HELSINKI UNIVERSITY OF TECHNOLOGY
DIGITAL SYSTEMS LABORATORY
Series **A:** Research Reports ISSN 0783–5396
No. **44**; March 1997 ISBN 951-22-3495-5

APPLYING REDUCTION RULES TO ALGEBRAIC PETRI NETS

Karsten Schmidt
Digital Systems Laboratory
Department of Computer Science
Helsinki University of Technology
Otaniemi, FINLAND

Helsinki University of Technology
Department of Computer Science
Digital Systems Laboratory
Otaniemi, Otakaari 1
SF–02150 ESPOO, FINLAND

HELSINKI UNIVERSITY OF TECHNOLOGY
DIGITAL SYSTEMS LABORATORY
Series **A:** Research Reports
No. **44**; March 1997

ISSN 0783–5396
ISBN 951-22-3495-5

Applying Reduction Rules to Algebraic Petri nets

KARSTEN SCHMIDT †
Digital Systems Laboratory
Department of Computer Science
Helsinki University of Technology
Otaniemi, Finland

Abstract: We study the specific problems concerning the application of reduction rules to algebraic Petri nets. In particular we express the application conditions for the rules by means of term operations and discuss the validity of reduction rules in different interpretations of an algebraic net. Thereby the discussion is rather informal, that is we do not add formal proofs of their correctness. Nevertheless we demonstrate the principle feasibility of the net reduction approach for algebraic nets.

Keywords: petri nets, structural analysis, net reductions.

†supported by a scholarship of German Academic Exchange Service(DAAD)

Printing: TKK Monistamo; Otaniemi 1997

Helsinki University of Technology
Department of Computer Science
Digital Systems Laboratory
Otaniemi, Otakaari 1
SF–02150 ESPOO, FINLAND

Phone: $\frac{90}{+358\text{–}0}$ 4511
Telex: 123 889 hutel sf
E-mail: lab@hutds.hut.fi

1 Introduction

The major application field of Petri net reduction is pre–processing for computer–aided analysis. Reduction rules can be applied before starting algorithms to decide standard properties of Petri nets, i.e. liveness, boundedness and others [Ber83], or they can be used to reduce parts of a Petri net which are not covered by a temporal logical formula before performing model–checking. In all cases the goal of the application is to reduce the complexity of the succeeding algorithms. Net reduction as a stand–alone tool for deciding system properties seems to be restricted to specific net classes, i.e. free–choice nets [DE95], and even there it cannot compete with other methods (discussed in the same book).

For the analysis of high–level nets, say with large or infinite color domains, any reduction of the usually huge complexity of computer aided algorithms is welcome. However, the major previous works on this area are not directly suitable for algebraic Petri nets, since they are made up upon the framework of colored Petri nets [Had89], including semantic terms like "orthonormal arc function" which are difficult to decide for pure syntax based arc inscriptions, or upon predicate/transition nets [KR89], where a different distribution of inscriptions between arcs and transitions is used. Therefore we decided to make a new list of reduction rules, basically containing the known rules, but including application conditions in the language of algebraic Petri nets. Where necessary, we modified the rules themselves. In this case we add the correctness proof which otherwise is left to the original papers.

In addition to previous works we discuss the aspects concerned with the polymorphism of algebraic nets. We study, which additional conditions assure that the application of a reduction rule is sound also for non–standard interpretations.

Instead of a formal correctness proof for our translated rules we put a short discussion for motivation. The major idea thereby is to characterize the high–level rule as a simultaneous application of the low–level rule. This approach bears some danger: First, the correctness proofs of the low level rules are made up upon the framework of (finite!) place transition nets, though the argument of finiteness seems to be never used; Second, the correctness of the low level rule provides evidence for the correctness of finitely many simultaneous applications, but not necessarily for infinitely many. Thus, for a high–level net with infinite data domains a new correctness proof would be necessary.

Providing these rather technical proofs would however lead far beyond the objective of this paper: *To show the principle feasibility of a net reduction approach for algebraic Petri nets within or without the framework of polymor-*

phism.

2 Basic Definitions

First we recall some basic concepts from the theory of abstract data types. For details, please refer to [EM85].

Definition 1 (Specifications) *A* **signature** $\Sigma = [S, \Omega]$ *consists of a set* S *of* **sorts** *and a family* $\Omega = \{\Omega_{w,s}\}_{w \in S^*, s \in S}$ *of* **operation symbols**. *For* e *being the empty word,* $\Omega_{e,s}$ *is the set of* **constant symbols** *of sort* s. *A set of* Σ–**variables** *is a family* $X = \{X_s\}_{s \in S}$ *of* **variables**. *The set* $T_{\Omega,s}(X)$ *of* (Ω, X)–**terms of sort** s *is inductively defined by 1.* $X_s \cup \Omega_{e,s} \subseteq T_{\Omega,s}(X)$ *and 2. for* $\omega \in \Omega_{s_1 \cdots s_n, s}$ *and* $T_i \in T_{\Omega,s_i}(X)$, $\omega(T_1, \cdots, T_n) \in T_{\Omega,s}(X)$. *The set* $T_{\Omega,s} := T_{\Omega,s}(\emptyset)$ *contains the* **ground terms of sort** s, $T_\Omega(X) := \bigcup_{s \in S} T_{\Omega,s}(X)$ *is the set of* Σ–**terms over** X, *and* $T_\Omega := T_\Omega(\emptyset)$ *is the set of* Σ–**ground terms**. *A* Σ–**equation** *of sort* s *over* X *is a pair* $[L, R]$ *of terms* $L, R \in T_{\Omega,s}(X)$. *A* **specification** $D = [\Sigma, E]$ *consists of a signature* Σ *and a set* E *of* Σ*–equations.*

In the sequel we consider only specifications where for every $s \in S$, $T_{\Omega,s} \neq \emptyset$.

Definition 2 (Algebras) *A* Σ–**algebra** $A = [S_A, \Omega_A]$ *consists of a family* $S_A = \{s_A\}_{s \in S}$ *of* **domains** *and a set* $\Omega_A = \{\omega_A \mid \omega \in \Omega\}$ *of* **operations**, *where* $\omega_A : s_{1A} \times \cdots \times s_{nA} \to s_A$ *for* $\omega \in \Omega_{s_1 \cdots s_n, s}$. *The elements* ω_A *for* $\omega \in \Omega_{e,s}$ *can be identified with elements of* s_A. *An* **assignment** *is a family* $\alpha = \{\alpha_s\}_{s \in S}$ *of mappings* $\alpha_s : X_s \to s_A$. *An* **evaluation** *according to an assignment* α *is a family of mappings* $\{\alpha_s^\#\}_{s \in S}$ *with* $\alpha_s^\# : T_{\Omega,s}(X) \to s_A$ *which is defined inductively by 1.* $\alpha_s^\#(x) := \alpha_s(x)$ *for* $x \in X_s$, *and 2.* $\alpha_s^\#(\omega(T_1, \cdots, T_n)) := \omega_A(\alpha_{s_1}^\#(T_1), \cdots, \alpha_{s_n}^\#(T_n))$ *for* $\omega \in \Omega_{s_1 \cdots s_n, s}$. *For ground terms* $T \in T_{\Omega,s}$ *we define the* **value** *of* T *in* A $\#_A(T) := \alpha_s^\#(T)$ *for an arbitrary assignment* α *(the value is actually not dependent on* α*, since ground terms do not contain variables). An element* a *of an algebra* A *is called* **term element** *iff there is a ground term* $T \in T_{\Omega,s}$ *such that* $a = \#_A(T)$. *An algebra is called* **term algebra** *iff it consists only of term elements. A* Σ*–equation* $[L, R]$ *is* **valid** *in a* Σ*–algebra* A *iff for all assignments* α, $\alpha^\#(L) = \alpha^\#(R)$. *For a specification* $D = [\Sigma, E]$ *the* Σ*–algebra* A *is a* D–**algebra** *(or a* **model** *of* D*) iff all the equations in* E *are valid in* A. *A set of models of a specification is called a* **theory**. *The* **general** *theory consists of all models of a specification, the* **inductive** *theory of all models which are term algebras. A* **morphism** *from a* Σ*–algebra* A *to a* Σ*–algebra* B *is a family* $h = \{h_s \mid s \in S\}$ *of mappings* $h_s : s_A \longrightarrow s_B$ *satisfying* $h_s(\omega_A(x_1, \ldots, x_n)) = \omega_B(h_{s_1}(x_1), \ldots, h_{s_n}(x_n))$ *for all operation symbols* ω

contained in some $\Omega_{s_1 \ldots s_n, s}$ *and all* $x_1 \in s_{1A}, \ldots, x_n \in s_{nA}$. *For an assignment* α, $h(\alpha)$ *is defined by* $h(\alpha)(c) = h(\alpha(c))$.

Definition 3 (Substitutions) *Let* X *and* Y *be two sets of* Σ*–variables. A* **substitution** *on* X *is an assignment* $\sigma : X \to T_\Omega(Y)$, $(X_s \to T_{\Omega,s}(Y))$. *A* **ground substitution** *is a substitution* $\sigma : X \to T_\Omega$. *For a substitution its domain* X *is denoted by* $\mathrm{dom}(\sigma)$ *and the smallest set* Y *such that* σ *maps to* $T_\Omega(Y)$ *is denoted by* $\mathrm{var}(\sigma)$. *An injective substitution* $\sigma : X \to Y$ *is called* **renaming**. *For a term* T *and a substitution* σ *the term* $\sigma(T)$ *(often written as* $T\sigma$*) results from simultaneously replacing the variables in* T *by their corresponding* σ*–values. For a set of* Σ*–variables* $\mathrm{Subst}(X)$ *denotes the set of all substitutions and* $\mathrm{GrSubst}(X)$ *the set of all ground substitutions on* X.

Definition 4 (Term Equivalence) *Two terms* T_1 *and* T_2 *are* **equivalent** *according to a theory* $(T_1 \equiv T_2)$ *iff for all algebras* A *contained in this theory and all assignments* α *in* A, $\alpha_A^\#(T_1) = \alpha_A^\#(T_2)$. *The equivalence according to the general theory is denoted by* $\equiv_E$, *the equivalence according to the inductive theory is denoted by* $\equiv_I$.

For all theories, $\equiv$ is an equivalence relation on $T_\Omega(X)$. It is actually a congruence relation, i.e. $T_1 \equiv T_2$ implies $T_1\sigma \equiv T_2\sigma$ for arbitrary substitutions σ. With $[T]_\equiv$ we denote the equivalence class of the term T according to the relation $\equiv$. We write $[T]_{E/I}$ instead of $[T]_{\equiv_{E/I}}$. $\equiv$ can be straightforward extended to substitutions $\sigma_1, \sigma_2 : X \to Y$: $\sigma_1 \equiv \sigma_2$ iff for all $x \in X$ holds $\sigma_1(x) \equiv \sigma_2(x)$.

In the sequel we consider mainly the general and inductive theories. Both theories correspond to a specific set of tools. The main tool in the framework of the general theory is term rewriting. By rewriting terms according to the specified equations new all–quantified equations can be derived. Additionally equations with "unknowns" can be solved. This approach is called unification.

Definition 5 (Unification) *A substitution* σ_1 *is* **more general** *than a substitution* σ_2 *iff there is a substitution* τ *such that* $\sigma_2 \equiv_E \sigma_1\tau$. *For a set* $\mathcal{M}$ *of substitutions the set* $\mathrm{Ground}(\mathcal{M})$ *denotes the set of all ground substitutions for which a more general one is included in* $\mathcal{M}$. *A* **unification problem** *consists of a finite set of pairs of terms, written as* $\{T_1 \equiv_E^? T_1', \ldots, T_n \equiv_E^? T_n'\}$. *A substitution* σ *is a* **unifier** *of a unification problem* $\mathcal{U}$ *iff for all* $T \equiv_E^? T'$ *in* $\mathcal{U}$ *it holds* $T\sigma \equiv_E T'\sigma$. *A set* $\mathcal{S}$ *of unifiers of a set* $\mathcal{T}$ *of terms is* **complete** *iff for every unifier of* $\mathcal{T}$ *there is a more general one included in* $\mathcal{S}$.

It is not for all specifications decidable whether unifiers exist for a given problem. Nevertheless there are specifications where this problem is decidable and where for an arbitrary set of terms a finite complete set of unifiers can be computed (a good survey on this topic is [JK91]). In the sequel we focus on such "well behaving" specifications.

In the inductive theory there exist more advanced techniques to proof theorems. Examples are: induction on the term structure and case considerations on the top operation symbol of a term. Since in the inductive theory every element of an algebra is represented by a ground term, these proof techniques cover all elements of any term algebra, while it is not guaranteed that they cover all elements in a non–term algebra. Tool support for these proof techniques already exists ([RM86, Pau86, Wol92]) and is under further development.

