap S^r = \emptyset$*,*
3. $F^r \subseteq S^r \times T \cup T \times S^r$*,*
4. $W^r : S^r \times T \cup T \times S^r \to \mathbb{N}$ *verifies* $W^r(x, y) = 0$ *iff* $(x, y) \notin F^r$*,*
5. $\forall u \in F^r$*,* $W^r(u) \geq 1$*,*
6. $\forall r \in S^r$*,* $\exists I^r \geq 0$*, such that* ${}^tI^r \cdot C = 0$ *and* $\|I^r\| \cap S^r = \{r\}$*.*

In Definition 3.1, S^r denotes the set of resource places. The fifth condition represents the resource use, and the last one assures the resource preservation.

The soundness properties can be formulated for WFR-nets, too [17, 19]. The soundness definition for WFR nets imposes also the quasi-liveness condition for the net.

We denote by $(\Sigma^r)^*$ the net obtained form Σ^r substituting Σ with its k-closure $\Sigma(k)$.

Proposition 3.1 *Let* Σ^r *be a WFR-net. If* Σ^r *is* (k, R)*-sound then* Σ *is* k*-sound.*

Theorem 3.1 *Let* Σ^r *be a WFR-net.* Σ^r *is* (k, R)*-sound if and only if* $(\Sigma^r)^*$ *is live and bounded.*

In [17, 19] are presented some WFR-net classes for which necessary and sufficient condition can be obtained and decided effectively.

3.2 Resource Constrained Workflow Nets

To introduce *resource-constrained workflow nets* (RCWF nets) [60] we consider place-extensions of workflow nets.

Definition 3.2 *A* place-extension *of a Petri net* Σ *is a Petri net* Σ' *satisfying* $S \subseteq S'$, $T' = T$, $F'|_{S\times T\cup T\times S} = F$, *and* $W'|_{S\times T\cup T\times S} = W$.

A place-extension Σ' *of* Σ *is called* empty *if* $S' = S$. *Otherwise, it is called* non-empty.

Definition 3.3 *Let* Σ *be a WF net. A* resource-constrained workflow net *associated to* Σ, *(*RCWF net*), is a non-empty place-extension* Σ^r *of* Σ. *The WF net* Σ *is called the* underlying WF net *or the* production net *of* Σ^r.

Places in S^r are called *resource places*.

For better readability, an RCWF net Σ^r will be written in the form

$$\Sigma^r = (S \cup S^r, T, F \cup F^r, W \cup W^r),$$

where $\Sigma = (S, T, F, W)$ is underlying WF net, S^r is the set of resource places, $S \cap S^r = \emptyset$, $F^r \subseteq S^r \times T \cup T \times S^r$, and $W^r : S^r \times T \cup T \times S^r \to \mathbb{N}$ verifies $W^r(x, y) = 0$ iff $(x, y) \notin F^r$.

A marking of an RCWF net Σ^r is a pair (M, R), where M is a marking over S and R is a marking over S^r (i.e., M is a function from S into $\mathbb{N}$, and R is a function from S^r into $\mathbb{N}$). R will be called a *resource marking*.

Example 3.1 *In Figure 3.1 is represented an RCWF net with just one resource place,* r. *The production net* Σ *is the net obtained by removing the resource place* r *and all its adjacent arcs. The net models a production line which manufactures some products following two stages represented by the transitions* t_1 *and* t_2. *After the first stage is completed, randomly some products are checked by a process modeled by* t_3 *which needs two resources (measuring and control instruments) from the resource place* r. t_4 *ends the verification process requiring one more resource, and finally releasing all three resources, and the product to be processed by* t_2.

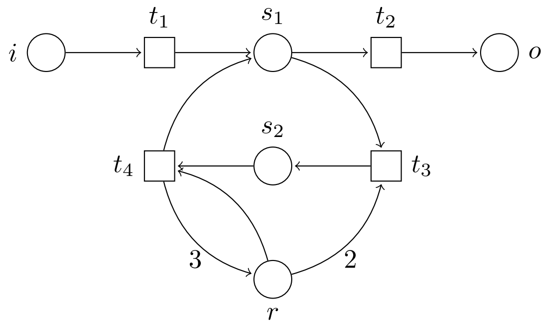

Figure 3.1: An RCWF net

3.3 Soundness of RCWF nets

The soundness property depends on the number of cases to be processed and on the number of available resources. This led us to consider case- and/or resources- dependent soundness criteria.

Definition 3.4 *Let Σ^r be an RCWF net, $k \geq 1$ an integer, and R a marking on S^r.*

1. *Σ^r is called* (k, R)-sound *if, for any $(M, R') \in [M_{ki}, R\rangle_{\Sigma^r}$, the following properties hold:*
   (a) *$R' \leq R$;*
   (b) *$(M_{ko}, R) \in [M, R'\rangle_{\Sigma^r}$.*
2. *Σ^r is called* $(\geq k, R)$-sound *if Σ^r is (m, R)-sound, for all $m \geq k$.*
3. *Σ^r is called* $(k, \geq R)$-sound *if Σ^r is (k, R')-sound, for all $R' \geq R$.*
4. *Σ^r is called* $(\geq k, \geq R)$-sound *if Σ^r is (m, R')-sound, for all $m \geq k$ and $R' \geq R$.*
5. *Σ^r is called* k-sound *if there exists R such that Σ^r is $(k, \geq R)$-sound.*
6. *Σ^r is called* $(\geq k)$-sound *if there exists R such that Σ^r is $(\geq k, \geq R)$-sound.*

7. Σ^r *is called* ($\leq k$)-sound *if there exists* R *such that* Σ^r *is* $(m, \geq R)$-*sound, for any* $1 \leq m \leq k$.

8. Σ^r *is called* sound *if there exists* R *such that* Σ^r *is* $(\geq 1, \geq R)$*-sound.*

We say that the soundness criteria from Definition 3.4 (1-4) are soundness criteria under specified resources because the property is checked w.r.t. a given resource R, for all $R' \geq R$. The criteria from Definition 3.4 (5-8) are called soundness criteria under unspecified resources because the resource marking R should be found. A sound RCWF net is capable to process correctly arbitrary many cases with unbounded resources.

Example 3.2 *The RCWF net* Σ^r *in Example 3.1 is* $(k, \geq(2k+1))$*-sound for any* $k \geq 1$*, where* $(2k+1)$ *is the marking for the resource* r. Σ^r *is not* $(\geq k, \geq(2k+1))$*-sound. Consider* $(M_{(k+1)i}, (2k+2))\ [t_1^{k+1} t_3^{k+1}\rangle (M', R')$ *with* $M'(s_1) = R'(r) = 0$ *and* $M'(s_2) = k+1$*. Because no transition is enabled at this marking, we have that the marking* $(M_{(k+1)o}, (2k+2))$ *is not reachable from* (M', R') *and* Σ^r *is not* $((k+1), (2k+2))$*-sound.*

Remark 3.1 1. *From the above remark results that if an RCWF net* Σ^r *is* k*-sound or sound, then* Σ *is* k*-sound or sound, respectively.*

2. *It is important to mention that only* (k, R)*-soundness of an RCWF net* Σ^r *does not necessarily imply the* k*-soundness of its underlying WF net* Σ.

Proposition 3.2 *An RCWF net* Σ^r *is* $(\geq k, R)$*-sound, where* $k \geq 1$*, if and only if* Σ^r *is* $(\geq 1, R)$*-sound.*

Corollary 3.1 *Let* Σ^r *an RCWF net. The following properties hold true:*

1. Σ^r *is* $(\geq k, \geq R)$*-sound if and only if* Σ^r *is* $(\geq 1, \geq R)$*-sound.*

2. Σ^r *is* $(\geq k)$*-sound if and only if* Σ^r *is sound.*

Definition 3.5 *Let* Σ^r *be an RCWF net.*

1. Σ^r *is called* structurally R-sound, *where* R *is a marking on* S^r*, if there exists* $k \geq 1$ *such that* Σ^r *is* (k, R)*-sound.*

2. Σ^r *is called* structurally sound *if there exist* $k \geq 1$ *and a marking* R *on* S^r *such that* Σ^r *is* (k, R)*-sound.*

Let as consider an RCWF net Σ^r, an integer $k \geq 1$, and R a marking on S^r. The (k, R)-*closure* of Σ^r is the Petri net $(\Sigma^r)^*$ obtained from Σ^r by adding a new transition t^*, two new arcs (o, t^*) and (t^*, i) with weight k, and, for each resource $r \in R$, two new arcs (r, t^*) and (t^*, r) with the weight $R(r)$.

3.4 Deciding Soundness of RCWF Nets

The soundness concepts in Definition 3.4 leads to some corresponding decision problems. For instance, the decision problem associated to the (k, R)-soundness is to decide for a given RCWF net Σ^r if it is (k, R)-sound for $k \geq 1$ and a resource marking R.

