Resource-Constrained Workflow nets

Kees van Hee Natalia Sidorova Marc Voorhoeve

Department of Mathematics and Computer Science

Technische Universiteit Eindhoven

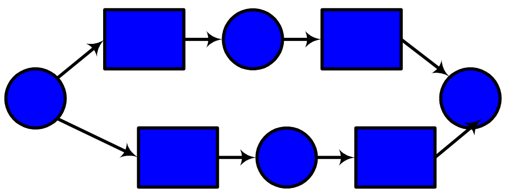
The Netherlands

TU/e

Workflow nets

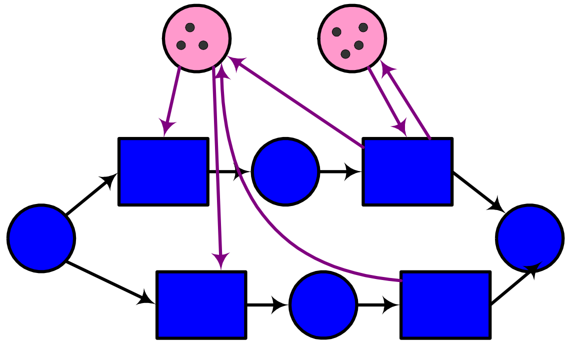
A Petri net N is a Workflow net (WF-net) iff:

- N has two special places (or transitions): an initial place (transition) i: ${}^\bullet i = \emptyset$, and a final place (transition) f: $f^\bullet = \emptyset$.
- For any node $n \in (P \cup T)$ there exists a path from i to n and a path from n to f.



Applications: business process modelling, software engineering, ...

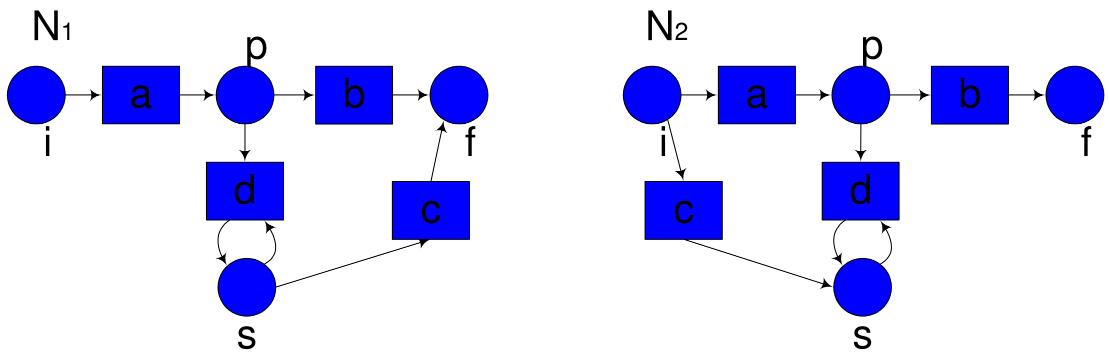
Resource-Constrained WF-nets



A Petri net $= \langle P_p \cup P_r, T, F_p^+ \cup F_r^+, F_p^- \cup F_r^- \rangle$ is a Resource-Constrained Workflow net (RCWF-net) iff:

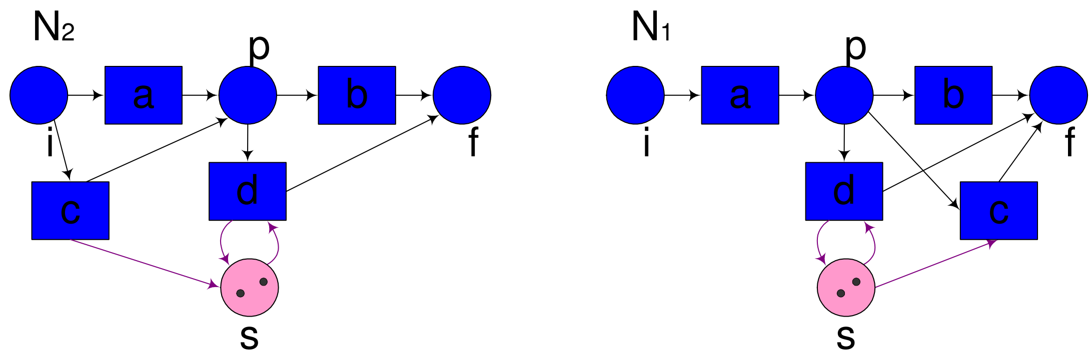
- $P_r \neq \emptyset$ and $P_p \cap P_r = \emptyset$,
- F_p^+ and F_p^- are mappings $(P_p \times T) \rightarrow \mathbb{N}$,
- F_r^+ and F_r^- are mappings $(P_r \times T) \rightarrow \mathbb{N}$, and
- $N_p = \langle P_p, T, F_p^+, F_p^- \rangle$ is a WF-net, which we call a production net of N.