Definition 6 (Special Algebras) *Let $D = [\Sigma, E]$ be a specification. The* **standard algebra** *St of D consists of the domains $s_{St} := \{[T]_E \mid T \in T_{\Omega,s}\}$ and the operations ω_{St} with $\omega_{St}([T_1]_E, \cdots, [T_n]_E) := [\omega(T_1, \cdots, T_n)]_E$. The* **skeleton algebra** *Sk consists of the domains $s_{Sk} := \{\bullet\}$ and the operation symbols ω_{Sk} with $\omega_{Sk}(\bullet, \ldots, \bullet) := \bullet$.*

Due to the properties of the relation $\equiv_E$ the standard algebra is a model of D. Furthermore it is a term algebra and minimal (there are no equations valid except those which are implied by E). For every model A of D, there is a unique morphism from St to A which is surjective iff A is a term algebra. The skeleton algebra is always a model of D, too. For every algebra A there is a unique morphism from A to Sk. If for every sort s it holds $T_{\Omega,s} \neq \emptyset$ (we have already restricted ourselves to this kind of specifications), then Sk is a term algebra.

Definition 7 (Multisets) *For a set M, a* **multiset** *over M is a mapping from M into the integer numbers. A multiset is* **semipositive** *iff all the values are ≥ 0. A multiset is* **finite** *iff it has finite support. The* **empty** *multiset over M, denoted by ϑ_M, assigns 0 to every element of M. For an element $m \in M$, the multiset $\underline{m}$ assigns 1 to m and 0 to every other $m' \in M$. The multisets $\mu_1 + \mu_2$ and $\mu_1 - \mu_2$ are defined by $(\mu_1 + \mu_2)(m) := \mu_1(m) + \mu_2(m)$ and $(\mu_1 - \mu_2)(m) := \mu_1(m) - \mu_2(m)$. This way every finite multiset can be written as a* **formal sum** *of the $\underline{m}(m \in M)$. We usually write m instead of $\underline{m}$. A multiset μ_1 is* **less or equal** *to μ_2 iff for all $m \in M$, $\mu_1(m) \leq \mu_2(m)$.*

Note, that we do without scalar multiplication of multisets. This will simplify some future considerations. For multisets of terms we have to define an alternative comparison relation, which is closer related to the relation $\equiv$.

Definition 8 (Multiterms) *A* **multiterm** *is a multiset over a set of terms. A multiterm μ_1 represented by the formal sum $\mu_1 = T_1 + \cdots + T_m$ is* **less or equal** *to $\mu_2 = T'_1 + \cdots + T'_n$* **with respect to the equivalence** *$\equiv$ ($\mu_1 \leq_\equiv$*

μ_2) iff $[T_1]_\equiv + \cdots + [T_m]_\equiv \leq [T'_1]_\equiv + \cdots + [T'_n]_\equiv$. For a finite multiterm μ and a term T we write $T \in \mu$ iff $\sum_{T' \equiv T} \mu(T') \neq 0$. For a finite multiterm $\mu = T_1 + \cdots + T_n$ define its interpretation with respect to an assignment α by $\alpha(\mu) = \sum_{i=1}^{n} \alpha(T_i)$.

For the evaluation of multiterms the following relation to morphisms holds: $(h(\alpha)(\mu))(h(c)) = \alpha(\mu)(c)$. The interpretation of a multiterm is a multiset in the underlying algebra.

Definition 9 (Colored Petri net) *A tuple $CN = [P, T, F, \Psi, \Lambda, m_0]$ is a* colored net *iff 1.) $[P, T, F]$ is a* net, *i.e. P and T are finite and disjoint sets called* **places** *and* **transitions**, *respectively, and F is a relation $F \subseteq (P \times T) \cup (T \times P)$, the elements of which are called* **arcs**; *2.) Ψ assigns a countable set of* **colors** *to every element of $P \cup T$; 3.) Λ assigns to every $[x, y] \in F$ a mapping which in turn assigns a finite semipositive multiset over $\Psi(p)$ to every color in $\Psi(t)$ where p is the place of x and y and t is the transition of x and y; 4.) m_0 is a* **marking**, *that is a mapping which assigns a finite semipositive multiset over $\Psi(p)$ to every $p \in P$. m_0 is called the* **initial marking**.

Definition 10 (Algebraic Petri Nets) *A tuple $AN = [D; P, T, F; \psi, \xi, \lambda; m_0]$ is an* **algebraic Petri net** *iff 1.) $D = [\Sigma, E]$ is a specification with $\Sigma = [S, \Omega]$; 2.) $[P, T, F]$ is a* **net**; *3.) ψ is a* **sort assignment** *$\psi : P \to S$; 4.) ξ assigns a set of Σ–variables $\xi(t)$ to each transition $t \in T$; 5.) λ is the* **arc inscription** *such that for $f = [p, t]$ or $f = [t, p]$ in F, $\lambda(f)$ is a multiterm over $T_{\Omega, \psi(p)}(\xi(t))$; 6.) m_0 is a* **marking**, *i.e. it assigns a finite multiterm over $T_{\Omega, \psi(p)}$ to every $p \in P$. m_0 is called the* **initial marking**.

For $f \notin F$ we define $\lambda(f) := \vartheta$. For every model A of the specification D a colored Petri net can be assigned to an algebraic Petri net. This way all the behavioral aspects of an algebraic net can be traced back to colored nets. The translation works as follows: Let $D = [S, \Omega, E]$ be a specification and A a D–algebra. For the algebraic Petri net AN the colored net $AN_A = [P, T, F, \Psi_A, \Lambda_A, m_{0A}]$ is defined by

- For $p \in P$, $\Psi(p) := \psi(p)_A$;
- For $t \in T$, $\Psi(t)$ is the set of all assignments on $\xi(t)$;
- Λ_A is defined by $\Lambda_A([x, y])(\alpha) := \alpha^{\#}(\lambda([x, y]))$;
- $m_{0A}(p) := \#_A(m_0(p))$.

A colored net in turn can be translated into a place transition net $[P', T', F', V, m'_0]$ where $[P', T', F']$ is a net, V is a mapping which assigns a positive number (multiplicity) to every $f \in F$ and m_0 is a marking, which assigns a non–negative number to every $p \in P$. The place–transition net corresponding to $CN = [P, T, F, \Psi, \Lambda, m_0]$ is constructed as follows:

- $P' := \{p.c \mid p \in P, c \in \Psi(p)\}$;
- $T' := \{t.d \mid t \in T, d \in \Psi(t)\}$;
- $[p.c, t.d] \in F'$ for $p \in P$ and $t \in T$ iff $\Lambda([p,t])(d)(c) \neq 0$, in this case $V([p.c, t.d])$ is set to $\Lambda([p,t])(d)(c)$;
- $[t.d, p.c] \in F'$ for $p \in P$ and $t \in T$ iff $\Lambda([t,p])(d)(c) \neq 0$, in this case $V([t.d, p.c])$ is set to $\Lambda([t,p])(d)(c)$;

By the translation of algebraic Petri nets to colored nets and further into place transition nets all behavioral aspects can be traced back to the definitions which are well known for place transition nets.

For further details according to algebraic Petri nets, please refer to [Rei91, Sch96c].

3 Getting started

In the remaining part of the paper we study the particular reduction rules — one per section. All these sections are organized in the same way.

First, we repeat the low level rule. We stick to the notations in [Sta90]. We present the application condition and the application of the rule. Thereby the primed notions stand for the reduced net while the unprimed ones represent the original net. Then we list the properties being kept by applying the rule. This list contains information about the boundedness or liveness of removed or transformed nodes, relations between the original and the removed net concerning liveness, boundedness, and model–checking. The statement about model–checking refers to a branching time logic which can be interpreted upon the interleaving semantics of the net. We refer to a logic consisting of atomic propositions which can be evaluated for every marking, boolean connectors, and a temporal operator $\diamond$, where a marking m satisfies a formula $\diamond\Phi$ iff there is a marking reachable from m which satisfies Φ. To the author's knowledge, the relations between net reduction and model–checking outside the reduction area, have not been proved yet (unless for the rules presented in [KR89]), but the invisibility of reduction rules outside the reduction area appears as a standard argument in the soundness proofs concerning the global properties (such as liveness and boundedness). Therefore we take these statements as apparant. For those rules where we generalize the application area, we add the soundness proofs. The part concerning the low level rule is closed by reporting the benefits that can be obtained by applying the rule. Thereby we express the benefits by the time and space costs of generating a reachability graph of the net, though that might not be the only algorithm where benefits can be obtained.

In the second part we discuss the translation of the low level rule into the language of high level nets. The main purpose is to characterize a simultaneous application of the low level rule by properties of the high–level net inscriptions. Thereby we try to trace back the verification of the application conditions to the calculi for the involved theories (equational calculus, i.e. term rewriting and unification for the general theory, term structural techniques for the inductive theory). Once having formulated an application condition in these frameworks, we rely on the tools being developed in the theorem proving community to verify them. Of course they provide additional restrictions and problems, but we leave those out of the scope of this paper. This way, with a further development of theorem provers the applicability of our rules increases automatically. As a specific feature of algebraic nets is their polymorphy (the ability of being interpreted according to different models of the specification), we attempt to derive application conditions, which are not only sound for the standard algebra, but also for all interpretations of the inductive or even the general theory. Of course usually stronger conditions must be applied for larger theories. Thus we obtain a scaled application condition according to the chosen set of interpretations for which the application shall be sound. With the general and inductive theory, plus the standard algebra as the most important stand–alone algebra, we hope to cover the most important environments concerning the interpretation of algebraic nets. Note, that for the discussion on high–level nets, node and arc sets refer to the high level net and not to its unfolding.

In the third and last part of every section we sum up formally the application condition and the application of the high–level rule. Thereby we mark a condition with ALL if it is only to be applied when we want the reduction to be sound for all algebras in the general theory, $TERM$, if we want it to be sound for all algebras in the inductive theory, and $STANDARD$ if soundness is only required for the standard algebra. Unmarked conditions are required independently of the chosen soundness area. We repeat our warning that, how much evidence the discussion on the application condition might provide, the derived rules lack a formal correctness proof. Even if their characterization as simultaneous applications of the corresponding low level rules is taken as apparant, this assures correctness only for nets with finite data domains, since otherwise the underlying net is infinite and the rules must be applied infinitely often on the low level to implement the high level rule. Nevertheless there remains a lot of evidence for the correctness of the derived high level rules.

4 Rule 1: Transitions without input places

Transitions without input places can fire whenever they want. Their output places can be arbitrarily marked. Therefore these places are no real obstacle for enabling their post–transitions. Thus, transitions without input places can be removed together with their output places.

4.1 Low level rule

Application condition

There is a transition t such that $Ft = \emptyset$.

Application

Remove t, tF and all arcs which are connected to removed nodes.

Properties

- Every removed transition is live in N;
- Every removed place is unbounded in N;
- Every formula which concerns only places in N' holds in N' if and only if it holds in N;
- N is live if and only if N' is live; if N' is empty then N is live and unbounded.

Benefits

Applying this rule it might happen that an unbounded net is replaced by a bounded one. In this case the reduction makes a lot of analysis methods applicable.

4.2 Discussion on application to algebraic nets

Transitions without input places can be easily detected. Since according to the syntax of arc inscriptions every firing mode of a transition has the same number of input places as any other mode of the same transition, the application condition fits to every mode of a transition which does not have input places at the high level.

The major problem for the application of this rule is the removal of the output places of a transition which does not have input places. The output of a transition does not necessarily cover all the colors of a place. If it does not, we are not allowed to remove this place color since that would not be covered by the low–level rule. In fact, removing a place color without an always enabled input transition can change dramatically the net's behavior (see Figure 1).

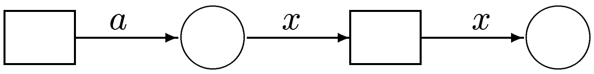

Figure 1: Non–applicability of rule 1

Let in the figure a be a constant symbol and x a variable. The place on the right hand side of this figure has an unbounded color a, while all remaining colors of this place are bounded. After removing the transition without input places on the left and its output place all other colors would become unbounded as well. Thus the removal of the transition had changed the behavior of the net outside the application area.

Unfortunately it is not possible to remove only some colors of an algebraic net's place. We can either remove the whole place or leave it untouched at all. The situation would be slightly different if we allowed sub–sorting, but then the whole approach would become more involved. Therefore we restrict the application of the rule to those cases where the output of a transition without input places covers all colors of its output places. As a slight generalization of this rule we could allow the application of the rule if for every output place of a transition without input places and for every color of this place there is a transition without input places which produces tokens on this place. In this case the simultaneous removal of all transitions without pre–places and their output places would be justified by the low–level rule.