3.4.1 Deciding Soundness of RCWF Nets Under Specified Resources

Proposition 3.3 *Let* Σ^r *be an RCWF net,* $k \geq 1$*, and* R *a marking on* S^r*. Then, the following properties hold:*

1. Σ^r *is* (k, R)*-sound if and only if* Σ^r *is* R*-bounded on* S^r *w.r.t.* (M_{ki}, R)*,* $(\Sigma^r)^*$ *is bounded w.r.t.* (M_{ki}, R)*, and* t^* *is live w.r.t.* (M_{ki}, R)*.*

2. Σ^r *is* (k, R)*-sound if and only if* Σ^r *is* R*-bounded on* S^r *w.r.t.* (M_{ki}, R)*, and* (M_{ko}, R) *is a home marking of* Σ^r *w.r.t.* (M_{ki}, R)*.*

Corollary 3.2 *The* (k, R)*-soundness problem for RCWF nets is decidable.*

Proposition 3.4 *The* $(\geq k, R)$*-soundness problem for RCWF nets is decidable.*

Proposition 3.5 *The* $(k, \geq R)$*-soundness problem for RCWF nets is decidable.*

3.4.2 Deciding Soundness of RCWF Nets Under Unspecified Resources

It is more difficult to decide soundness under unspecified resources because we must find a lower bound for the resource marking such that the RCWF net satisfies the soundness criteria w.r.t. it. We will show that k-soundness is decidable, but the result can not be extended to soundness.

A simple transition sequence of Σ from M_{ki} starts with M_{ki} and reaches M_{ko} without repeating any marking or, if it reaches a marking already encountered, then it stops there. $R_{\Sigma^r,k}$ represents the minimal amount of resources such that all simple transition sequences of Σ can fire in Σ^r from $(M_{ki}, R_{\Sigma^r,k})$.

Theorem 3.2 *Let Σ^r be an RCWN and $k \geq 1$ an integer. Σ^r is k-sound if and only if $R_{\Sigma^r,k}$ exists, and Σ^r is $(k, \geq R_{\Sigma^r,k})$-sound.*

Corollary 3.3 *k-soundness problem for RCWF nets is decidable.*

Corollary 3.4 *Let Σ^r be an RCWF net and $k \geq 1$ an integer. Then, Σ^r is $(\leq k)$-sound if and only if $R_{\Sigma^r,k}$ exists, and Σ^r is $(\leq k, \geq R_{\Sigma^r,k})$-sound.*

Corollary 3.5 *$(\leq k)$-soundness problem for RCWF nets is decidable.*

Given an RCWF net Σ^r and an integer $k \geq 1$, denote by $R_{\Sigma^r,\geq k}$ the minimal marking on S^r, if it exists, with the property $R_{\Sigma^r,\geq k} \geq w^-|_{S^r}$, for any $w \in L_s(\Sigma, m)$ and $m \geq k$.

Corollary 3.6 *Let Σ^r be an RCWF net and $k \geq 1$ an integer. If $R_{\Sigma^r,\geq k}$ exists, then Σ^r is $(\geq k, \geq R_{\Sigma^r,\geq k})$-sound if and only if Σ^r is $(\geq k)$-sound.*

Conjecture 3.1 *It is decidable, given an RCWF net Σ^r and an integer $k \geq 1$, whether $R_{\Sigma^r,\geq k}$ exists.*

3.4.3 Deciding Structural Soundness of RCWF Nets

For an RCWF net Σ^r and a marking R on S^r, we denote by $k_{\Sigma^r,R}$ the least $k \geq 1$, if it exists, satisfying $(M_{ko}, R) \in [M_{ki}, R\rangle_{\Sigma^r}$.

Proposition 3.6 *An RCWF net Σ^r is structurally R-sound, where R is a marking on S^r, if and only if $k_{\Sigma^r,R}$ exists and Σ^r is $(k_{\Sigma^r,R}, R)$-sound.*

Corollary 3.7 *The structural R-soundness problem for RCWF nets is decidable.*

Theorem 3.3 *[108] The soundness problem for RCWF nets is decidable.*

Chapter 4

Priority Workflow Nets

In many practical situations it is necessary to associate priorities to some task to ensure the execution of the activities in the correct order.

The results presented in this chapter are published in [121]. We consider *priority (resource-constrained) workflow nets.* There are many situations when a resource is preferable to be used before other. This leads us to the idea of associating priorities to the resources. The priorities on the resources can be simulated by priorities on some tasks using those resources. We show that the soundness property is undecidable for priority (resource-constrained) workflow nets, but we identify some workflow net classes with soundness decidable. In that cases soundness can be reduced to the soundness of the underlying net. These classes satisfies the EQUAL-conflict condition, the conflict-freeness condition, the CBhC condition, or the SBhC-condition.

4.1 Basic Definitions

Definition 4.1 *A* priority relation *over a non-empty finite set T is any binary relation $\rho \subseteq T \times T$ with the following two properties:*

1. *ρ is irreflexive, asymmetric, and transitive;*
2. *$\bar{\rho}$ is an equivalence relation.*

$t \, \rho \, t'$ denotes the fact that "t' has priority over t".

T_0, the set of lowest priority elements of T, will be denoted by Ω.

Definition 4.2 *A* priority Petri net *(PP net) is a pair* (Σ, ρ)*, where* Σ *is a Petri net and* ρ *is a priority relation over the set of transitions of* Σ*.*

We say that t is *ρ-enabled* at M, and we denote by $M[t\rangle_\rho$, if t is enabled at M, and there is no other transition enabled at M with a higher priority than t. If $M[t\rangle_\rho$, then t may fire at M yielding a new marking M' given by $M[t\rangle M'$.

Boundedness, liveness, and home markings can be defined for priority Petri nets just by replacing $[\cdot\rangle$ by $[\cdot\rangle_\rho$.

Because workflow nets and resource-constrained workflow nets are special cases of Petri nets, one can talk about *priority workflow nets* (PWF nets) and *priority resource-constrained workflow nets* (PRCWF nets). We emphasize that priorities in a PRCWF net are imposed on transitions and not on resource places.

4.2 Soundness Criteria for PWF Nets

We extend the soundness criteria from WF (RCWF) nets to PWF (PRCWF) just by replacing $[\cdot\rangle$ by $[\cdot\rangle_\rho$.

Priorities restrict the behavior of priority Petri nets controlling the enabled transition to be applied. This is the reason why standard characterizations of soundness for workflow nets do not work for priority workflow nets.

Proposition 4.1 *1. A PWF net* (Σ, ρ) *is k-sound, where* $k \geq 1$*, if and only if* M_{ko} *is a home marking of* (Σ, ρ) *w.r.t.* M_{ki}*.*

2. A PRCWF net (Σ^r, ρ) *is* (k, R)*-sound, where* $k \geq 1$ *and R is a marking on* S^r*, if and only if* (Σ^r, ρ) *is R-bounded on* S^r *w.r.t.* (M_{ki}, R)*, and* (M_{ko}, R) *is a home marking of* (Σ^r, ρ) *w.r.t.* (M_{ki}, R)*.*

4.3 Soundness Undecidability for PWF and PRCWF Nets

Deterministic counter machines [57] can be simulated by priority Petri nets. We can extend this result for priority (resource-constrained) workflow nets. The undecidability of the halting problem for such counter machines [88] leads to the undecidability of soundness for priority (resource-constrained) workflow nets.

Theorem 4.1 *The 1-soundness problem for PWF nets is undecidable.*

Theorem 4.2 *The (k, R)-soundness problem for PRCWF nets is undecidable.*

4.4 Classes of Priority WF-Nets with Decidable Soundness

Soundness for PWF nets (or PRCWF nets) is decidable only for classes where the home marking problem (or the home marking problem and boundedness on resources) are decidable. Bause analyzes priority Petri nets which satisfy the *EQUAL-conflict* (EqC) *condition* [21], and Yen studied *conflict-free Petri nets* [130]. We formulate two new conditions: the CBhC and SBhC conditions.

Definition 4.3 *Let (Σ, ρ) be a priority Petri net and M_0 a marking of Σ. (Σ, ρ) satisfies the* EQUAL-conflict *(EqC)* condition *w.r.t. M_0 if for any two transitions t and t' the following property denoted $EqC(t, t')$ holds:*

$$^\bullet t \cap ^\bullet t' \neq \emptyset \quad \Rightarrow \quad \begin{array}{l} 1.\ t\,\bar{\rho}\,t', \textit{and} \\ 2.\ W(s,t) = W(s,t') \ \vee \ t, t' \in \Omega. \end{array}$$

Definition 4.4 *Let Σ be a Petri net. Σ is a conflict-free Petri net if each place s satisfies one of the two conditions:*

- $|s^\bullet| \leq 1$, *or*
- *if $|s^\bullet| > 1$, then t and s are on a self-loop, for any $t \in s^\bullet$.*

Definition 4.5 *Let Σ be a Petri net, M_0 a marking of it, and t and t' transitions of Σ (not necessarily distinct).*

1. *t and t' are* concurrently enabled at a marking M of Σ, *if $M(s) \geq W(s,t) + W(s,t')$, for any place s. We denote that $M[\{t,t'\}\rangle$.*

2. *t and t' satisfy the* concurrent enabling property w.r.t. M_0, *denoted $CE(t,t')$, if the following property holds:*

$$(\forall M \in [M_0\rangle)(M[t\rangle \ \wedge \ M[t'\rangle \ \Rightarrow \ M[\{t,t'\}\rangle).$$

3. *t and t' satisfy the* transfer of concurrent enabling property w.r.t. M_0, *denoted $TCE(t,t')$, if the following two properties hold:*

   (a) $(\forall M \in [M_0\rangle)(M[t\rangle \ \Rightarrow \ M[t'\rangle)$;

   (b) $(\forall t'' \in T)(CE(t'',t) \ \Rightarrow \ CE(t',t''))$.