Let in Figure 2 again a be a constant symbol and x a variable. No removal of a single transition is justified by the low level rule (it resembles the example depicted in Figure 1). However the simultaneous removal of both transitions at the left end of the net and its output places is a sound simultaneous application of the low level rule. Therefore the outline of the application of this rule is:

Application condition: The set T_0 of all transitions without input places is

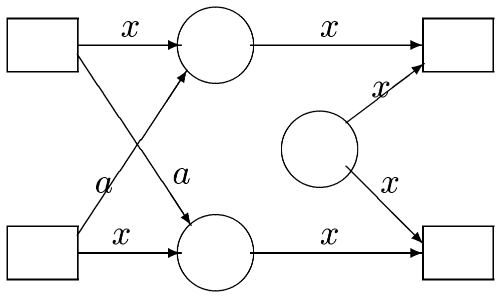

Figure 2: Simultaneous applicability of rule 1

non–empty. Let $P_0 = T_0F$. For every $p \in P_0$ and every color c of p there is a transition $t \in T_0$ and a firing mode β of t such that the firing of $[t, \beta]$ produces c–colored tokens on p.

Application: Remove T_0, P_0 and all arcs connected to them.

The remaining problem is the verification of the second part of the application condition. Actually the truth value of this condition can differ according to the chosen interpretation of the net. Consider for example again figure 1. Assumed there are other ground terms than a and no equations specified, then the application condition is not satisfied according to the standard algebra. However it is satisfied according to the skeleton algebra. If there is no other ground term than a, then the application condition is satisfied according to both the standard and the skeleton algebra, but not to algebras which are not term algebras. Assume finally that a is not a constant symbol but a variable. In this case the application condition is obviously satisfied independently of the interpretation. Thus the verification method itself depends on the range of soundness (the theory for which we want the application to be sound).

Consider the set $\mathcal{T}$ of terms which appear at arcs pointing from a transition $t \in T_0$ to a place $p \in P_0$. Given an algebra A, the set of all colors in which the transitions in T_0 put tokens on p is $\{\alpha^{\#}(T) \mid T \in \mathcal{T}, \alpha \text{ assignment in } \mathcal{A}\}$.

If $\mathcal{T}$ contains a term that consists only of a variable, then the case is clear. The transition where this variable belongs to, can put tokens of arbitrary colors on p — in arbitrary algebras. If $\mathcal{T}$ does not contain stand alone variables, then every term value is subject of function applications (by ω_A, if ω is the root symbol of the term). In most cases it is possible to interpret operation symbols non–surjective. Even a set of operation symbols can usually be interpreted in a way that the values do not cover all elements of a certain sort. A situation where we can assure that the interpretation of an operation symbol is surjective in every algebra is for instance the presence of an equation $x = y$ for a sort. In this case every interpretation assigns a singleton set to this sort, making every interpretation of an operation symbol which maps to this sort surjective. Probably these situations are quite rare and we can restrict ourselves to the easily verifiable criterion worked out above.

If we are interested only in results for term algebras, we have some more tools for verifying whether $\{\alpha^{\#}(T) \mid T \in \mathcal{T}, \alpha \text{ assignment }\}$ covers all elements of $\psi(p)$. Return for example once more to Figure 1. If a is the only operation symbol of its sort, then (since every element of the sort is represented by a ground term) $\#_A(a)$ is the only element of its sort (in every term algebra) and we can evaluate the application condition of our reduction rule with true. The behavior of the net is kept by the application of this rule for every term algebra interpretation, while it can be changed for other algebras.

Completeness (verifying that a certain set of terms represents all elements of a sort) is actually one of the major applications of term structural methods (see [Wol92]). Therefore for our purposes we can rely on methods developed there. For our concerns it is sufficient to know that these methods yield results which are sound within the inductive theory of a specification.

Consider finally the case that we are interested in the standard algebra only. This algebra is a term algebra. Furthermore it has a unique morphism to every other model of the specification. This morphism is surjective for all term algebras since a morphism always maps values of ground terms to the corresponding values of the same ground term in the other interpretation. Thus, if $\{\alpha^{\#}(T) \mid T \in \mathcal{T}, \alpha \text{ assignment }\}$ covers the whole sort in the standard algebra, it covers the whole sort in every term algebra. This means, if the application condition is satisfied for the standard algebra, it is satisfied as well in every term algebra. This way there is no difference between checking the applicability of rule 1 in the standard algebra and checking it for arbitrary term algebras.

4.3 Application

Application condition

- A set T_0 of transitions without input places is not empty. Let P_0 be the set of all output places of transitions in T_0.
- *ALL:* For every $p \in P_0$ there is a $t \in T_0$ such that at least one single variable appears as a term in $\lambda([t,p])$.
- *TERM/STANDARD:* There is for every $p \in P_0$ a term–structural proof that the set of ground instances of the terms appearing in the $\lambda([t,p])$ $(t \in T_0)$ covers all elements of $\psi(p)$.

Application

Remove T_0, P_0 and all arcs connected to removed nodes.

Justification

All removed transitions are in every interpretation of the specification transitions without input places and therefore subject of the low–level rule outlined in the beginning of this section. If the second or third item of the application condition is true, then P_0 is just the set of places to be removed by the application of this rule — in every interpretation, if the second item is true, or in every term algebra interpretation, if the third item is true, since in those cases the set of removed places is just the set of post-places of the removed transitions (in every unfolding according to a considered algebra) and thus to be removed according to the low–level rule.

5 Rule 2: Places without input transitions

Places without input transitions can never be marked again. Once empty, all their post–transitions are dead and can be removed from the net as well as the places themselves.

5.1 Low level rule

Application condition

There is a place p such that $Fp = \emptyset$ and for all $t \in pF$ it holds $m_0(p) < V([p, t])$ (the tokens on p do not enable any transition).

Application

Remove p, pF and all arcs which are connected to removed nodes.

Properties

- Every removed place is bounded in N;
- Every removed transition is dead in N;

- Every formula which concerns only places in N' holds in N' if and only if it holds in N;
- N is bounded if and only if N' is bounded;
- if N' is empty then N is not live but bounded.

Benefits

Once being removed, it is no longer necessary to check whether the transitions in pF are enabled or not at a new marking. However, the size of the graph remains the same.

5.2 Discussion on application to algebraic nets

Since we cannot remove only some colors of a place, the application of this rule is restricted to places where no color at all has pre–transition modes. These places are exactly those which do not have pre–transitions on the high level. The removal of post–transitions of these places is much easier than the removal of post–places in the previous section. Every transition where an arc from a place without pre–transitions is pointing to has a color of this place as input color for every firing modes. Thus in every interpretation all modes of this transition are removed by the low level rule if it is applied simultaneously to all colors of the place in question.

The only difference according to the soundness of the rule in different algebras is the condition, that the place to be removed must not have enough tokens to enable any of its post transitions. Since the skeleton algebra is both a model of the specification and a term algebra, the enabling of a transition in this interpretation violates the application condition when the general or inductive theories are considered while the disabling of a transition in this interpretation is sufficient for its disabling in any interpretation (since there is a morphism from every algebra to the skeleton one, enabling of a transition in an arbitrary algebra implies its enabling in the skeleton one). The enabling condition for t with respect to p in the skeleton algebra is just $card(m_0(p)) \geq card(\lambda([p,t])$ and thus easy to verify.

For checking the enabling condition in the standard algebra only, we have to assign injectively a token term on p to every appearance of a term in $\lambda([p,t])$ (if a term appears with multiplicity greater than one, several copies must be regarded). If there is a substitution on $\xi(t)$ such that every arc term is E–equivalent to a its corresponding marking term, then every ground instance of this substitution is an enabled firing mode of t. Vice versa every enabled firing mode of the transition can serve as such a substitution (for more details on the relation between enabled transitions in the standard algebra and unification of terms, refer to [Sch95b, Sch96c]). Thus the application condition in the case of regarding the standard algebra only is the non–unifiability of arc multiterm and marking multiterm in every combination of mapping the marking terms to the arc terms.

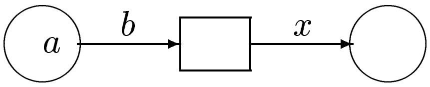

Figure 3: On the application condition for rule 2

For illustrating the difference between arbitrary or term algebras on the one hand and the standard algebra on the other, regard Figure 3. Assume, a and b are non equivalent constant symbols. The deletion of the left place and the transition is sound for the standard algebra, since the removed transition is dead. It is however not sound in the skeleton algebra, since t can fire once and mark the right place while in the reduced net this place can never be marked (due to the absence of transitions).

5.3 Application

Application condition

- There is a place p_0 without input transitions. Let T_0 be the set of all output transitions of p_0.
- *ALL/TERM:* For every $t \in T_0$ it holds $card(m_0(p_0)) < card(\lambda([p_0, t]))$.
- *STANDARD:* For all $t \in T_0$ there is no injective mapping φ from the terms appearing in $\lambda([p_0, t])$ to those in $m_0(p_0)$ (multiple terms considered as multiple copies of a term) such that the unification problem $\{T \equiv_E^? \varphi(T) \mid T \in \lambda([p_0, t])\}$ has solutions.

Application

Remove p_0, T_0 and all arcs connected to removed nodes.

Justification

If a transition is disabled according to the skeleton algebra (which is just the inequation in the application condition) then it is disabled in every algebra, particularly in every term algebra interpretation. The third item of the application condition is just the disabling condition for the standard algebra. Thus for every color of the removed place the low level application condition of the rule at the beginning of the section is satisfied. All modes of the removed transitions are post–transition of a removed color in the low level net according to arbitrary algebras. Therefore the high level rule implements exactly the simultaneous application of the low level rule for all colors of p_0.

6 Rule 3: Parallel places

When two places are connected in the same way to all transitions, then only the one with the smallest initial marking forms real restrictions for its environment, and the remaining ones can be removed without influencing the net's behavior. This rule is actually a case of an implicit place [Ber83].

6.1 Low level rule

Application condition

There are places p_1 and p_2 such that for all $t \in T$, $V([p_1, t]) = V([p_2, t])$ and $V([t, p_1]) = V([t, p_2])$. It holds $m_0(p_1) \geq m_0(p_2)$.

Application

Remove p_1 and all arcs which are connected to it.

Properties

- For every $m \in R_N(m_0)$, $m(p_1) = m(p_2) - m_0(p_2) + m_0(p_1)$;
- It holds $m \xrightarrow{t} m'$ in N' iff it holds $m \cup \{[p_1 \mapsto m(p_2) - m_0(p_2) + m_0(p_1)]\} \xrightarrow{t} m' \cup \{[p_1 \mapsto m'(p_2) - m_0(p_2) + m_0(p_1)]\}$ in N;
- Every place appearing in N' is bounded in N iff it is bounded in N';
- Every transition is live in N iff it is live in N';
- Every formula which concerns only places in N' holds in N' if and only if it holds in N;

- Every formula concerning p_1 can be translated into a formula concerning p_2 according to the first and second item;

Benefits

Applying this rule, every marking vector of the reduced net has one entry less than in the original net. Moreover, to check the enabling of post–transitions, one place less is to be considered. However, the number of nodes and arcs of a reachability graph remains the same.

6.2 Discussion on application to algebraic nets

Since we cannot remove single colors of a place, the application of this rule is restricted to cases where all colors of a place p_1 are parallel to other ones. In general it is imaginable that the parallel place color justifying the removal is different for different colors to be removed. However, we think about reduction as a kind of pre–processing. Thus we cannot allow to spend too much time for checking very sophisticated application conditions. Therefore we consider only the case that the parallel nodes remaining in the net are colors of one and the same place p_2. Furthermore, except when regarding the standard interpretation only, we do not have a good concept for representing permutations of colors. Thus we apply the further restriction that the parallel instance to $p_1.c$ is just $p_2.c$, that is the same color in p_2. Then the verification problem for the application condition is reduced to the question whether every transition mode removes or puts equally many tokens of every color to p_1 and to p_2.

Nevertheless this condition may evaluate to different truth values dependent on the involved algebra. Consider for instance the net in Figure 4.

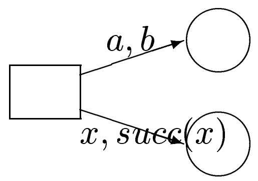

Figure 4: Nodes which are parallel in some but not all interpretations

For this net we assume that a and b are constant symbols and $succ$ is a unary operation symbol, all of the same sort. No other operation symbols are assumed. Furthermore the equations $succ(a) = b$ and $succ(b) = a$ are assumed. x is a variable. There is a model C of this specification where the considered sort is interpreted by $\{a, b, c\}$, the constant symbols are interpreted by themselves and $succ(c)$ is set to c. In this interpretation the transition, when fired in the mode $[x \mapsto c]$, puts two tokens of color c on the lower place but a token of color a and one of color b on the upper one. That is, these places are not parallel with respect to C. They are however parallel with respect to the standard algebra, since every firing mode ($[x \mapsto [a]]$ and $[x \mapsto [b]]$) puts a token of color a and one of color b on both places.

Fortunately places which are parallel in the standard algebra, are parallel in every other term algebra: Since there is a surjective morphism from the standard algebra to every term algebra and the token colors to be produced and consumed by a transition are term values, every multiset of tokens to be removed or produced on a place by a transition according to a term algebra is the morphism's image of a multiset produced or consumed in the standard algebra. Of course the images of equal multisets are equal.

These observations suggest that the methods for verifying the application condition are basically the same for term algebras and for the standard algebra alone while different methods must be applied when we want sound results for all algebras. For arbitrary algebras we have the concept of E–equivalence on multisets. Of course a transition puts in every firing mode equal multisets of tokens on p_1 and p_2 in every interpretation of the specification if and only if $\lambda([t, p_1]) \equiv_E \lambda([t, p_2])$. This condition can be checked using unification.

Accordingly we can use the relation $\equiv_I$ between multiterms to check the condition for all term algebras and for the standard algebra. The verification of this condition can be tried using again term structural methods like induction, case consideration on the top symbol and so on.

In the net in Figure 4 it holds $a + b \equiv_I succ(x) + x$. A term structural proof can be established via the lemma that all elements of the sort are represented by the terms a or b which can be proved by induction on the term structure. Then considering the two remaining cases we can prove the multiset equation.

Concerning the calculation of the maximum number of tokens on different places we chose again the easy way (possible generalizations are left to the implementor's skills). We assume that for every color there are more tokens on p_2 than on p_1. Fortunately the truth value of this condition is the same for all algebras as for all term algebras and the standard algebra alone: Since initial markings are made of ground terms, they evaluate to the unique morphism's image of the initial marking in the standard interpretation. The relation $\leq$ on multisets implies the same relation on the morphism's images. If therefore $m_0(p_1) \leq m_0(p_2)$ in the standard interpretation, this relation holds in every interpretation. Vice versa, if the relation holds in every interpretation it holds as well in every term interpretation and also in the standard interpretation which is a term algebra. Thus for the verification of the marking inequation we need only one condition independently of for which algebras we want sound results.

6.3 Application

Application condition

- There are places p_1 and p_2 such that $\psi(p_1) = \psi(p_2)$ and for all $t \in T$:
  - *ALL:* $\lambda([p_1, t]) \equiv_E \lambda([p_2, t])$ and $\lambda([t, p_1]) \equiv_E \lambda([t, p_2])$;
  - *TERM/STANDARD:* $\lambda([p_1, t]) \equiv_I \lambda([p_2, t])$ and $\lambda([t, p_1]) \equiv_I \lambda([t, p_2])$, resp.;
- $m_0(p_1) \leq_E m_0(p_2)$.

Application

Remove p_2 and all arcs connected to it.

Justification

The application condition is sufficient to assure that all colors of p_1 are parallel to the corresponding color of p_2. Thus the rule implements the simultaneous application of the low level rule for all colors of $\psi(p_1)$ $(= \psi(p_2))$.

7 Rule 4: Parallel transitions

When two transitions are connected in the same way to all places, firing of one transition can be replaced by the other one, and one of the transitions can be removed.

7.1 Low level rule

Application condition

There are transitions t_1 and t_2 such that for all $p \in P$ $V([t_1, p]) = V([t_2, p])$ and $V([p, t_1]) = V([p, t_2])$.

Application

Remove t_1 and all arcs which are connected to it.

Properties

- For every $m \in R_N(m_0)$, $R_N(m) = R_{N'}(m)$
- Every place in N' is bounded in N iff it is bounded in N';
- t_1 is live in N iff t_2 is live in N';
- Every transition in N' is live in N iff it is live in N';
- Every formula holds in N' if and only if it holds in N;

Benefits

In the reachability graph, parallel transitions appear as double arcs from a marking to another. The reduction decreases therefore the number of arcs of the graph, but not the number of nodes. However, during graph generation the enabledness check must only be performed for one transition rather than for all parallel ones.

7.2 Discussion on application to algebraic nets

In contrast to place colors it is in principle possible to remove firing modes of transitions without removing the whole transition. For this purpose it is however necessary to substitute the variables appearing around a transition, say through a substitution σ and to replace $\xi(t)$ by the set of variables $\mathrm{var}(\sigma)$. The set of firing modes of the new transition coincides with the set of all those firing modes of the old transition which are equivalent to a ground instance of σ. This is obviously a subset of the set of all firing modes. Therefore the application of σ to a transition can be considered as the removal of all firing modes which are *not* equivalent to a ground instance of σ.

Unfortunately in non–trivial cases there is no common representation of the complement of a set of terms or substitutions which is valid for all algebras or all term algebras: Since the skeleton algebra contains one element per sort, the complement of a non–empty set of terms of a sort is always empty while the representation of the complement can be non–empty as soon as there is a model of the specification which assigns a set containing more than one element to a sort. This is however the usual case. For this reason it is difficult to benefit from the possibility to remove only some firing modes but not a whole transition, at least if we want to obtain sound results for more than only the standard algebra. An application would be possible if there were a way to represent transitions *not* to be removed — say by one or a set of substitutions. However, in our case we have a straight forward notion of all firing modes which *are* to be removed: by exhibiting a corresponding firing mode of another transition. Therefore we consider it to be necessary to restrict the application of the current rule to the removal of whole transitions only.

For assuring that for every firing mode to be removed there is a parallel mode of another transition we can allow ourselves some more freedom than in the case of parallel places. Thus firing modes of a transition to be removed must be mapped to parallel firing modes of a transition to remain in the net. Since different transitions usually refer to different variable spaces, we cannot use the identity mapping for this purpose (as we did in the previous section). But we can apply the concept of substitutions. If we find a substitution σ for the variables of the remaining transition such that the substituted arc terms are equivalent to the arc terms of the transition to be removed (without substituting the terms around this one!) then we can assure that every firing mode of the removed transition is parallel to the corresponding ground instance of the σ–instance of the remaining transition.

Consider for instance the net in Figure 4, but consider every rectangle there to be a place and every circle to be a transition. Then the transition at the arc inscribed with a, b can be removed, since its firing can be replaced by firing the remaining transition in the mode $[x \longrightarrow a]$ (which serves as σ in this case). Remember that $succ(a) = b$ is a specified equation. Thus this removal is valid in every interpretation of the net.

In every term model also the removal of the other transition would be correct, since (taking the empty substitution σ it holds $a + b \equiv_I x + succ(x)$. In fact in every term algebra there are only the firing modes $[x \longrightarrow a]$ and $[x \longrightarrow b]$ which both evaluate the arc to the only existing place by $a+b$ (due to specified equations) which is exactly the inscription of the other arc. However, the latter removal is not correct for non–term models. See for instance the algebra C defined in the previous section. Removing the transition at the arc inscribed with $x + succ(x)$ we would lose in this interpretation the firing mode putting c onto the place.

While this example illustrated the difference between the application to all interpretations and to term algebras only, there is again no difference between term algebras and the standard algebra alone. As in the previous section the existence of a surjective morphism proves that the application condition being satisfied for the standard algebra, is also satisfied for any term algebra.

The remaining problem—to get the substitution σ—can be solved by using unification in the case of all algebras while we leave the problem of finding it to a tool for reasoning on the inductive theory of a specification to find one there.

7.3 Application

Application condition

There are transitions t_1 and t_2 and a substitution σ such that for all $p \in P$:

- *ALL:* $\lambda([t_1, p]) \equiv_E \lambda([t_2, p])\sigma$ and $\lambda([p, t_1]) \equiv_E \lambda([p, t_2])\sigma$;
- *TERM/STANDARD:* $\lambda([p_1, t]) \equiv_I \lambda([p_2, t])\sigma$ and $\lambda([t, p_1]) \equiv_I \lambda([t, p_2])\sigma$;

Application

Remove t_1 and all arcs connected to it.

Justification

Every firing mode β of t_1 is by application condition parallel to the corresponding firing mode $\sigma\beta$ and can thus be removed according to the low level rule.

8 Rule 5: Equivalent places

This rule covers a situation where it is actually not necessary to distinguish between two places, since all transitions which depend on these places have the same effect on the remaining net. In this case the two places can be merged and tokens produced on either of these places are collected on the merged place. A specific item of the application condition assures that there cannot be additional enabled transitions. (Consider for instance two places bearing one token each and a post–transition consuming two tokens from either of this place. This transition is not enabled in the original net but it would be enabled when the merged place has two tokens.)

8.1 Low level rule

Application condition

There are two places p_1 and p_2 and a natural number k such that $p_1F \cap p_2F = \emptyset$ and the following conditions hold:

1. For every $t \in p_1F$ there is a $t' \in p_2F$ and for every $t' \in p_2F$ there is a $t \in p_1F$ such that
   - for all $p \in P$, $V([t,p]) = V([t',p])$;
   - for all $p \in P \setminus \{p_1, p_2\}$, $V([p,t]) = V([p,t'])$;
   - $V([p_1,t]) = V([p_2,t']) = k$;
2. For every $f \in F \cap (T \times \{p_1,p_2\})$, $k \mid V(f)$;
3. $k \mid m_0(p_1)$, $k \mid m_0(p_2)$.

Application

Remove p_1, p_1F and all arcs in $F \cap (\{p_1\} \times T)$. Replace every arc in $F \cap (T \times \{p_1\})$ by an arc from the same transition to p_2 having the same multiplicity as the original one. If this arc already exists, sum up the multiplicities. Let $m'_0(p_2) = m_0(p_1) + m_0(p_2)$.

Properties

- p_1 or p_2 is unbounded in N iff p_2 is unbounded in N';
- If a post–transition of p_2 is live in N then it is live in N';
- If a post–transition of p_1 is live in N then the corresponding post–transition of p_2 (according to item 1 of the application condition) is live in N';
- Every formula which does not concern p_1 nor p_2, holds in N iff it holds in N'.

Proof

For this rule is a generalized version of the one found in [Sta90], we have to prove the soundness of this rule. However, the stated properties of the rule follow straightforward from the two lemmas given below. These lemmas as well as their proofs make large use of replacing a post–transition of p_1 by one of p_2 and vice versa, namely by the one which resembles the replaced one according to the first item of the application condition. In order not to be too formal, we refer to this transition as to the *counterpart* of the original one, if there is more than one possible counterpart for a transition, we mean any of them (mathematically we would need to define mappings between p_1F and p_2F and vice versa).

Lemma 1 *For every sequence of transitions w and marking m with $m_0 \xrightarrow{w} m$ in N there are a sequence of transitions w' and a marking m' in N' such that $m'_0 \xrightarrow{w'} m'$ and the following properties hold:*

1. *w and w' have the same length and are equal in all positions except those where a transition t in p_1F appears in w; at these positions a counterpart t' of t in p_2F appears in w';*
2. *For all $p \in P \setminus \{p_1, p_2\}$, $m(p) = m'(p)$;*
3. *$m'(p_2) = m(p_1) + m(p_2)$.*

Lemma 2 *For every sequence of transitions w' and marking m' with $m'_0 \xrightarrow{w'} m'$ in N' there are a sequence of transitions w and a marking m in N such that $m_0 \xrightarrow{w} m$ and the following properties hold:*

1. *w and w' have the same length and are equal in all positions except those where a transition t in p_2F appears in w'; at these positions either t itself or a counterpart t' of t in p_1F appears in w;*
2. *For all $p \in P \setminus \{p_1, p_2\}$, $m(p) = m'(p)$;*
3. *$m'(p_2) = m(p_1) + m(p_2)$.*

Proof of Lemma 1: (*by induction on w*)
For w being the empty word all items are obviously satisfied if we choose the empty word also as w'. Assume now the lemma holds for a word w. We show that it holds for all wt ($t \in T$). For this purpose let $m_0 \xrightarrow{w} m^* \xrightarrow{t} m$.

According to the induction assumption there is a w' such that $m'_0 \xrightarrow{w'} m^{*\prime}$, and w' and $m^{*\prime}$ satisfy the requirements of the lemma with respect to w and m^*. If t is not a post–transition of p_1, than it is — according to the second and third item of the lemma — enabled at $m^{*\prime}$ in N' as well and $w't$ together with the marking reached by firing $w't$ from m'_0 satisfy the conditions of the lemma. In particular the third item of the lemma holds since all arcs pointing to p_1 in N point to p_2 in N'.