Definition 4.6 *Let (Σ,ρ) be a priority Petri net and M_0 a marking of Σ. (Σ,ρ) satisfies the* concurrent behavioral conflict (CBhC) condition w.r.t. M_0 *if for any two transitions t and t' the following property, denoted $CBhC(t,t')$, holds:*

$$\neg CE(t,t') \quad \Rightarrow \quad \begin{array}{ll} 1. & t\,\bar{\rho}\,t', \mathit{and} \\ 2. & TCE(t',t) \ \vee \ t,t' \in \Omega. \end{array}$$

Proposition 4.2 *Let (Σ,ρ) be a priority Petri net, and M_0 a marking of Σ. If (Σ,ρ) satisfies the EqC condition, then (Σ,ρ) satisfies the CBhC condition w.r.t. M_0.*

Definition 4.7 *Let Σ be a Petri net, M_0 a marking of it, and t and t' two transitions of Σ (not necessarily distinct).*

1. *t and t' are* sequentially enabled at a marking M of Σ *if $M[tt'\rangle$.*

2. *t and t' satisfy the* sequential enabling property w.r.t. M_0, *denoted $SE(t,t')$, if the following property holds:*

$$(\forall M \in [M_0\rangle)(M[t\rangle \ \wedge \ M[t'\rangle \ \Rightarrow \ M[tt'\rangle).$$

3. *t and t' satisfy the* transfer of sequential enabling property w.r.t. M_0, *denoted $TSE(t,t')$, if the following two properties hold:*

(a) $(\forall M \in [M_0\rangle)(M[t\rangle \Rightarrow M[t'\rangle)$*;*
(b) $(\forall t'')(SE(t'', t) \Rightarrow SE(t', t''))$.

Definition 4.8 *Let* (Σ, ρ) *be a priority Petri net and* M_0 *a marking of it.* (Σ, ρ) *satisfies the* sequential behavioral conflict (SBhC) condition w.r.t. M_0 *if for any two transitions* t *and* t' *the following property, denoted* $SBhC(t, t')$*, holds:*

$$\neg SE(t,t') \Rightarrow \begin{array}{l} 1.\ t\,\bar{\rho}\,t'; \\ 2.\ TSE(t',t) \ \vee \ t,t' \in \Omega. \end{array}$$

Proposition 4.3 *A priority conflict-free Petri net satisfies the SBhC condition w.r.t. any initial marking.*

Remark 4.1 *Regarding the relations between the four conditions, we obtain the diagram in Figure 4.1. A solid arrow denotes an implication, while a dashed line denoted an incomparably (CF = "conflict-freeness").*

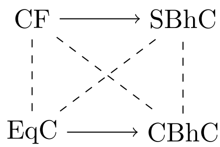

Figure 4.1: Relationships between the EqC, CF, CBhC, and SBhC conditions

Lemma 4.1 (Interchanging lemma) *Let* (Σ, ρ) *be a priority Petri net,* M_0 *a marking of* Σ*, and* w *a non-empty transition sequence such that we have* $M_0[*\rangle M[w\rangle M'$ *for some markings* M *and* M'*. If* (Σ, ρ) *satisfies the EQUAL-conflict condition, the conflict-freeness condition, the CBhC condition, or the SBhC condition, and* $T(M') \subseteq \Omega$*, then there exists a permutation* w' *of* w *such that* $M[w'\rangle_\rho M'$*.*

Corollary 4.1 *Let* (Σ, ρ) *be a priority Petri net,* M_0 *a marking of* Σ*, and* M *a home marking of* Σ *w.r.t.* M_0*. If* (Σ, ρ) *satisfies the EQUAL-conflict condition, the conflict-freeness condition, the CBhC condition, or the SBhC condition w.r.t.* M_0 *and* $T(M) \subseteq \Omega$*, then* M *is a home marking of* (Σ, ρ) *w.r.t.* M_0*.*

Corollary 4.2 *Let (Σ, ρ) be a priority WF net and $k \geq 1$. If Σ is k-sound and satisfies the EQUAL-conflict condition, the conflict-freeness condition, the CBhC condition, or the SBhC condition w.r.t. M_{ki}, then (Σ, ρ) is k-sound.*

Corollary 4.3 *Let (Σ^r, ρ) be a priority RCWF net, $k \geq 1$, and R a marking on S^r. If Σ^r is (k, R)-sound and satisfies the EQUAL-conflict condition, the conflict-freeness condition, the CBhC condition, or the SBhC condition w.r.t. (M_{ki}, R), then (Σ^r, ρ) is (k, R)-sound.*

4.4.1 Decidability of the CBhC and SBhC Conditions

The EqC and conflict-freeness conditions can be checked syntactically. The CBhC and SBhC conditions must consider the behavior of the Petri net. This makes the verification process more complex.

Theorem 4.3 *The CBhC and SBhC conditions for priority Petri nets are decidable in exponential space.*

4.5 Relaxing the Priority Relation

If we give up to the requirement that $\bar{\rho}$ is an equivalence relation from the definition of the priority relation, the interchanging lemmata in Section 4.5 do not hold. For these lemmas still hold we have to strengthen the CBhC and SBhC conditions by replacing $(\forall M \in [M_0\rangle)(M[t\rangle \Rightarrow M[t'\rangle)$ in Definition 4.5(3a) and Definition 4.7(3a) by $(\forall M \in [M_0\rangle)(M[t\rangle_\rho \Rightarrow M[t'\rangle_\rho)$

With this modification, the CBhC (SBhC) condition will be called the *generalized CBhC (SBhC) condition*. The generalized CBhC and SBhC conditions are decidable in exponential space.

Chapter 5

Workflow Nets with Time, Resource, and Task Priority Constraints

In this chapter we introduce *multi-level workflow nets with resource constraints* (mlRCWF nets) as the composition of standard workflow nets with resource constraints. The soundness of mlRCWF nets can be reduced to the soundness of standard RCWF nets [122].

For that model, time constraints for mlRCWF nets are time durations. It is shown that the soundness property for *timed priority mlRCWF nets* can be reduced to soundness of untimed mlRCWF nets with priorities [122].

Sometimes is necessary that a resource to be relocated. We considered *time priority mlRCWF nets with resource relocation.* We showed that the soundness property of these nets can be reduced to the soundness property of untimed priority mlRCWF nets if a task can be indefinitely delayed.

5.1 Multi-level Resource Constrained Workflows

In many practical situations, workflows must share common resources. Consider an office equipped with a printer machine (PM), a printer-copy machine (PCM), and a printer-copy-fax machine (PCFM). Figure 5.1 illustrates the resulting net.

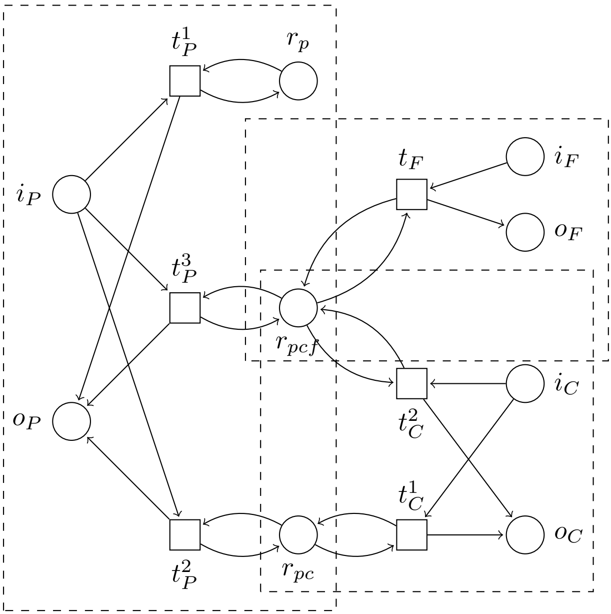

Figure 5.1: Inter-connecting workflow nets

In Figure 5.1 we notice the three inter-connected workflows. The transitions for printing jobs are t_P^1 (on PM), t_P^2 (on PCM), and t_P^3 (on $PCFM$), the transitions for coping jobs are t_C^1 (on PCM) and t_C^2 (on $PCFM$), and the transition for faxing jobs is t_F. We can consider the resulted Petri net (which is not a workflow net) as a composition of three workflow nets, in order to process correctly three kinds of jobs which share common resources.

Two Petri nets $\Sigma_1 = (S_1, T_1, F_1, W_1)$ and $\Sigma_2 = (S_2, T_2, F_2, W_2)$ are called *S-compatible*, where S is a set of places, if $S_1 \cap S_2 = S$ and $T_1 \cap T_2 = \emptyset$. If Σ_1 and Σ_2 are S-compatible then they can be composed along S resulting the Petri net $\Sigma_1 \circ_S \Sigma_2 = (S_1 \cup S_2, T_1 \cup T_2, F_1 \cup F_2, W_1 \cup W_2)$. This Petri net is called the *S-composition* of Σ_1 and Σ_2.