If t is a post–transition of p_1 then — according to the application condition — t has a counter–part t' in p_2F which has the same effect t on all places except removing tokens from p_2 instead of p_1 (but in the same quantity). Since p_2 contains in $m^{*\prime}$ at least as many tokens as p_1 in m^*, t' is enabled at $m^{*\prime}$ in N'. Firing it leads obviously to a marking which satisfies together with $w't'$ the requirements of the lemma. □

Proof of Lemma 2: (*by induction on w'*)
Apart from the main goal of the lemma we prove, that the number of tokens on p_1 as well as the number of tokens on p_2 are for every marking reachable in N' a multiple of k. For the empty word there is nothing left to show (the last item of the application condition and the setting of m'_0 prove our additional goal).

Assume now the lemma holds for w'. We show that it holds for any $w't'$ ($t' \in T'$) such that $m'_0 \xrightarrow{w'} m^{*\prime} \xrightarrow{t'} m'$. By induction assumption there are a w and m^* which satisfy the requirements of the lemma with respect to w' and $m^{*\prime}$. In particular it holds $m_0 \xrightarrow{w} m^*$. If t' is not contained in p_2F then it is — according to the last item of the lemma — enabled at m^* in N as well and can be fired, leading to a marking which satisfies all the requirements of the lemma (if t' produces tokens on p_1 or p_2 in N, then it puts as many tokens on p_2 in N' since the arcs pointing to p_1 in N point to p_2 in N'). The additional goal is satisfied as well since the multiplicity of arcs producing tokens on p_1 or p_2 in N is a multiple of k.

If t' is contained in p_2F then there are tokens on p_2 at $m^{*\prime}$ in N'. Since $m^*(p_1) + m^*(p_2) = m^{*\prime}(p_2)$, there are tokens on p_1 or on p_2 in N. By our additional goal this number is always a multiple of k. Thus on p_1 or on p_2 there are at least k token which is according to the application condition enough to enable either t' or one of its counterparts t (by induction assumption both transitions are enabled with respect to all other pre–places; the application condition assures that no transition has both p_2 and p_2 as input place). Consequently either wt' or wt can be fired at m_0, the equation for the markings reached by firing $w't'$ and wt (or wt') is straightforward. The additional goal holds as well since only multiples of k are produced or removed from p_1 and p_2. □

Benefits

In the original net, post–transitions of p_1 and p_2 may interleave, causing a state explosion according to different distributions of tokens on p_1 and p_2. This explosion is no longer present in the reduced net. Thus the reduction leads in general to smaller graphs and, by the way, to smaller marking vectors and less transitions to be checked for enabledness.

8.2 Discussion on application to algebraic nets

As usual only whole places can be removed from N. Thus it is necessary to assure the simultaneous applicability of the rule for all colors of a pair of places. Assume there is a post–transition of one of these places which is connected to this place via an arc inscribed with more than one term (say, by at least two non–equivalent ones). Then there are firing modes of this transition (in at least one algebra) where tokens of two different colors are removed from the place. If we want to apply the rule to remove all colors of the place then at least some colors serve sometimes as the place to be removed and sometimes as a third place, not involved in the removal. This situation might be too complicated for the moment. Thus we restrict ourselves to the case that the post–transitions of the place to be removed are inscribed by one term only, possibly occuring with a multiplicity greater than one. This multiplicity takes the role of k in the low–level rule. A simple criterion to assure all application conditions related to k is of course to require, that in all inscriptions of arcs pointing to p_1 and p_2 all terms appear with a multiplicity which is a multiple of k. Accordingly the multiplicity of terms appearing in the initial marking of p_1 and p_2 shall be multiples of k. With these convention the last two items of the low level application condition are covered.

The first item however resembles the verification of the application condition of the parallel transitions rule, except that arcs from p_1 to t are related to arc from p_2 to t' and vice versa. But that does not inherit any additional complications.

8.3 Application

Application condition

There are two places p_1 and p_2 and a non–zero natural number k such that

1. $p_1 F \cap p_2 F = \emptyset$;

2. For every $t \in p_1F$ there is a $t' \in p_2F$, a substitution σ on $\xi(t')$ and a term T such that
   - *ALL:* For every $p \in P$, $\lambda([t,p]) \equiv_E \lambda([t',p])\sigma$;
   - *ALL:* For every $p \in P \setminus \{p_1, p_2\}$, $\lambda([p,t]) \equiv_E \lambda([p,t'])\sigma$;
   - *ALL:* $\lambda([p_1,t]) \equiv_E \lambda([p_2,t'])\sigma$;
   - *TERM/STANDARD:* For every $p \in P$, $\lambda([t,p]) \equiv_I \lambda([t',p])\sigma$;
   - *TERM/STANDARD:* For every $p \in P \setminus \{p_1, p_2\}$, $\lambda([p,t]) \equiv_I \lambda([p,t'])\sigma$;
   - *TERM/STANDARD:* $\lambda([p_1,t]) \equiv_I \lambda([p_2,t'])\sigma$;
   - $\lambda([p_1,t]) = k \cdot T$;
3. For every $t \in p_2F$ there is a $t' \in p_1F$, a substitution σ on $\xi(t)$ and a term T such that
   - *ALL:* For every $p \in P$, $\lambda([t,p]) \equiv_E \lambda([t',p])\sigma$;
   - *ALL:* For every $p \in P \setminus \{p_1, p_2\}$, $\lambda([p,t]) \equiv_E \lambda([p,t'])\sigma$;
   - *ALL:* $\lambda([p_2,t]) \equiv_E \lambda([p_1,t'])\sigma$;
   - *TERM/STANDARD:* For every $p \in P$, $\lambda([t,p]) \equiv_I \lambda([t',p])\sigma$;
   - *TERM/STANDARD:* For every $p \in P \setminus \{p_1, p_2\}$, $\lambda([p,t]) \equiv_I \lambda([p,t'])\sigma$;
   - *TERM/STANDARD:* $\lambda([p_2,t]) \equiv_I \lambda([p_1,t'])\sigma$;
   - $\lambda([p_2,t]) = k \cdot T$;
4. For all $t \in Fp_1 \cup Fp_2$ and for all terms T, $k \mid \lambda([t,p_1])(T)$, $k \mid \lambda([t,p_2])(T)$, $k \mid m_0(p_1)(T)$ and $k \mid m_0(p_2)(T)$;

Application

Remove p_1, p_1F and all arcs starting at p_1. Replace all arcs to p_1 by arcs to p_2, inscribed in the same way. If those arcs already exist, add the inscription of the arc to p_1 to the one of the arc to p_2. Add $m_0(p_1)$ to $m_0(p_2)$.

Justification

For this rule it might be a bit more difficult to see that the same result could be obtained by simultaneous application of the low–level rule, because we have to assure that after some applications of the rule the application condition for the remaining ones is still satisfied. But in fact the first item of the application condition of the low level rule remains satisfied since the "counterpart" relation is symmetric and we do not remove post–transitions of p_2. Therefore there remains a counterpart for other modes of post–transitions of p_1 in the net even if several post–transitions of p_1 share the same counterparts. The other places around post–transitions of p_1 and p_2 remain untouched as well as their arcs, except pre–transitions of p_1. But if two transition modes point to the same color of p_1 before applying the rule, they both point to the same color of p_2 afterwards. Thus they still satisfy the conditions established in item 1 of the low level application condition. The other items remain obviously satisfied.

9 Rule 6: Free choice decisions

When a conflict is solved free choice, then its solution can be integrated into the previous conflict solution. This way a sequence of two transitions is replaced by one single transition and two conflicts are solved at once.

9.1 Low level rule

Application condition

There is a place p which has at least one pre–transition and at least one post–transition satisfying $pF \cap Fp = \emptyset$, $(pF)F \neq \emptyset$, $F(pF) = \{p\}$, and there is a number v such that for all $t \in pF$: $V([p,t]) = v$, for all $t \in Fp$: $V([t,p]) = v$, and $m_0(p) < v$.

Application

For every pair $[t_i, t_j]$ ($t_i \in Fp$ and $t_j \in pF$), insert a new transition t_{ij}, arcs $[p', t_{ij}]$ for every $[p', t_i] \in F$ with $V([p', t_{ij}]) := V([p', t_i])$, arcs $[t_{ij}, p']$ for every p' such that $[t_i, p'] \in F$ or $[t_j, p'] \in F$ with $V([t_{ij}, p']) := V([t_i, p']) + V([t_j, p'])$. Remove p, pF, Fp and all arcs connected to removed nodes.

Properties

- t_i is live in N iff there is a j such that t_{ij} is live in N';
- p in unbounded in N iff a place in $(pF)F$ is unbounded in N';
- Every formula which does not concern p and $(pF)F$ holds in N iff it holds in N'.

The exclusion of $(pF)F$ for formulas is necessary since by replacing the sequence $t_i t_j$ by a single transition t_{ij} an intermediate state (the one with tokens on p) is lost.

Benefits

If both Fp and pF contain more than one transition, the number of transitions in the net increases by the application of this rule. However, a sequence of two transitions of the original net is replaced by a single transition in the reduced net. Thus this sequence can no longer be interleaved with other events in the net. Consequently the number of states of a reachability graph decreases.

9.2 Discussion on application to algebraic nets

If every post–transition mode of a color of p has only one pre–place, then the post–transitions of p have only p as its input place. Moreover, in all firing modes only tokens of one color are removed from p. Thus, the terms appearing at an arc starting from p must be equivalent with respect to the considered class of algebras. This necessary condition is also sufficient to assure that all conflicts involving colors of p are solved free choice. To obtain an equal multiplicity v, the sum of multiplicities of (equivalent) terms must equal v. The condition on m_0 can be easily checked involving standard and skeleton algebras. Those algebras give a minimum and a maximum number of tokens of one and the same color.

For the pre–transitions of p we may allow more than one term at their arc to p. But nevertheless every firing mode must put v tokens of every color onto p. We can assure this condition for the standard algebra when all (equivalence classes) of terms occur with multiplicity v and terms belonging to different equivalence classes are not even unifiable. In this case there is no firing mode where non equivalent terms represent the same color (this firing mode would in fact be a unifier). As later considerations shall show, we do not need to consider the application conditions for non–standard algebras.

Consider next the conditions $pF \neq \emptyset$ and $Fp \neq \emptyset$. They must be satisfied for every color to be removed. Thus the terms which appear at arcs connected to p must represent all elements of the sort $\psi(p)$. This problem has already been studied for rule 1. In the case of considering the standard algebra only, we are allowed to apply term structural proof methods to check this condition.

The major problem for the application of this rule is to combine the post–transitions with the pre–transitions of p. For this purpose we have to relate firing modes of pre–transitions to firing modes of post–transitions which generate the same color. If an input transition t_i produces a token of color c on p in a mode β and a transition t_j consumes a token of color c from p in a mode β', then there is a term T appearing in $\lambda([t_i, p])$ and a term $T' \in \lambda([p, t_j])$ such that $\beta^{\#}(T) = \beta'^{\#}(T')$ in the algebra currently under consideration. In the standard interpretation this means that $\beta \cup \beta'$ forms a unifier for T and T'. Thus, in the standard algebra the combination can be done by unification.

For the purpose of simplicity consider first the case that T and T' are the only terms appearing in the inscriptions of $[t_i, p]$ and $[p, t_j]$, respectively. We can describe the combined transition mode there by applying β to all terms appearing at arcs around t_i, β' to all arcs around t_j and then combining those instances to a new transition with an empty set of variables. But we can do this operation as well simultaneously for many firing modes, if we have a (non ground-) unifier σ of T and T'. If we build a new transition which has var(σ) as its variable set, the input arcs of t_i as input arcs (inscribed by the σ–instances of the original ones) and the output arcs of t_i and t_j (except the one to p) as its output arcs, also inscribed by the σ–instances of the original ones, then every firing mode β of this transition implements the low–level combination of the mode $\sigma\beta$ (restricted to $\xi(t_i)$) of t_i and the mode $\sigma\beta$ (restricted to $\xi(j)$) of t_j. Performing this operation for every every pair of input and output transitions and every element of a complete set of unifiers for the two involved terms we obtain finally all combinations we would obtain by the low level reduction rule (since a complete set of unifiers covers also all ground unifiers, that is all pairs of firing modes to be combined).

It may happen that some ground unifier is covered by more than one element of a complete set of unifiers. This is however no problem. In this case one and the same combination of firing modes appears several times which does not influence the behavior of the net at all since they are parallel (in the low level sense).

If there are different (non–equivalent) terms appearing in the inscription of a pre–transition of p, the case is a bit more involved though the general idea works as well. We must only pay attention to the fact that these transitions are subject of more than one applications of the reduction rule since in every firing mode they are pre–transitions of more than one color of p. Thus we have to combine every pre–transition t_i of p with $\frac{card(\lambda([t_i,p])}{v}$ post–transitions. Thereby it may occur that a firing mode of a pre–transition must be combined with two (or more) different firing modes of one and the same post–transition. For allowing multiple appearances of one and the same transition in different modes in a unification problem, we have to consider the variables belonging to one appearance as different from the variables of the other appearance (thus allowing for binding these variables to different firing modes). This distinction between variables can be obtained by renaming them with new, previously unused variables. The unification problem to be solved is given by unifying every term of the pre–transition with a term of a post–transition. Doing this in all combinations for assigning post–transitions to the terms appearing at the arc pointing to p, we finally cover all combinations of firing modes as we did in the low level case.

All considerations so far concern the standard algebra. For the purpose of discussing the problem for arbitrary and term algebras, consider the net in the upper part of figure 5.

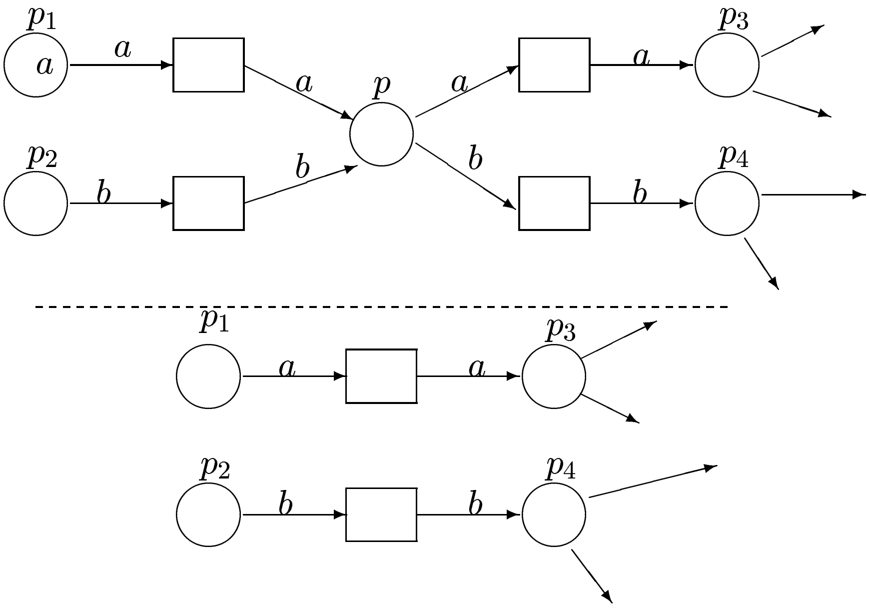

Figure 5: Application problem for different algebras

Let a and b be non–equivalent constant symbols and the only operation symbols of their sort. The net in the lower part of the figure show the result of the application as described above, which implements the application of the low level rule for the standard interpretation. This way of application is however wrong for the skeleton algebra, since there it would be possible to fire in the original net first t_1 and then t_4 which would produce tokens on p_4. However it is not possible to produce a token on p_4 in the reduced net. Thus the skeleton algebra requires different ways of combining the pre- and post–transitions of p. Actually in the case of only ground terms appearing around p the number of combinations to be build by the reduction rule depends only on which equations are valid between those ground terms. But this is exactly the property which distinguishes different term algebras. Thus there is no straight forward way to implement the reduction rule in a manner which covers in a common way all term algebras (or even worse: all arbitrary algebras). Thus we have to restrict the application of the whole rule to the standard algebra.

9.3 Application

Application condition

(Let $Ground(M)$ the set of all ground instances of terms appearing in M.)
STANDARD:
There is a place p and a number $v > 0$ such that

- $F(pF) = \{p\}$;
- $Ground(\{T \mid \exists t \in Fp : \lambda([p,t])(T) > 0\})$ covers all elements of $\psi(p)$ (to be done by term structural methods);
- $Ground(\{T \mid \exists t \in pF : \lambda([t,p])(T) > 0\})$ covers all elements of $\psi(p)$ (to be done by term structural methods);
- $(pF)F \neq \emptyset$;
- $pF \cap Fp = \emptyset$;
- for every $t \in pF$ there is a T_t such that $\lambda([p,t]) = v \cdot T_t$;
- for every $t \in Fp$ there is a number n_t and terms $T_{t1}, \ldots, T_{tn_t}$ such that $\lambda([t,p]) = v \cdot T_{t1} + \cdots + T_{tn_t}$ and T_{ti} and T_{tj} are not unifiable for $i \neq j$;
- for all ground terms T, $m_0([T]_E) < v$.

Application

For every $t \in Fp$, for all $[t_1, \ldots, t_{n_t}] \in pF^{n_t}$, let τ_i be a renaming of $\xi(t_i)$ such that all the $\mathrm{var}(\tau_i)$ are pairwise disjoint and disjoint to $\xi(t)$, and compute a complete set of unifiers $\mathcal{U}$ for the unification problem

$$\{T_1\tau_1 \equiv_E^? T_{t1}, \ldots, T_{n_t}\tau_{n_t} \equiv_E^? T_{tn_t}\}$$

For every $\sigma \in \mathcal{U}$ insert a new transition t^* with $\xi(t^*) = \mathrm{var}(\sigma)$, $\lambda([p',t^*]) = \lambda([p',t])\sigma$ for every $p' \in P \setminus \{p\}$, $\lambda([t^*,p']) = \lambda([t,p'])\sigma + \sum_{i=1}^{n_t} \lambda([t_i,p'])\tau_i\sigma$.

Remove p, pF, Fp and all arcs connected to removed nodes.

Justification

This is in fact a sound and complete implementation of a simultaneous application of the low level rule to all colors of p, since we cover all variations of combining post–transitions with pre–transitions, and all elements of a complete set of unifiers which in turn covers all ground unifiers. Ground unifiers of the constructed unification problem are exactly the matching firing modes to be combined.

10 Rule 7: Internal Conflicts

When a place is the only output place of some transitions and these transitions do not conflict other transitions then this firing can be combined with firing the post–transitions of this place. This way a sequence of two transitions is replaced by one single transition.

10.1 Low level rule

Application condition

There is a place p such that $(Fp)F = \{p\}$, $pF \neq \emptyset$, $pF \cap Fp = \emptyset$, $(F(Fp))F = Fp$, all arcs starting at or pointing to p have the same multiplicity v, every transition in Fp has an input place, and $m_0(p) < v$.

Application

For every $t_i \in Fp$ and every $t_j \in pF$, insert a new transition t_{ij} such that $V([p', t_{ij}]) = V([p', t_i]) + V([p', t_j])$ for all $p' \in P \setminus \{p\}$ and $V([t_{ij}, p']) = V([t_j, p'])$ for all $p' \in P \setminus \{p\}$.

Remove p, pF, Fp and all arcs connected to removed nodes.

Properties

- If p is unbounded in N then some elements of Fp are unbounded in N';
- N is bounded iff N' is bounded;
- N is live iff N' is live;
- Every formula which does not concern p and $F(Fp)$ holds in N iff it holds in N';
- All items except the second one are also valid when the condition, that every transition in Fp has an input place is skipped.

Proof

Since this is a generalization of the rule presented in [Ber83, Had89, Sta90] we have to present the proof of its soundness here.

The major idea of the reduction is that firing transitions contained in Fp in N can be delayed until their post–transition fires, since they to not enable other transitions than those in pF and do not disable other transitions than those which are also contained in Fp. But firing a transition in Fp immediately followed by firing a transition in pF is implemented in N' by firing the combined transition.

In the following lemma we formalize this idea. It serves as a pre–requisite for proving the properties listed above.

Lemma 3 *For every sequence w of transitions in T with $m_0 \xrightarrow{w} m$ in N there are two sequences w_1 and w_2 of transitions such that $m_0 \xrightarrow{w_1 w_2} m$ in N, $\#_{w_1 w_2}(t) = \#_w(t)$ for all $t \in T$ ($w_1 w_2$ is a permutation of w), $w_1 \mid_{T \setminus Fp} = w \mid_{T \setminus Fp}$ (all occurrences of transitions outside Fp are not permuted), $w_2 \in (Fp)^*$ and in w_1 every appearance of an element of Fp is immediately followed by an appearance of an element of pF*.*

Proof. *(by induction on w.)* For w being the empty sequence there is nothing to be shown. Let $w = w't$ for some sequence w' and a transition t. Let $m_0 \xrightarrow{w'} m' \xrightarrow{t} m$ in N. By induction assumption we obtain two sequences w'_1 and w'_2 such that $m_0 \xrightarrow{w'_1 w'_2} m'$ and the requirements of the lemma are satisfied with respect to w'. We consider three cases for t:

First case: $t \notin pF \cup Fp$. Since according to the application condition ($(Fp)F = \{p\}$) the only output place of transitions in w'_2 is p which is by case assumption not an input place of t, t is already enabled after firing w'_1 from m_0 and does not disable any transition in Fp, since by application condition ($(F(Fp))F = Fp$) and case assumption it does not conflict any transition that appears in w_2. Thus $w_1 = w'_1 t$ and $w_2 = w'_2$ satisfy all requirements of the lemma.

Second case: $t \in Fp$. Then $w_1 = w'_1$ and $w_2 = w'_2 t$ satisfy obviously the requirements of the lemma.

Third case: $t \in pF$. By application condition ($Fp \cap pF = \emptyset$ and $(F(Fp))F = Fp$), t does not have any input places which are input places of Fp. Thus firing t does not disable any transition in Fp. On the other hand, since all arcs connected to p have the same multiplicity v and $m_0(p) < v$, there cannot be

*With $\#_w(t)$ we denote the number of occurrences of t in the word w. For a word w and a set of letters Q, $w \mid_Q$ denotes the word which can be obtained by w through removing all letters which are not contained in Q.

more appearances of transitions of pF in w than appearances of transitions in Fp. Since elements of Fp and pF appear pairwise in w_1', w_2' cannot be empty. But t is already enabled after firing the first element t^* of w_2' since appearances of other transitions in w_2' put only tokens on p but not on other input places of t. Thus we can set $w_1 = w_1' t^* t$ and $w_2 = w^*$ where $w_2' = t^* w^*$ and all requirements of the lemma are satisfied. □

Obviously firing a transition $t_i \in Fp$ in N immediately followed by firing a $t_j \in pF$ has exactly the same effect as firing t_{ij} in N'. Thus, using Lemma 3 we can relate transition sequences of N to sequences of N' and (without need to use this lemma) vice versa. This way we can prove all the properties listed above.

Lemma 4 *Every formula which does not concern p and $F(Fp)$, holds in N iff it holds in N'.*

Sketch of the proof. We can divide every sequence w of transitions that can fire from m_0 in N into two sequences w_1 and w_2 according to Lemma 3. Since the difference between w and $w_1 w_2$ is restricted to Fp, this difference cannot be observed outside $\{p\} \cup F(Fp)$. Since in w_1 the pre– and post–transitions of p appear pairwise, there is a sequence w_1' of transitions fireable in N' where all these pairs have been replaced by the corresponding combined transitions. Since the state before firing the pre–transition and the one after firing it cannot be distinguished outside $\{p\} \cup F(Fp)$, all observations which can be made in N outside $\{p\} \cup F(Fp)$ can also be made in N'. The other way round, by replacing every combined transition of N' by the pair of original ones, every observation which can be made in N' outside p, can as well be made in N.

Finally the truth value of a formula depends only on the observations which can be made for its concerned places. □

Lemma 5 *If p is unbounded in N then there exists a $p' \in F(Fp)$ which is unbounded in N'.*

Proof. We show that for every k there is a place $p' \in F(Fp)$ and a reachable marking m' in N' such that $m'(p') > k$.

Since p in unbounded in N, there is a sequence w such that $m_0 \xrightarrow{w} m$ in N and $m(p) > k \cdot v \cdot card(Fp)$. Thus, by applying Lemma 3 there are sequences w_1 and w_2 such that $m_0 \xrightarrow{w_1} m_1 \xrightarrow{w_2} m$ and w_1, w_2 satisfy the requirements of Lemma 3. In this case it holds especially $m_1(p) = 0$. Thus (since all arcs to p have multiplicity v) we obtain $length(w_2) > k \cdot card(Fp)$. Hence there is one pre–transition t^* of p which appears more than k times in w_2. Since all transitions appearing in w_2 put tokens only on p and therefore not on Ft^*, we get $m_1(p') > k$ for every input $p' \in Ft^*$. Since w_1 can be replaced by a corresponding sequence in N', the marking $m_1 \mid_{P\setminus\{p\}}$ can be reached in N' and has more than k tokens on p'. □

Taking the previous two lemmas together, we obtain

Corollary 1 *N is bounded iff N' is bounded.*

The most difficult proof might be the one concerning liveness.