Definition 5.1 *The class of* multi-level resource constrained workflow *(mlRCWF)* nets *is the least class of Petri nets which satisfies:*

1. *any resource constrained workflow net Σ^r is a multi-level resource constrained workflow net whose input (output, resource) places are exactly the input (output, resource) places of Σ^r;*
2. *if Σ^r_1 and Σ^r_2 are two R-compatible multi-level resource constrained workflow nets, where R is a set of common resource places, then $\Sigma^r_1 \circ_R \Sigma^r_2$ is a multi-level resource constrained workflow net whose input (output, resource) places are exactly the input (output, resource) places of Σ^r_1 and Σ^r_2.*

A mlRCWF net which is a composition of $n \geq 1$ RCWF nets will be called an *n-level resource constrained workflow* (n-lRCWF) *net*. It is obvious that any 1-lRCWF net is an RCWF net.

Example 5.1 *The Petri net in Figure 5.1 is a 3-level workflow net obtained by composing three resource constrained workflow nets. It has three input places, i_P, i_C, and i_F, three output places o_P, o_C, and o_F, and three resource places r_p, r_{pc}, and r_{pcf}.*

Let $\overline{k} = (k_1, \ldots, k_n)$ be a vector of non-negative integers. $M_{\overline{k}i}$ ($M_{\overline{k}o}$) denotes the marking which marks each input (output) place i_j (o_j) by k_j, $1 \leq j \leq n$, and leaves unmarked all the other places.

Definition 5.2 *Let Σ^r be an n-lRCWF net, $\overline{k} = (k_1, \ldots, k_n) > \overline{0}$ a tuple of non-negative integers, and $R \geq 1$ a marking on S^r. Σ^r is called $(\overline{k}, R)$-*sound *if, for any $(M, R') \in [M_{\overline{k}i}, R\rangle$, the following properties hold:*

1. *$R' \leq R$;*
2. *$(M_{\overline{k}o}, R) \in [M, R'\rangle$.*

Definition 5.3 *Let Σ^r be an n-lRCWF net, $\overline{k} = (k_1, \ldots, k_n) > \overline{0}$ a tuple of positive integers, and $R \geq 1$ a marking on S^r. The* $(\overline{k}, R)$-closure of *Σ^r is the Petri net $(\Sigma^r)^*$ obtained from Σ^r by adding a new transition t^*, new arcs (o_j, t^*) and (t^*, i_j) with the weight k_j, for all $1 \leq j \leq n$, and new arcs (r, t^*) and (t^*, r) with the weight $R(r)$ for any resource place r.*

Proposition 5.1 *Let Σ^r be an n-lRCWF net, $\overline{k} = (k_1, \ldots, k_n) > \overline{0}$ a tuple of positive integers, and $R \geq 1$ a marking on S^r. The following two properties hold:*

1. *Σ^r is $(\overline{k}, R)$-sound iff Σ^r is R-bounded on S^r w.r.t. $(M_{\overline{k}i}, R)$, its $(\overline{k}, R)$-closure is bounded w.r.t. $(M_{\overline{k}i}, R)$, and t^* is live w.r.t. $(M_{\overline{k}i}, R)$.*

2. *Σ^r is $(\overline{k}, R)$-sound iff Σ^r is R-bounded on S^r w.r.t. $(M_{\overline{k}i}, R)$ and $(M_{\overline{k}o}, R)$ is a home marking of Σ^r w.r.t. $(M_{\overline{k}i}, R)$.*

Corollary 5.1 *The $(\overline{k}, R)$-soundness is decidable for mlRCWF nets.*

Definition 5.4 *An mlRCWF net Σ^r is called* generalized sound *if, there exists $R \geq 1$ such that Σ^r is $(\overline{k}, R')$-sound, for all $\overline{k} > \overline{0}$ and $R' \geq R$.*

5.2 Timed Priority RCWF Nets

We shall consider time as discrete and it will be represented by the set $\mathbb{N}$ of non-negative integers. To model time constraints we split each transition t into two parts, t^+ which initiates the activity modeled by t, and t^- which closes this activity [117, 118, 119]. A transition t has associated a time duration $\delta(t)$. After t^+ fires, the transition t will be included into a set of current transitions, and it will remain in this set for at least $\delta(t)$ time units. Priority constraints will be included as in [121].

Definition 5.5 *A* timed priority Petri net *(TPPN) is a pair $\gamma = (\Sigma, \rho, \delta)$, where Σ is a Petri net called the* underlying net *of γ, ρ is a* priority relation on *T (i.e., ρ is an irreflexive, asymmetric, and transitive binary relation on T), and $\delta : T \to \mathbb{N}$ is a function called the* time duration function *of γ.*

In what follows we consider the *type L behavior* of timed Petri nets [117] together with the standard behavior of priority Petri nets [121], and we write $(M, C, \tau)[e\rangle(M', C', \tau')$ if one of the following cases applies:

1. if $e = t^+$ then:

   (a) $t \notin C$ and $M[t\rangle$;

   (b) $(\forall t')(t' \in T - C \wedge M[t'\rangle \Rightarrow \neg(t \,\rho\, t'))$;

   (c) $M'(s) = M(s) - W(s,t)$ for all s, $C' = C \cup \{t\}$, $\tau'(t) = \delta(t)$, and $\tau'|_C = \tau|_C$;

2. if $e = t^-$ then:

   (a) $t \in C$ and $\tau(t) = 0$;

   (b) $M'(s) = M(s) + W(t,s)$ for all s, $C' = C - \{t\}$, and $\tau' = \tau|_{C'}$;

3. if $e = (\nu)$ for some $\nu \in N - \{0\}$, then $M' = M$, $C' = C$, and $\tau' = \tau \dot{-} \nu$.

1(a)(b) is the *enabling rule* for t^+ and 2(a) is the *enabling rule* for t^-; 1(c) and 2(b) are the *computation rules* for t^+ and t^-, respectively. Rule 3 models the time passing.

The concept of *reachability* for TPPN is defined in a similar manner as for classical Petri nets. $[(M, C, \tau)\rangle$ denotes the set of all reachable states from (M, C, τ).

Example 5.2 *For the Petri net in Figure 5.1 we add the time duration by function δ given by $\delta(t_F) = 1$, $\delta(t_C^1) = \delta(t_C^2) = 2$, $\delta(t_P^1) = \delta(t_P^2) = \delta(t_P^3) = 5$, and the priority relation ρ given by $t_P^2 \,\rho\, t_P^1$, $t_P^3 \,\rho\, t_P^1$, $t_P^3 \,\rho\, t_P^2$, $t_C^2 \,\rho\, t_C^1$, $t_P^2 \,\rho\, t_C^1$, $t_P^3 \,\rho\, t_C^2$, $t_P^3 \,\rho\, t_F$, $t_C^2 \,\rho\, t_F$.*

A *timed priority mlRCWF net* is any TPPN having the underlying net a mlRCWF net. States of timed priority mlRCWF net will be written in the form $((M, R), C, \tau)$.

Definition 5.6 *Let $\gamma = (\Sigma^r, \rho, \delta)$ be a timed priority mlRCWF net, $\overline{k} > \overline{0}$ a tuple of non-negative integers, and $R \geq 1$ a marking on S^r. γ is called* $(\overline{k}, R)$-sound *if for any $((M, R'), C, \tau) \in [((M_{\overline{k}i}, R), \emptyset, \emptyset)\rangle$ the following properties hold:*

1. $R' \leq R$;

2. $((M_{\overline{k}o}, R), \emptyset, \emptyset) \in [((M, R'), C, \tau)\rangle$.

Proposition 5.2 *A timed priority mlRCWF net* $\gamma = (\Sigma^r, \rho, \delta)$ *is* $(\overline{k}, R)$*-sound if and only if* γ *is* R*-bounded on* S^r *and* $((M_{\overline{k}o}, R), \emptyset, \emptyset)$ *is a home state of* γ *w.r.t.* $((M_{\overline{k}i}, R), \emptyset, \emptyset)$.

Proposition 5.3 *The* $(\overline{k}, R)$*-soundness property of timed priority multi-level RCWF nets is equivalent to the* $(1, R)$*-soundness property of priority RCWF nets in the sense that for any timed priority mlRCWF net* $\gamma = (\Sigma, \rho, \delta)$ *there exists a priority RCWF net* $\gamma' = (\overline{\Sigma}^r, \rho')$ *such that* γ *is* $(\overline{k}, R)$*-sound if and only if* γ' *is* $(1, R)$*-sound.*

Corollary 5.2 *The* $(\overline{k}, R)$*-soundness property of timed priority mlRCWF nets is undecidable.*

We identified some classes of priority RCWF nets with soundness decidable in Section 4.4. If we add time constraints to the priority RCWF nets in these classes accordingly to the Definition 5.5, we obtain timed priority RCWF nets with soundness decidable (Corollary 5.3).

5.3 Resource Relocation

Definition 5.7 *A* timed priority mlRCWF net with resource relocation *is a tuple* $\gamma = (\Sigma^r, \rho, \delta, \theta)$*, where* (Σ^r, ρ, δ) *is a timed priority mlRCWF net and* $\theta : T \to \mathbb{N} \cup \{\infty\}$ *is a function, called the* delay function*, which gives the maximum amount of time a transition can be delayed.*

The case $\theta(t) = \infty$ means that the transition t can be indefinitely delayed.

A *state* of a timed priority mlRCWF net γ with resource relocation is a tuple $((M, R), C, \tau, D, \eta)$, where:

- (M, R) is the current marking of γ;
- $C \subseteq T$ is the set of current transitions;
- $\tau : C \to \mathbb{N}$ is the residual time function;

- $D \subseteq T \times T \times \mathbb{N}$ is a set of triples (t, t', x) whose meaning is that the transition t has delayed the transition $t' \in C$ when the residual time of t' was x;

- $\eta : C \cup pr_2(D) \to \mathbb{N}$ is a function which gives the maximum delay time the transitions in $C \cup pr_2(D)$ still support.

When we have $\theta(t) = \infty$ for all transitions t, the tuples $(\Sigma^r, \rho, \delta, \theta)$ and $((M, R), C, \tau, D, \eta)$ can be simplified to (Σ^r, ρ, δ) and $((M, R), C, \tau, D)$, respectively (with the meaning above). In such a case we say that (Σ^r, ρ, δ) is a *timed priority mlRCWF net with resource relocation and indefinite delay*.

We write $((M, R), C, \tau, D, \eta)[e\rangle_\gamma((M', R'), C', \tau', D', \eta')$ if one of the following cases applies:

1. if $e = t^+$ then:

   (a) $t \notin C$ and $(M, R)[t\rangle$;

   (b) $(\forall t')(t' \in T - C \wedge (M, R)[t'\rangle \Rightarrow \neg(t \,\rho\, t'))$;

   (c) $M'(s) = M(s) - W(s, t)$ and $R'(r) = R(r) - W(r, t)$ for all $s \in S$ and $r \in R$;

   (d) $C' = C \cup \{t\}$ and $D' = D$;

   (e) $\tau'(t) = \delta(t)$ and $\tau'|_C = \tau|_C$;

   (f) $\eta'(t) = \theta(t)$ and $\eta'|_{C \cup pr_2(D)} = \eta|_{C \cup pr_2(D)}$;

   (The rules 1(a-b) mean that t is not a current transition, is enabled at the current marking, and no other transition in $T - C$ which is enabled at the current marking has priority over t. The rules 1(c-f) mean that t becomes current (enters in C), its residual time is set to $\delta(t)$, its delay time is set to $\theta(t)$, and the current marking is properly updated);

2. if $e = d_{t,t'}$, where t and t' are distinct transitions, then:

   (a) $t \notin C$, $M(s) \geq W(s, t)$ for all $s \in S$, and $R(s) < W(r, t)$ for some $r \in R$;

   (b) $(\forall t')(t' \in T - C \wedge (M, R)[t'\rangle \Rightarrow \neg(t \,\rho\, t'))$;

(c) $t' \in C - pr_2(D)$ and:

- $t' \,\rho\, t$ (i.e., t has priority over t');
- $W(r,t') \geq W(r,t)$ for all $r \in R$ (i.e., t can use the resources allocated to t');
- $W(r,t) = W(t,r)$ for all $r \in R$ (i.e., t returns all resources when it ends);
- $\eta(t') \geq \delta(t)$ (i.e., the delay supported by t' allows t's execution);

(d) $M'(s) = M(s) - W(s,t)$ for all $s \in S$, and $R' = R$;

(e) $C' = (C - \{t'\}) \cup \{t\}$ and $D' = D \cup \{(t, t', \tau(t'))\}$;

(f) $\tau'(t) = \delta(t)$ and $\tau'|_{C-\{t'\}} = \tau|_{C-\{t'\}}$;

(g) $\eta'(t) = \theta(t)$ and $\eta'|_{C \cup pr_2(D)} = \eta|_{C \cup pr_2(D)}$;

(The rules 2(a-b) illustrate the case when the transition t is not current, it is enabled to the marking M, but there are not enough resources to fire, and no other transition in $T - C$ which is enabled at the current marking has priority over t. The current transition t' (2(c)) over which t has priority, has enough resources, and its delay time is sufficient to support the execution of t. The rules 2(d-g) mean that t becomes current, the residual and delay time of t is set to maximum, t' enters in the set of delayed transitions together with its current residual time, and the set of current markings is updated);

3. if $e = t^-$ then:

(a) $t \in C$ and $\tau(t) = 0$;

(b) $M'(s) = M(s) + W(s,t)$ for all $s \in S$, $R' = R$ if $t \in pr_1(D)$, and $R'(r) = R(r) + W(r,t)$ for all $r \in R$, otherwise;

(c) $C' = (C - \{t\}) \cup \{t' | \exists x : \ (t, t', x) \in D\}$;

(d) $D' = D - \{(t, t', x) | \exists t', x : \ (t, t', x) \in D\}$;

(e) $\tau'(t'') = \tau(t'')$ for all $t'' \in C - \{t\}$, and $\tau'(t') = x$ where t' and x satisfy $(t, t', x) \in D$;

(f) $\eta'|_{C' \cup pr_2(D')} = \eta|_{C' \cup pr_2(D')}$;

(3(a) models the case when t is a current transition whose residual time is zero. The rules 3(b-f) mean that t closes its activity (exits from C), the transition delayed by t becomes now current, and the current marking is correspondingly updated);

4. if $e = (\nu)$ for some $\nu \in N - \{0\}$ then $M' = M$, $R' = R$, $C' = C$, $D' = D$, $\tau' = \tau \dot{-} \nu$, and $\eta' = \eta \dot{-} \nu$ (The passage of time affects only the residual time of the current transitions and the delay time of the delayed transitions.).

The soundness property is defined in a similar manner as for timed priority mlRCWF nets.

Proposition 5.4 *The* $(\overline{k}, R)$*-soundness property of timed priority mlRCWF nets with resource relocation and indefinite delay is equivalent to the* $(1, R)$*-soundness property of priority RCWF nets in the sense that for any timed priority mlRCWF net* $\gamma = (\Sigma, \rho, \delta)$ *with resource relocation and indefinite delay there exists a priority RCWF net* $\gamma' = (\overline{\Sigma}^r, \rho')$ *such that* γ *is* $(\overline{k}, R)$*-sound if and only if* γ' *is* $(1, R)$*-sound.*

Conclusions and Future Work

This thesis has focused on the the modeling and verification of the workflows with resources, priority, and time constraints. We chose the Petri net formalism to model these workflows.

We analyzed the soundness problem for *resource-constrained workflow nets* (RCWF nets). We gradually refined the soundness criteria for RCWF nets, and we grouped them into three categories: *soundness criteria under specified resources*, *soundness criteria under unspecified resources*, and *structural soundness*. Soundness criteria under specified resources suppose that some marking for the resource places is given, and we have to decide whether the RCWF net is sound with respect to that marking. These soundness criteria were studied using closure nets and instantiation nets, and it was shown that they are decidable. For the case of the soundness criteria under unspecified resources, the main question is to decide whether there is a marking on resource places that makes the RCWF net sound. We showed that k-soundness (i.e., k-soundness with respect to a minimal resource marking) is decidable. Structural R-soundness was also proved to be decidable. Generalized soundness was proved decidable in [108]. We are interested in find a proof for the general soundness property based on instantiation nets.

A natural idea is to add priorities to RCWF nets. We proposed *priority (resource-constrained) workflow nets* (P(RC)WF nets). We used deterministic counter machines to prove that the soundness property for priority (resource-constrained) WF nets is undecidable. If additional conditions are imposed, the soundness for priority (resource-constrained) workflow nets can be reduced to the soundness of the underlying (resource-constrained) workflow net. The first condition we proposed is the *CBhC condition* which generalizes the *EQUAL-conflict condition* and the second one is the *SBhC condition* which generalizes the *conflict-freeness condition*. Using the Petri nets path logic we showed that the CBhC and SBhC conditions are decidable. For the future we expect to find more classes of priority (resource-constrained) workflow nets with soundness decidable.

There are situations when some workflows must share resources. This led us to the idea to "compose" workflow nets, and we obtained *multi-level workflow nets with resource constraints* (mlRCWF nets). We studied the decidability of the soundness problem for mlRCWF nets with priorities and time durations associated to tasks. The soundness of *timed priority mlRCWF nets* is undecidable because it can be reduced in linear time to the corresponding untimed priority model. Considering the fact that sometimes a resource used by a task must be released because it is needed by a task with a higher priority, we described a *resource relocation* policy. We have shown that soundness of *timed priority mlRCWF nets with resource relocation and indefinite delay* can be reduced in quadratic time to soundness of priority mlRCWF nets. The indefinite delay may not be appropriate in some real cases. Therefore finding an algorithmic characterization of soundness of timed priority mlRCWF nets with resource relocation and bound task delay is of a real interest.

We believe that this thesis makes a step forward in workflow modeling, because it manages to combine into a suitable model resource, priorities, and time constraints, analyzing the specific soundness properties.

Bibliography

[1] V.M.P. van der Aalst, *Structural Characterization of Sound Workflow Nets*, Computing Science Report 96/23, Eindhoven University of Technology, 1996.