Lemma 6 *N is live iff N' is live.*

Proof. Assume first, N is not live but N' is live. We show that this is a contradiction. If N is not live then there are a transition t^*, a marking m and a sequence w such that $m_0 \xrightarrow{w} m$ in N and t^* is dead at m in N. By Lemma 3 there are sequences w_1 and w_2 which satisfy the requirements of this lemma and $m_0 \xrightarrow{w_1} m_1 \xrightarrow{w_2} m$ in N. w_1 can be replaced by a corresponding sequence w_1' in N' where all pairs of pre– and post–transitions of p have been replaced by the corresponding combined transitions. Thus it holds $m_0' \xrightarrow{w_1'} m_1'$ in N' and $m_1' = m_1 \mid_{P\setminus\{p\}}$.

We consider a transition $t^{*\prime}$ in N': If $t^* \notin Fp \cup pF$, let $t^{*\prime} = t^*$; if $t^* = t_i \in Fp$, let $t^{*\prime} = t_{ij}$ for some $t_j \in pF$; and if $t^* = t_j \in pF$, let $t^{*\prime} = t_{ij}$ for some $t_i \in Fp$. Now we construct a sequence in N' which can fire at m_1' and which enjoys the following properties: (a) For all i $(1 \leq i \leq \mathit{length}(w_2))$, the ith occurrence of a combined transition is one which is combined from the transition which appears at the position i in w_2 and some post–transition of p; (b) the marking reached by firing the sequence from m_1' enables $t^{*\prime}$.

Of course the reason for this construction is to re–translate this sequence to N and to construct a sequence which enables t^*. Property (a) is necessary, since we need a sequence to be fired from m rather than from m_1.

Consider any sequence u that can be fired at m_1' in N', does not contain combined transitions but enables one. This sequence exists since N' is live. Let this transition be combined from $t_i \in Fp$ and $t_j \in pF$. Thus, after firing u in N from m_1, $t_i t_j$ is enabled. Thus before firing t_i, t_j has already all tokens on its input places except on p. Since u does not contain any occurrences of transitions in Fp, $w_2[1]$ is still enabled after firing u. Hence instead of $t_i t_j$ we can fire $w_2[1] t_j$ in N and the corresponding combined transition in N'. Keeping in mind, that after firing $w_2[1]$ in N there are still enough tokens on $F(Fp)$ to enable $w_2[2] \ldots w_2[\mathit{length}(w_2)]$ and N' is live, we can continue this argumentation yielding finally a sequence in N' which enjoys property (a). Since N' is still live then, it is possible to enlarge this sequence to satisfy property (b), too.

Consider for a sequence satisfying properties (a) and (b) its counterpart in N. Since the transitions in Fp do not share their input places with transitions outside Fp and w_2 is already enabled at m_1, we can shift the first $length(w_2)$ transitions of Fp to the left end of this sequence. By property (a) they form exactly w_2. Thus the remaining sequence can be fired at m and according to property (b) it enables either t^* or (if $t^* \in pF$) a sequence consisting of a transition in Fp and t^*. This is a contradiction to the assumption that t^* is dead at m.

Assume now that N is live but N' is not live. Then there is a sequence in N' where a transition t^* is dead in N'. The counterpart of this sequence can be fired in N and, since N is live, be enlarged by a sequence which enables t^* (or t_j when t^* is a combined transition t_{ij}). Translating this enlarged sequence back to N' after preparing it according to Lemma 3, we obtain an enlarged sequence in N' that enables t^* which is again a contradiction. $\Box$

Benefits

Again a sequence of two transitions can not be interleaved with other events in the reduced net, since it is replaced by a single transition. Thus we pay again a larger number of transitions (if both pF and Fp contain more than one element) for a smaller number of states in the reachability graph.

10.2 Discussion on application to algebraic nets

Since as the previous one this rule combines pre- and post–transitions of a place, we have to restrict the application again to the standard algebra, since (as discussed in the previous section) a combination of all pairs of transitions cannot be done independently of the particular algebra under consideration. Thus we do not need to discuss all the other application conditions for non–standard algebras, though they would raise additional problems, as figure 6 illustrates for the condition $(Fp)F = \{p\}$.

Assume a and b are non–equivalent constant symbols while T and T' are arbitrary terms. In the left part of the figure we have sketched a conflict between t_1 and t_2 which is present in the skeleton algebra, but not in the standard algebra. Thus the considered application condition would be satisfied in the standard algebra but violated in the skeleton algebra. In the right part of the figure t_3 and t_4 are in conflict in every of their firing modes, but while in the standard algebra they produce tokens on different colors of p they produce tokens on the same color of p in the skeleton algebra. Therefore the considered

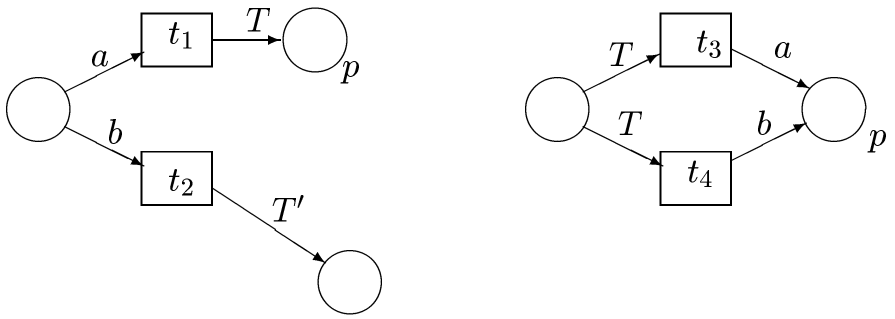

Figure 6: Rule 7 is not polymorphic

part of the application condition would be satisfied in the skeleton algebra but violated in the standard algebra. Consequently even this condition would cause trouble when a class of algebras is to be considered rather than a single one.

Once being restricted to the standard algebra, we can describe *all* conflicts between transitions by means of unification. Two transitions t_1 and t_2 are in (static) conflict iff they share a common input place p^* and there are terms $T_1 \in \lambda([p^*, t_1])$ and $T_2 \in \lambda([p^*, t_2])$ such that T_1 and T_2 are unifiable. According to the general assumption that there is at least one ground term for every sort, such a unifier can be ground–instantiated to firing modes β_1 and β_2 of t_1 and t_2, respectively, both of which consume tokens of the color $[T_1\beta_1]_E$ $(= [T_2\beta_2]_E)$ from p^*. The other way round, a static conflict between the modes β_1 of t_1 and β_2 of t_2 can be arranged to a (ground) unifier $\beta_1 \cup \beta_2$ of some terms appearing at the arcs between p^*, t_1, and t_2.

Thus, the unifiers of terms appearing at input arcs of shared input places do not only answer whether there are static conflicts. A complete set of unifiers additionally describes completely the set of firing modes where these conflicts appear—this is exactly the set of ground instances of the most general unifiers.

With this description we can go ahead and check whether all the conflicting firing modes have the same color of p as their only output. First of all, if every pre–transition mode of a color of p should have only one single output place, then the pre–transitions of p must have only p as their output place. Furthermore there must not be non–equivalent terms at the output arcs of these transitions, since otherwise some modes would put tokens on different colors of p. Thereby, according to the restrictions applying for the previous condition, equivalence is meant with respect to standard algebra. We can leave the task to unify equivalent terms at arc inscriptions to a pre–processing and require that there is only one single term appearing in $\lambda([t, p])$.

Then we can apply the unifiers of input terms of conflicting transitions and check whether their single output colors are equal. For this purpose we simply have to apply the most general unifiers to the terms appearing at the arcs to p and to check whether they become equivalent by this application. The suitable equivalence in this case is $\equiv_I$, since the standard algebra is a term algebra and equivalence in the standard algebra coincides with the equivalence in every term algebra (the basis of $\equiv_I$).

Up to now we have studied the conditions on the pre–transitions of p, except one: that there is one at all. In the low level case this condition follows from $(Fp)F = \{p\}$: if $Fp = \emptyset$ then $(Fp)F$ would be empty as well. In the high–level case this condition is not yet covered by the discussed conditions. So we have to add the condition that every color of p has an input transition mode. We need such a test also for checking that every color of p has a post–transition mode. Such a test was already discussed for rule 1. Being restricted to the standard algebra we can apply term structural methods to check these two conditions, but we do not need to repeat the arguments for applying them. Neither do we need to repeat the technique to apply the rule, since the combination of pairs of transitions works in the same way as for the previous rule.

Thus the remaining condition to be discussed is that post–transitions of p must have the multiplicity v. Since for this rule post–transitions of p are allowed to have other pre–places, we have to consider the case, that a post–transition mode takes tokens of different colors from p. Since we try to apply the rule simultaneously to all colors of p, such a mode would be object of more than one application of the rule. Fortunately one application of the rule does not effect the application condition for other colors. The new transitions inserted by the application remain post–transitions of the remaining colors of p, with the same multiplicity (since the input places of transitions in Fp are different from p). They cannot add new conflicts with pre–places of other colors of p since then the application condition $(F(Fp))F = Fp$ would have been violated before. The application can neither effect the other application conditions for the remaining places. Thus the simultaneous application of the rule for all places is allowed. We must only assure that (a) the multiplicity of a post–transition mode of a color of p equals always v, and (b) the high–level application of the rule implements the multiple application of the low–level rule. Item (a) is implemented by the application condition that different terms appearing at arcs starting form p are not unifiable (thus the colors described by these terms are always different). The second item is implemented by combining a transition in pF where n different terms appear at its arc from p, with n transitions in Fp (including n copies of one and the same transition).

10.3 Application

Application condition

STANDARD: There is a place p and a number v such that

- For all $t \in Fp$ there is a T_t such that $\lambda([t,p]) = v \cdot T_t$;
- For all $t, t' \in Fp$, $p' \in Ft \cap Ft'$, $T \in \lambda([p',t])$, $T' \in \lambda([p',t'])$ and all unifiers σ of T and T', $\lambda([t,p])\sigma \equiv_I \lambda([t',p])\sigma$;
- For all $t \in pF$, there are n_t, $T_{1t}, \ldots, T_{n_t t}$ such that $\lambda(t) = v \cdot T_{1t} + \cdots + v \cdot T_{n_t t}$ and the T_{it} are pairwise not unifiable;
- $pF \cap Fp = \emptyset$;
- *(only for boundedness:) Every $t \in Fp$ has an input place;*
- *There is a term–structural proof that the set of ground instances of $\{T \mid T \in \lambda([t,p]), t \in Fp\}$ covers all elements of $\psi(p)$;*
- *There is a term–structural proof that the set of ground instances of $\{T \mid T \in \lambda([p,t]), t \in pF\}$ covers all elements of $\psi(p)$;*
- *for all $T \in m_0(p)$, $\sum_{T' \equiv_E T} m_0(T') < v$.*

Application

For every $t \in pF$, for all $[t_1, \ldots, t_{n_t}] \in (Fp)^{n_t}$, let τ_i be a renaming of $\xi(t)$ such that all the $\mathrm{var}(\tau_i)$ are pairwise disjoint and disjoint to $\xi(t)$, compute a complete set of unifiers $\mathcal{U}$ for the unification problem

$$\{T_1\tau_1 \stackrel{?}{\equiv}_E T_{t1}, \ldots, T_{n_t}\tau_{n_t} \stackrel{?}{\equiv}_E T_{tn_t}\}$$

For every $\sigma \in \mathcal{U}$ insert a new transition t^* with $\xi(t^*) = \mathrm{var}(\sigma)$, $\lambda([t^*,p']) = \lambda([t,p'])\sigma$ for every $p' \in P \setminus \{p\}$, $\lambda([p',t^*]) = \lambda([p',t])\sigma + \sum_{i=1}^{n_t} \lambda([p',t_i])\tau_i\sigma$.

Remove p, pF, Fp and all arcs connected to removed nodes.

Justification

As in the previous rule we implement the combination of pre– and post–transitions of p based on unification. The repeated application of the rule to transitions which have more than one input color of p is implemented by the combination with tuples of input transitions. Due to the application condition the several combined transitions represent belong for every firing mode of the new transition to different applications of the rule.

11 Rule 8: Side condition places

If all transitions only test a place rather than changing its marking, then it can be removed.

By applying this rule interleaved events may become concurrent, but the interleaving semantics of the net remains unchanged.

Test places may occur in nets which are the Petri net output of a translator from other system descriptions.

11.1 Low level rule

Application condition

There is a place p such that for all transitions t, $V([p,t]) = V([t,p])$.

Application

Remove p, all transitions satisfying $m_0(p) < V([p,t])$ and all arcs connected to removed nodes.

Properties

- All removed transitions are dead in N;
- p is bounded in N;
- All properties and formulas which concern non–removed nodes hold in N iff they hold in N'.