[2] W.M.P. van der Aalst, *Three Good Reasons for Using a Petri net based Workflow Management System*, Proceedings of the International Working Conference on Information and Process Integration in Enterprises (IPIC'96), Cambridge, Massachusetts, 1996, 179–201.

[3] W.M.P. van der Aalst, *The Application of Petri Nets to Workflow Management*, The Journal of Circuits, Systems and Computers, 8(1), 1998, 21–66.

[4] W. van der Aalst and K. van Hee, *Workflow Management: Models, Methods, and Systems*, The MIT Press, Cambridge, Massachusetts, 2002.

[5] M. Ajmone-Marsan, G. Balbo, G. Conte, S. Donatelli, and G. Franceschinis, *Modelling with Generalized Stochastic Petri Nets*, Wiley Series in Parallel Computing. John Wiley and Sons, 1994.

[6] W. M. P. van der Aalst, K. M. van Hee, A. H. M. der Hofstede, N. Sidorova, H. M. W. Verbeek, M. Voorhoeve, M. T. Wynn, *Soundness of Workflow Nets: Classification, Decidability, and Analysis*, Formal Aspects of Computing, vol. 23(3), 2011, 333–363.

[7] N. R. Adam, V. Atluri, W. K. Huang, *Modeling and Analysis of Workflows Using Petri Nets*, Journal of Intelligent Information Systems, vol. 10(2), March 1998, 131–158.

[8] T. Araki, T. Kasami, *Decidable Problems of the Strong Connectivity of Petri Nets Reachability sets*, Theoretical Computer Science, vol. 4, 1977, 99–119.

[9] T. Araki and T. Kasami, *Some Decision Problems Related to the Reachability Problem for Petri Nets*, Theoretical Computer Science, vol. 3(1), 1977, 85–104.

[10] M.F. Atig, P. Habermehl, *On Yen's Path Logic for Petri Nets*, International Journal of Foundations of Computer Science, vol. 22(4), 2011, 783–799.

[11] K. Barkaoui, I. Ben Abdallah, *Analysis of a Resource Allocation Problem in FMS Using Structure Theory of Petri Nets*, Proc. of the 1st International Workshop on FMS and Petri Nets, Osaka (Japan), 1996.

[12] K. Barkaoui, M. Minoux, *Polynomial-Time Graph Algorithms for Proving or Disproving Commoner's Structural Property in Petri Nets*, Rapport de recherce, N. 88-04, Conservatoire National des Arts et Metiers, Paris, France, 1987.

[13] K. Barkaoui, M. Minoux, *A Polynomial-Time Graph Algorithm to Decide Liveness of Some Basic Classes of Bounded Petri Nets*, Proceedings of the 13th International Conference on Application and Theory of Petri Nets, Lecture Notes in Computer Science, vol. 616, 1992, 62–75.

[14] K. Barkaoui, J. M. Couvreur, C. Dutheillet, *On liveness in Extended Non Self-Controlling Nets*, Proceedings of the 16th International Conference on Application and Theory of Petri Nets, Lecture Notes in Computer Science, vol. 935, 1995, 25–44.

[15] K. Barkaoui and J.F. Pradat-Peyre, *On Liveness and Controlled Siphons in Petri Pets*, Proceedings of the 17th International Conference on Application and Theory of Petri Nets, Lecture Notes in Computer Science, vol. 1091, 1996, 57–72.

[16] K. Barkaoui, L. Petrucci, *Structural Analysis of Workflow Nets with Shared Resources*, Proceeding of the Workshop "Workflow

Management: Net-based Concepts, Models, Techniques and Tools WFM'98", Lisbon (Portugal), June 1998, 82–95.

[17] K. Barkaoui, R.B. Ayed, Z. Sbai, *Workflow Soundness Verification Based on Structure Theory of Petri Nets*, International Journal of Computing and Information Sciences 5(1), 2007, 51–61.

[18] K. Barkaoui, J.M. Couvreur, K. Klai, *On the Equivalence between Liveness and Deadlock-Freeness in Petri Nets*, ICAPTN '05, ser. Lecture Notes in Computer Science, vol. 3536, 2005, 1121–1126.

[19] K. Barkaoui, R.B. Ayed, *Uniform Verification of Workflow Soundness*, Transactions of the Institute of Measurement and Control, 2010, 1–16.

[20] A. Basu, Ph. Bidinger, M. Bozga, J. Sifakis, *Distributed Semantics and Implementation for Systems with Interaction and Priority*, in Formal Techniques for Networked and Distributed Systems – FORTE'08, ser. Lecture Notes in Computer Science, vol. 5048, 2008, 116–133.

[21] F. Bause, *On the Analysis of Petri Nets with Static Priorities*, Acta Informatica, vol. 33, 1996, 669–685.

[22] F. Bause, *Analysis of Petri Nets with a Dynamic Priority Method*, Proceedings of the 18th International Conference on Application and Theory of Petri Nets (ICATPN'97), Toulouse, France, Eds. P. Azma and G. Balbo, June 1997, ser. Lecture Notes in Computer Science, vol. 1248, 215–234.

[23] F. Bause, P. S. Kritzinger, *Stochastic Petri Nets - An Introduction to the Theory*, Vieweg Monographs, 2002.

[24] B. Berthomieu, M. Diaz, *Modeling and Verification of Time Dependent Systems Using Time Petri Nets*, IEEE Transactions on software Engineering, Vol. 17(3), March 1991, 259–173.

[25] E. Best, Maciej Koutny, *Petri Net Semantics of Priority Systems*, Theoretical Computer Science, 1992, 175–215.

[26] G. Bhat, R. Cleaveland, G. Lttgen, *Dynamic Priorities for Modeling Real-Time*, in Proc. of the Formal Description Techniques and Protocol Specification, Testing and Verification (FORTE X/PSTV XVII '97), Osaka, 1996, 321–336.

[27] C. Bocăneală, *On Coverability Structures for Nested Petri Nets*, Proceedings of The Third International Conference on Mathematical Sciences, ICM2008, 3-6 march 2008, Al-Ain, United Arab Emirates, Volume 1, 234–243.

[28] C. Bocăneală, *n-Levels Petri Nets*, The Annals of the University "Dunărea de Jos" of Galaţi, Fascicle II, Mathematics, Physics, Chemistry, Informatics, vol. XXXI, 2008, 105–114.

[29] C. Bocăneală, *Termination Problem for Petri Nets with multi Levels*, The Annals of the University "Dunărea de Jos" of Galaţi, Fascicle II, Mathematics, Physics, Chemistry, Informatics, vol. XXXI, 2008, 118–125.

[30] C. Bocăneală, *Modeling Intelligent Systems with Level Petri Nets*, The Annals of the University "Dunărea de Jos" of Galaţi, Fascicle III, Electrotechnics, Electronics, Automatic Control and Informatics, vol. 2, 2008, 31–36.

[31] C. Bocăneală, *Place Invariants for Nested Petri Nets*, The Annals of the University "Dunărea de Jos" of Galaţi, Fascicle II, Mathematics, Physics, Theoretical Mechanics, vol. XXXII, 2009, 209–222.

[32] C. Bocăneală, *T-Invariants for Nested Petri Nets*, The Annals of the University "Dunărea de Jos" of Galaţi, Fascicle II, Mathematics, Physics, Theoretical Mechanics, Facicle II, Year II (XXXIV), 2010, 345–353.

[33] M. Boyer, O. H. Roux, *Comparison of the Expressiveness of Arc, Place and Transition Time Petri Nets*, Lecture notes in computer Science, Vol. 4546, Springer Verlag, 2007, 63–82.

[34] M. Boyer, O. H. Roux, *On the Compared Expressiveness of Arc, Place and Transition Time Petri Nets*, Fundamenta Informaticae, Vol. 88, 2008, 225–249.

[35] P. Bremond-Gregoire and I. Lee, *A Process Algebra of Communicating Shared Resources with Dense Time and Priorities*, Theoretical Computer Science, vol. 189(1-2), 1997, 179–219.

[36] G. Bucci, A.Fedeli, L.Sassoli, E. Vicario, *Timed State Space Analysis of Real-Time Preemptive Systems*, IEEE Transactions on Software Engineering, vol. 30(2), February 2004, 97–111.

[37] G. Bucci, L. Sassoli, E. Vicario, *Correctness Verification and Performance Analysis of Real Time Systems Using Stochastic Preemptive Time Petri Nets*, IEEE Transactions on Software Engineering, vol. 31(11), November 2005, 913–927.

[38] A. Cheng, J. Esparza, J. Palsberg, *Complexity Results for 1-safe nets*. 13th Conference on Foundations of Software Technology and Theoretical computer Science, Bombay, 1993.

[39] G. Chiola, M. Ajmone-Marsan, G. Balbo, G. Conte, *Generalized Stochastic Petri Nets: A Definition at the Net Level and its Implications*, IEEE Transactions on Software Engineering, vol. 19, 1993, 89–107.

[40] G. Chiola, S. Donatelli, G. Franceschinis, *Priorities, Inhibitor Arcs, and Concurrency in P/T Nets*, Proceedings of the 12th International Conference on Application and Theory of Petri Nets, Gjern, Denmark, June, 1991, 182–205.

[41] R. Cleaveland and M. Hennessy, *Priorities in Process Algebras*, Information and Computation, vol. 87(1-2), 1990, 58–77.

[42] R. Cleaveland and G.Y. Luttgen and V. Natarajan and S. Sims, *Modeling and Verifying Distributed Systems Using Priorities: A Case Study*, Software Concepts and Tools, vol. 17, 1996, 50–62.

[43] A. David, J. Håkansson, K. G. Larsen, P. Petterson, *Model Checking Timed Automata with Priorities Using DBM Substraction*, In 4th International Conference on Formal Modelling and Analysis of Timed Systems (FORMATS'06), ser. Lecture Notes in Computer Science, E. Asarin and P. Bouyer, Eds., Springer Verlag, September 2006, 128–142.

[44] J. Desel, J. Esparza, *Free-Choice Petri Nets*, Cambridge Tracts in Theoretical Computer Science, nr. 40, Cambridge University Press, 1995.

[45] Y. Du, C. Jiang, *Towards a Workflow Model of Real-Time Cooperative Systems*, Proceedings of ICFEM, 2003, vol. 2885, 452–470.

[46] D.F. Escrig, C. Johnen, *Decidability of Home Space Property*, Universite Paris-Sud, LRI Report 503, 1989.

[47] J. Esparza, M. Nielsen, *Decidability Issues for Petri Nets*, BRICS Report Series RS-94-8, 1994.

[48] J. Esparza, *Decidability and Complexity of Petri Net Problems – An Introduction*, in "Lectures on Petri Nets I: Basic Models. Advances in Petri Nets" (G. Rozenberg and W. Reisig, eds.), Lecture Notes in Computer Science 1491, 1998, 374–428.

[49] J. Esparza, M. Silva, *A Polynomial-Time Algorithm to Decide Liveness for Bounded Free Choice Nets*, Theoretical Computer Science, 102, 1992, 185–205.

[50] J. Ezpeleta, J.M. Colom, J. Martinez, *A Petri Net Based Deadlock Prevention Policy for Flexible Manufacturing Systems*, IEEE Transactions on Robotics and Automation, vol. 11(2), 1995, 173–183.

[51] E. Fersman, P. Petterson, W. Yi, *Timed Automata with Asynchronous Processes: Schedulability and Decidability*, In Proceedings of the 8th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS 2002), ser. Lecture Notes in Computer Science, J. P. Katoen and P. Stevens, Eds., vol. 2280, Springer-Verlag, 2002, 67–82.

[52] D. Frutos-Escrig, *Decidability of Home States in Place Transition Systems*, Internal report. Dpto. Informatica y Automatica, University Complutense de Madrid, 1986.

[53] D. Frutos-Escrig, C. Johnen, *Decidability of Home Space Property*, Report LRI-503, Univ. de Paris-Sud, Centre d'Orsay, Laboratoire de Recherche en Informatique, 1989.

[54] C. Frye, *Move to Workflow provokes Business Process Scrutiny*, Software Magazine, 1994.

[55] D. Georgakopoulos, M. Hornick, A. SHeth, *An Overview of Workflow Management: From Process Modeling to Workflow Automation Infrastructure*, Distributed and Parallel Databases, vol. 3, 1995, 119–153.

[56] C. Girault, R. Valk, *Petri Nets for Systems Engineering. A Guide to Modeling, Verification, and Applications*, Springer-Verlag, 2003.

[57] M. Hack, *Petri Net Languages*, Massachusetts Institute of Technology, Cambridge, MA, USA, Tech. Rep., 1976.

[58] M.H.T. Hack, *Decidability Questions for Petri Nets*, Ph.D. Thesis, M.I.T., 1976.

[59] K. van Hee, N. Sidorova, M. Voorhoeve, *Generalized Soundness of Workflow Nets is Decidable*, Proc. 25th Int. Conf. on Application and Theory of Petri Nets, Bologna, Italy, Lecture Notes in Computer Science 3099, 2004, 197–216.

[60] K. van Hee, N. Sidorova, M. Voorhoeve, *Resource-Constrained Workflow Nets*, In Proc. of Concurrency Specification and Programming, Caputh, September 24-26, 2004, Informatik-Bericht Nr. 170 (G. Lindemann, ed.), Humboldt-Universität zu Berlin, 166–177 (also in Fundamenta Informaticae, 71(2-3), 2006, 243–257).

[61] K. van Hee, A. Serebrenik, N. Sidorova, M. Voorhoeve, *Soundness of Resource-Constrained Workflow Nets*, In Proc. of the 26th International Conference on Application and Theory of Petri Nets, Lecture Notes in Computer Science (G. Ciardo, Ph. Darondeau, eds.), vol. 3536, 2005, 250–267.

[62] K. van Hee, N. Sidorova, M. Voorhoeve, *Resource-Constrained Workflow Nets*, Fundamenta Informaticae, 71(2-3), 2006, 243–257.

[63] P. A. Hsiung, S. W. Lin, *Model Checking Timed Systems with Priorities*, In Proceedings of the 11th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications, 2005, 539–544.

[64] R. Janicki, *A formal Semantics for Concurrent Systems with a Priority Relation*, Acta Informatica, vol. 24(1), 1987, 33–55.

[65] K. Jensen, *Colored Petri Nets and the Invariant-Method*, Theoretical Computer Science, vol. 14(3), 1981, 317-336.

[66] K. Jensen, *Colored Petri Nets*, Lecture Notes in Computer Science, vol. 254, 1987, 248–299.

[67] L. Jenner, *Further Studies on Timed Testing of Concurrent Systems*, Technical Report 1998-4, Institute fur Mathemätik, Universität Augsburg, 1998.

[68] Z. Jiang, M. C. Zhou, *Performance analysis of prioritized parallel transmission MAC protocol in all-IP wireless WAN*, International Journal Communication Systems. vol. 20(4), 2007, 477–490.

[69] T. Jucan, F.L. Ţiplea, *Petri Nets. Theory and Practice*, Romanian Academy Press, Bucharest, 1999.

[70] R.M. Karp, R.E. Miller, *Parallel Program Schemata*, Journal of Computer and System Science, vol. 3, 147–195, 1969.

[71] S.R. Kosaraju, *Decidability of Reachability in Vector Addition Systems*, Proceeding of the 14th Annual ACM Symposium on Theory of Computing, 267–281, 1982.

[72] C. Lakos, L. Petrucci, *Modular State Spaces for Prioritized Petri Nets*, Foundations of Computer Software. Modeling, Development, and Verification of Adaptive Systems, ser. Lecture Notes in Computer Science, Eds. R.Calinescu, E Jackson, Springer-Verlag, vol. 6662, 2011, 136–156.

[73] G. B. Lee, H. Zandong, J. S. Lee, *Generalized State Equation of Petri Nets with Priority*, International Journal of Intelligent Systems, vol. 18(11), Wiley, 2003, 1145–1153.

[74] Q. Li and Y. S. Fan and M. C. Zhou, *Approximate Performance Analysis of Workflow Model*, Proceedings of IEEE International Conference on Systems, Man and Cybernetics, IEEE Computer Society Press, vol. 2, 2003, 1175–1180.

[75] Q. Li, Y. S. Fan, M. C. Zhou, *Timing Constraint Workflow Nets for Workflow Analysis*, IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans, vol. 32(2), 2003, 179–193.

[76] J.Q. Li, Y.S. Fan, M.C. Zhou, *Performance Modeling and Analysis of Workflow*, IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans, vol. 34(2), 2004, 229–241.

[77] H. Li, Y. Yang, T.Y. Chen, *Resource Constraints Analysis of Workflow Specifications*, The Journal of Systems and Software, vol. 73, 2004, 271–285.

[78] Z. W. Li, N. Q. Wu, M. C. Zhou, *Deadlock Control of Automated Manufacturing Systems Based on Petri Nets - A Literature Rewiew*, SCH. Electro-Mech. Eng., Xidian Univ., Xi'an, China, 2010.

[79] S. Ling, H. Schmidt, *Time Petri Nets for Workflow Modeling and Analysis*, IEEE International Conference on Systems, Man, and Cybernetics (SMC'2000), vol. 4, 2000, 3039–3044.

[80] R.J. Lipton, *The reachability Problem Requires Exponential Space*, Department of Computer Science, Research Report 62, Yale University, 1976.

[81] I.A. Lomazova, *Nested Petri Nets a Formalism for Specification and Verification of Multi-Agent Distributed Systems*, Fundamenta Informaticae, vol. 43, 2000, 195–214.

[82] I.A. Lomazova, *Nested Petri Nets: Multi-Level and Recursive Systems*, Fundamenta Informaticae, vol. 47, 2001, 283–293.

[83] E.W. Mayr, *Persistence of Vector Replacement Systems is Decidable*, Acta Informatica, vol. 15, 1981, 309–318.

[84] E. Mayr, *An Algorithm for the General Petri Net Reachability Problem*, Proceeding of the 13th Annual ACM Symposium on Theory of Computing, 1981, 238–246.

[85] G. Memmi, G. Roucaiol, *Linear algebra in Net Theory*, Lecture Notes in Computer Science, vol. 84, 1980, 213–223.

[86] P. Merlin, *A Study of the Recoverability of Computing Systems*, Ph.D. Thesis, Dep. Of Information and Computer Science, University of California, Irvine, CA, 1974.