Benefits

The size of the graph does not change at all, but one component of the marking vector is removed and the enabledness check for transitions becomes faster. Moreover interleaved transitions become concurrent which makes this rule interesting for partial order verification methods.

11.2 Discussion on application to algebraic nets

The application condition itself does not cause any trouble. If we want the application to be sound for *all* algebras, we have to check whether a transition t takes in every firing mode as many tokens of every color from p as it puts on it. This means that in every algebra every evaluation of $\lambda([p,t])$ equals the corresponding evaluation of $\lambda([t,p])$ or, in other words: $\lambda([p,t]) \equiv_E \lambda([t,p])$. If we assume that $\lambda([p,t]) = T_1 + \cdots + T_n$ (multiple occurrences of a term are to be represented as repeated occurrence of this term in the sum) and $\lambda([t,p]) = T_1' + \cdots + T_m$, then for satisfying the application condition it is obviously necessary that $m = n$ and sufficient that there is a bijective mapping π such that for all i, $T_1 \equiv_E T'_{\pi(i)}$.

If we restrict our concern to term algebras, we can involve term structural methods like induction on a transition variable or case considerations into the proof that a transition t takes as many tokens of every color as it puts on p. Since the standard algebra is a term algebra and has a surjective morphism to every other term algebra, I–equivalence of $\lambda([p,t])$ and $\lambda([t,p])$ is both necessary and sufficient for satisfying the application condition when only the standard algebra is considered. Thus we do not need any specific methods for the standard algebra only.

The more sophisticated part of translating this rule to algebraic nets is however the application itself. The problem is that we have to distinguish enabled modes of a transition from disabled ones. While the first ones must remain in the net, we have to remove the second ones. A problem occurs when a transition has both enabled and disabled modes. In this case we can only partly provide a solution, especially when classes of algebras are considered. Therefore we have to add additional application conditions for the high–level rule which guarantee that the set of enabled firing modes of a transition can be represented by a set of substitutions. Applying these substitutions to the transition can be read as removing all non–represented firing modes.

Let us consider this problem in detail. Of course, when the number of tokens on p is smaller than the cardinality of the arc inscription of $[p,t]$, then t is dead in every interpretation of the net and we can remove it at all. In all other cases t is at least enabled in the skeleton algebra. Thus we can remove only some colors of t (by applying substitutions to the arc inscriptions). This can in general not be done for every algebra. Assume for instance a token of color a on p and an arc inscription $1 \cdot b$ for $[p,t]$ (a and b are two non–equivalent constant symbols). Then *all* colors of t must be removed when the standard algebra is considered while t must remain in the net when the net is interpreted according to the skeleton algebra. In this case we cannot allow the reduction when *all* algebras are considered. If on the other hand

a and b *are* E–equivalent, we can allow the reduction since in this case t is enabled (according to p) in every firing mode and in every interpretation (E–equivalence means equality in every model of the specification). Thus the reduction can be applied when the arc inscription consists only of ground terms and for every of these ground terms a token with equivalent inscription lies on p. Otherwise the reduction must not be applied when all algebras are considered.

Consider next the case that the arc inscription consists of a single variable, and a single token lies on p. Then every enabled firing mode must assign the inscription of this token to x, independently of the interpretation of the net. That is, we can substitute x in all arc inscriptions around t by this value without removing any enabled firing mode of t. After having done this we can continue as above, since the arc inscription is now a ground term. When there is more than one token on p, then the enabled firing modes assign one of their inscriptions to x. When we insert for every possible choice of a token for x a transition which is obtained by copying t and applying the substitution for x according to the inscription of the chosen token, we have again kept all enabled firing modes of t.

The situation becomes more difficult when the arc inscription consists of more than one variable. As long as all variables appear with multiplicity 1 we can manage the problem. For keeping all enabled firing modes of t, we have to insert a new transition for every possible assignment of tokens to the variables and to substitute in all these transitions the variables by the assigned token inscriptions. But if only one variable appears at least twice and there are two tokens with non–equivalent inscriptions, we get into trouble. Assume the arc inscription of $[p, t]$ is $2 \cdot x$ and there are tokens inscribed with a and b (not equivalent) on p. In the standard algebra t does not have any enabled mode while in any algebra where the same value is assigned to a and b, t is enabled with respect to p. That is, the whole transition must be removed according to the standard algebra while it must remain according to the skeleton algebra. This problem does not appear when all token inscriptions on p are equivalent.

Ground terms and variables can be combined in arc inscriptions. After having assigned a token to every ground term, we have to traverse all the combinations of assigning the remaining tokens to the variables.

Thus it remains the case that terms appear in the arc inscriptions which contain variables but are not variables themselves. In this case the set of firing modes in which t can be enabled with respect to p depends so strong on the interpretation of the specification that we decided not to allow a reduction in this case at all.

The situation remains roughly the same when we restrict our concern to term algebras (we were able to explain all major obstacles using standard and skeleton algebra—both term algebras). Not even the equivalence between ground terms at the arc inscription and in the marking changes, since for ground terms E–equivalence and I–equivalence are the same.

So we can finally regard the case that we restrict ourselves to the standard algebra only. In this case we can say much more about enabled and disabled firing modes. The enabled modes of t with respect to p can be exactly characterized by a set of substitutions, as discussed in detail in [Sch95b, Sch96c]. We must only traverse all combinations of assigning a token inscription to a term appearing in the arc inscriptions and unify the assigned pairs. The set of all computed most general unifiers characterizes the set of all enabled firing modes. Thus we can replace t by a set of copies of t were to every copy one of the unifiers has been applied. This way the enabled modes of t remain in the net while the disabled ones are deleted.

11.3 Application

Application condition

There is a place p such that for all transitions t:

- *ALL:* $\lambda([p,t]) \equiv_E \lambda([t,p])$;
- *TERM/STANDARD:* $\lambda([p,t]) \equiv_I \lambda([t,p])$;
- *ALL/TERM:*
  - $card(m_0(p)) < card(\lambda([p,t])$ or
  - $\lambda([p,t])$ consists only of ground terms and variables, where
    - · every variable appears with multiplicity 1 or
      · all terms appearing in $m_0(p_)$ are equivalent,
      and
    - for every ground term T it holds

$$card(\lambda([p,t])([T]_E) \geq card(m_0)(p)([T]_E).$$

Application

For every injective assignment of an occurrence[†] of a term in $m_0(p)$ to $\lambda([p,t])$ and every unifier σ for the set of pairs of assigned terms, insert a transition

[†] When a term appears with multiplicity k, then we consider k occurrences.

t_σ where $\xi(t_\sigma) = var(\sigma)$, and for all $p' \in P \setminus \{p\}$, $\lambda([p', t_\sigma]) = \lambda([p', t])\sigma$ and $\lambda([t_\sigma, p']) = \lambda([t, p'])\sigma$.

Remove p, pF and all arcs connected to removed nodes.

Justification

When a ground term and a variable are unified, then assigning the ground term to the variable is the most general unifier. Therefore, though we have described the application in a common way for standard algebra and classes of algebras, we have implemented in both cases the procedure we have explained above.

12 Rule 9: Loop transitions

When a transition puts as many tokens to every places as it takes, it can be removed.

12.1 Low level rule

Application condition

There is a transition t such that for all places p, $V([p, t]) = V([t, p])$.

Application

Remove t and all arcs connected to it.

Properties

- N is bounded iff N' is;
- All transitions in $T \setminus \{t\}$ are live in N iff they are live in N';
- Every formula holds in N iff it holds in N'.

Benefits

Firing a loop transition leads back to the same marking. Thus the size of a reachability graph does not change at all by this reduction. However we save the arcs corresponding to t and the corresponding enabling checks.

12.2 Discussion on application to algebraic nets

Being dual to the previous rule, we can translate it to algebraic nets in the same manner. Thereby things become much easier since we do not need to care about enabling or disabling of transitions. Thus we have simply to check the equivalence of the arc inscriptions.

12.3 Application

Application condition

There is a transition t such that for all places p:

- *ALL:* $\lambda([p,t]) \equiv_E \lambda([t,p])$;
- *TERM/STANDARD:* $\lambda([p,t]) \equiv_I \lambda([t,p])$;

Application

Remove t and all arcs connected to it.

Justification

This rule is apparrent.

13 Conclusion

Though the discussion on the applicability of the different rules was rather informal, we could show that net reduction is a feasible tool as a pre–processing to other analysis algorithms for algebraic Petri nets. The approach allows partly for a fully automated reduction, but as well for a semi–automated approach where the user controls the application interactively.

Both approaches are supported by tools to check the aplication conditions: there are tools trying to reason upon specifications automatically, and others where the user selects proof strategies. This way the reduction approach can be customized to different types of application.

The application of the rules can be scaled according to the range of soundness. For most rules we were able to find application conditions where the rules are sound for *all* interpretations or for all *term* interpretations. Thereby for every rule specific additional requirements were necessary. That is, there seems to be no general correlation between net reduction and the range of soundness for the application. For two rules (6 and 7) the application is totally restricted to the standard interpretation. Unfortunately these two belong to the more powerful rules which allow to reduce the state space of the net.

In two cases we had to generalize the low level rules in order to find a suitable high–level approach. We have proved the soundness of these rules which low level reduction tools can now benefit from.

The reduction approach complements several other symbolic approaches for symbolic analysis of high level nets. The symbolic approach to the analysis of algebraic nets covers already reachability analysis including stubborn reduction [Sch95b], invariant verification and calculation [Sch94, Sch95a], and the verification and calculation of siphons and traps [Sch96a, Sch96b].

While for the calculus of siphons, traps and place invariants strong restrictions to the involved specification are necessary, we were able to keep the application conditions for reduction rather general. Of course the application of tools for verifying the conditions depends on "nice behaving" specifications, for instance on the decidability of the equivalence problem, the existence of finetely many most general unifiers and so on, but these are restrictions which apply to the whole symbolic approach. When judging about these restrictions, one should keep in mind that the approach is an attempt to cover as far as possible the case of large and infinite data types, where analysis problems are in general undecidible. For finite data types we get an additional option to other approaches. When running several procedures in parallel, say on different machines, one can benefit even from incomplete and restricted procedures.

References

[Ber83] G. Berthelot. *Transformation et analyse de réseaux de Petri, application aux protocoles.* PhD thesis, Université P. et M. Curie, Paris, 1983.

[DE95] J. Desel and J. Esparza. *Free Choice Petri Nets.* Cambridge Tracks in Theoretical Computer Science 40, 1995.

[EM85] H. Ehrig and B. Mahr. *Fundamentals of Algebraic Specifications*, volume 1 of *EATCS Monographs on Theoretical Computer Science 6*. Springer, Berlin, 1985.

[Had89] S. Haddad. A reduction theory for coloured nets. *Advances in Petri nets, LNCS 424*, 1989.

[JK91] J.-P. Jouannaud and C. Kirchner. Solving equations in abstract algebras: A rule–based survey of unification. *Computational Logic: Essays in Honor of Alan Robinson*, pages 257–321, 1991.

[KR89] P. Krohn and M. Rauhamaa. Reduction transformations of prt–nets. Technical report, Helsinki University of Technology, Otaniemi, 1989.

[Pau86] L.C. Paulson. Isabelle. *LNCS 828, Springer, 1994*, 1986.

[Rei91] W. Reisig. Petri nets and algebraic specifications. *Theoretical Computer Science*, 80:1–34, 1991.

[RM86] R.S.Boyer and J.S. Moore. Overview of a theorem–prover for a computational logic. *Proceedings of the 8th Conference on Automated Deduction, Springer*, 1986.

[Sch94] K. Schmidt. T–invariants of algebraic Petri nets. *Informatik–Bericht*, 31, 1994.

[Sch95a] K. Schmidt. On the computation of place invariants for algebraic Petri nets. *Proceedings of the STRICT workshop Berlin*, 1995.

[Sch95b] K. Schmidt. Parameterized reachability trees for algebraic Petri nets. *Proc. of the 16th International Conference on Application and Theory of Petri Nets*, pages 392–411, 1995.

[Sch96a] K. Schmidt. How to calculate symbolically siphons and traps for some algebraic Petri nets. Technical report, Helsinki University of Technology, Otaniemi, 1996.

[Sch96b] K. Schmidt. Symbolic verification of siphons and traps for algebraic Petri nets. *In preparation*, 1996.

[Sch96c] K. Schmidt. *Symbolische Analysemethoden für algebraische Petri–Netze*. PhD thesis, Humboldt–Universitt zu Berlin, 1996.

[Sta90] P. Starke. *Analyse von Petri–Netz–Modellen.* B.G. Teubner Stuttgart, 1990.

[Wol92] D. Wolz. A tool for checking adt completeness and consistency. *Proceedings of the 3rd LOTOSPHERE workshop, Pisa*, 1992.