[87] P. Merlin, D. Farber, *Recoverability of Communication Protocols - Implications of a Theoretical Study*, IEEE Transactions on Communications, vol. COM-24(9), September 1976, 1036–1043.

[88] M. Minsky, *Recursive Unsolvability of Post's Problem of TAG and Other Topics in Theory of Turing Machines*, Annals of Mathematics, vol. 74 (3), 1961, 437–455.

[89] T. Murata, *Petri Nets: Properties, Analysis and Applications*, Proceedings of the IEEE, vol. 77, No. 4, 541–580, April 1989.

[90] M. Netjes, W. M .P van der Aalst, H. A. Reijers, *Analysis of Resource Constrained Processes with Colored Petri Nets*, Proceedings of 6th Workshop Pract. Use Coloured Petri Nets CPN Tools, K. Jensen, Ed. Aarhus, Denmark, Oct. 2005, 251–266.

[91] C. Oliveira, R. Lima, T. Andre, H. A. Reijers, *Modeling and Analyzing Resource-Constrained Business Processes*, Proceedings of IEEE International Conference on Systems, Man and Cybernetics, IEEE Computer Society Press, 2009.

[92] J. Peterson, *Petri Net Theory and the Modeling of Systems*, Prentice-Hall, Englewood Cliffs, 1981.

[93] S. Petkov, E. Oren, A. Haller, *Aspects in Workflow Management* DERI Technical Report 2005-04-10, 2005.

[94] C. A. Petri, *Kommunication mit Automaten*, Schriften des Institut fr Instrumentelle Mathematik, Bonn, (Germany), 1962.

[95] O. Prisecaru, *Resource Workflow Nets: a Petri Net Formalism for Workflow Modeling*, Proceedings of the 7th Int. Workshop MSVVEIS, Funchal, Madeira, Portugal, 2007, 11–21.

[96] O. Prisecaru, *The analysis of Resource Constrained Workflows Using Petri Nets*, Proceedings of the 8th Int. Workshop MSVVEIS, Funchal, Madeira, Portugal, 2010, 20–29.

[97] C. Rackoff, *The Covering and Boundedness Problem for Vector Addition Systems*, Theoretical Computer Science, 6, 1978, 223–231.

[98] C. Ramchandani, *Analysis of Asynchronous Concurrent Systems by Petri Nets*, Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, MA, Project MAC Report MAC-TR-120, 1974.

[99] W. Reisig, *Petri Nets. An Introduction*, EATCS Monographs on Theoretical Computer Science, Springer-Verlag, 1985.

[100] W. Reisig, G. Rozemberg (Eds.), *Lectures on Petri Nets I: Basic Models: Advances in Petri Nets* Lecture Notes in Computer Science, Springer, 2004.

[101] L.E. Rosier, H.C. Yen, *A Multiparameter Analysis of the Boundedness Problem for Vector Addition Systems*, Journal of Computer and System Sciences, 32, 1986, 105–135.

[102] V. V. Ruiz, D. de Frutos Escrig, F. C. Gomez, *Simulation of Timed Petri Nets by Ordinary Petri Nets and Applications to Decidability of the Timed Reachability Problem and Other Related Problems*, in Proceedings of the 4th International Workshop on Petri Nets and Performance Models, Melbourne, Australia, 1991, 154–163.

[103] V. V. Ruiz, D. de Frutos Escrig, F. C. Gomez, *Decidability of the Strict Reachability Problem for TPN's with Rational and Real Durations*, in Proceedings of the 5th International Workshop on Petri Nets and Performance Models, Melbourne, Australia, 1993, 56–65.

[104] W. Sadiq, M.E. Orlowska, *Applying Graph Reduction Techniques for Identifying Structural Conflicts in Process Models*, Proceedings of the 11th International Conference on Advanced Information Systems Engineering CAiSE99, Lecture Notes in Computer Science, vol. 1626, Springer-Verlag, 1999, 195–209.

[105] W. Sadiq, M.E. Orlowska, *On Correctness Issues in Conceptual Modeling of Workflows*, Proceedings of the 5th International Conference on information Systems (ECIS'97), Cork, Ireland: IEEE Computer Society Press, 1997, 19–21.

[106] W. Sadiq, M.E. Orlowska, *Analyzing Process Models Using Graph Reduction Techniques*, Information Systems, vol. 25, no. 2, 2000, 117–134.

[107] S. Sadiq, M. Orlowska, Sadiq, C. Foulger, *Data Flow and Validation in Workflow Modelling*, Proceedings of the 15th Australasian Database Conference (ADC 2004), Dunedin, New Zeeland, Australian Computer Society, K.-D. Schewe and H.E. Williams, vol. 27, 2004, 207–214.

[108] N. Sidorova, C. Stahl, *Soundness for Resource-Constrained Workflow Nets is Decidable* IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans, to appear.

[109] P. Starke, *Some Properties of Timed Nets Under the Earliest Firing Rule*, in Advances in Petri Nets 1989, vol. 424, New York: Springer-Verlag, 1989, 418–432.

[110] F.L. Ţiplea, T. Jucan, C. Masalagiu, *Conditional Petri Net Languages*, Journal of Information Processing and Cybernetic EIK, 27, 1991, 55–66.

[111] F.L. Ţiplea, *Selective Petri Net Languages*, International Journal of Computer Mathematics, 43, 1992, 61–80.

[112] F.L. Ţiplea, *On Conditional Grammars and Conditional Petri Nets*, Mathematical Aspects of Natural and Formal Languages, World Scientific, Singapore, 1994, 431–455.

[113] F.L. Ţiplea, T. Jucan, *Jumping Petri nets*, Foundation of Computing and Decision Sciences 19, no. 4, 1994, 319–322.

[114] F.L. Ţiplea, *Computational Power of Jumping Petri Nets*, Proceedings of the Workshop on Semigroups, Formal Languages and Computer Systems, RIMS Kokyuroku 960, Kyoto (Japan), 1996, 165–177.

[115] F.L. Ţiplea, D.C. Marinescu, *Structural Soundness of Workflow Nets is Decidable*, Information Processing Letters 96, 2005, 54–58.

[116] F.L. Ţiplea, A. Ţiplea, *Instantiating Nets and Their Applications to Workflow Nets*, Proc. of the 7th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, Romania, September 25-29, 2005, IEEE Computer Society Press.

[117] F. L. Ţiplea, G. I. Macovei, *Timed Workflow Nets*, Proceedings of the 7th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, Timişoara, Romania, IEEE Computer Society Press, September 2005, 361–366.

[118] F. L. Ţiplea, G. I. Macovei, *E-Timed Workflow Nets*, Proceedings of the 8th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, Timişoara, Romania, IEEE Computer Society Press, September 2006, 423–429.

[119] F. L. Ţiplea, G. I. Macovei, *Soundness for S- and A-Timed Workflow Nets is Undecidable*, IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans, vol. 39(4), 2009, 924–932.

[120] F.L. Ţiplea, C. Bocăneală, *Decidability Results for Soundness Criteria of Resource-Constrained Workflow Nets*, IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans, vol. 42(1), 2012, 238–249.

[121] F.L. Ţiplea, C. Bocăneală, *Priority Workflow Nets*, IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 43(2), 2013, 402–415.

[122] F.L. Ţiplea, C. Bocăneală, *Resource Relocation in Workflow Nets with Time, Resource, and Task Priority Constraints*, IEEE Transactions on Systems, Man, and Cybernetics: Systems, to appear.

[123] F. L. Ţiplea, A. Ţiplea, *Petri Net Reactive Modules*, Theoretical Computer Science, vol. 359(1), August 2006, 77–100.

[124] E. Vicario, *Static Analysis and Dynamic Steering of Time-Dependent Systems*, IEEE Transactions on Software Engineering, vol. 27(1), August 2001, 728–748.

[125] W. Vogler, *Timed Testing of Concurrent Systems*, Proceedings of 20th International Colloquium, ICALP 93 Lund, Sweden, July 59, 1993, Lecture Notes in Computer Science, vol. 700, 1993, 532–543.

[126] R. Valk, *On the Computational Power of Extended Petri Nets*, Mathematical Foundations of Computer Science, Lecture Notes in computer Science, 64, 1978, 526–535.

[127] J. Wang, *Timed Petri Nets: Theory and Application*, in The International Series on Discrete Event Dynamic Systems, Norwell, MA: Kluwer, 1998.

[128] H. Wang, Q. Zeng, *Modeling and Analysis for Workflow Constrained by Resources and Nondeterministic Time: An Approach Based Petri Nets*, IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans, vol. 38(4), 2008, 802–817.

[129] H. C. Yen, *A Unified Approach for Deciding the Existence of Certain Petri Net Paths*, Information and Computation, vol. 96, 1992, 119–137.

[130] H. C. Yen, *Priority Conflict-Free Petri Nets*, Acta Informatica, vol. 35, 1998, 673–688.

[131] M. C. Zhou, N. Q. Wu, *System Modeling and Control with Resource-Oriented Petri Nets*, New York: CRC Press, 2009.

[132] WFMC, "Workflow Management Coalition Terminology and Glossary WFMC-TC-1011," The Workflow Management Coalition, Brussels, Tech. Rep., 1999.