Non-redundancy and non-persistency



- Non-redundancy: every transition can potentially fire and every production place can potentially obtain tokens, provided that there are enough tokens on the initial place and resource tokens.
- Non-persistency: it should be possible for every production place (except for f) to become unmarked again.

Redundancy and persistency



- Redundancy: no resource place can ever obtain tokens, if it was not marked initially.
- Persistency: every resource place should become marked again when the net terminates.

Formally:

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

- A place $p \in P$ is non-redundant iff $\exists k \in \mathbb{N}, m \in \mathbb{N}^P : k[i] \xrightarrow{*} m \wedge p \in m$.
- A place $p \in P$ is non-persistent iff $\exists k \in \mathbb{N}, m \in \mathbb{N}^P : p \in m \wedge m \xrightarrow{*} k[f]$.
- A transition t is non-redundant iff $\exists k \in \mathbb{N}, m \in \mathbb{N}^P : k[i] \xrightarrow{*} m \xrightarrow{t}$.

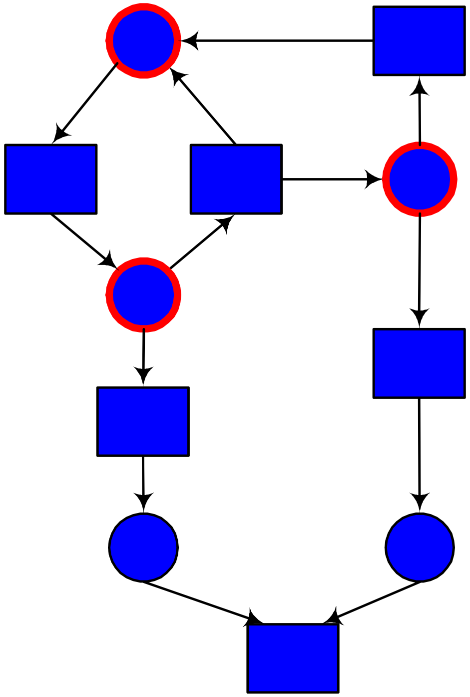
All production places should be non-redundant and non-persistent;
all resource places should be redundant and persistent.

Siphons

A set R of places is a siphon if ${}^{\bullet}R \subseteq R^{\bullet}$.
A siphon is a proper siphon if it is not empty.

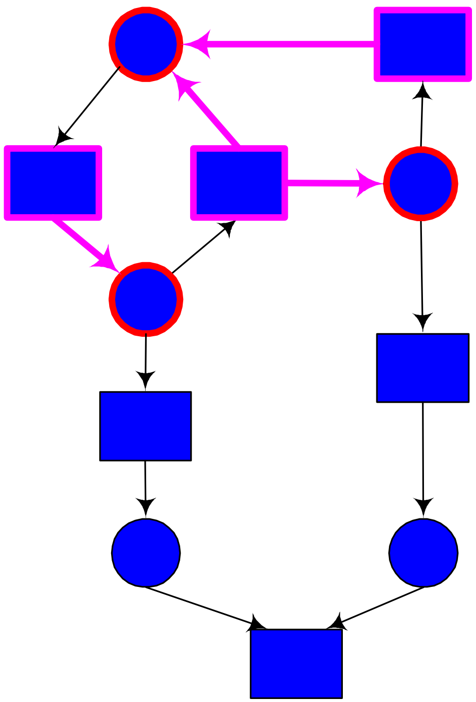
Siphons

A set R of places is a siphon if $^\bullet R \subseteq R^\bullet$.
A siphon is a proper siphon if it is not empty.



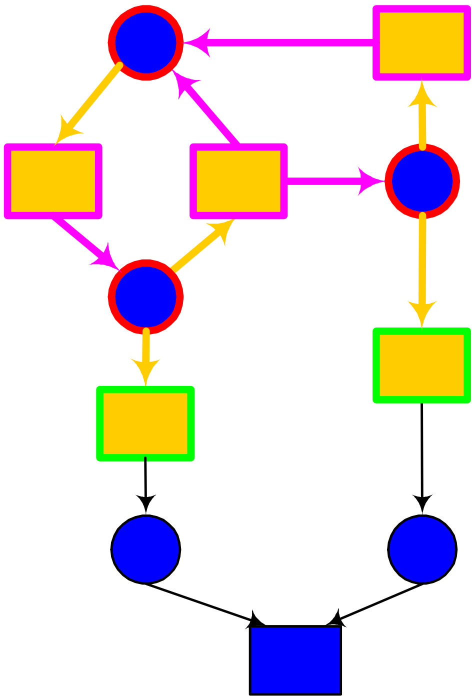
Siphons

A set R of places is a siphon if ${}^\bullet R \subseteq R^\bullet$.
A siphon is a proper siphon if it is not empty.



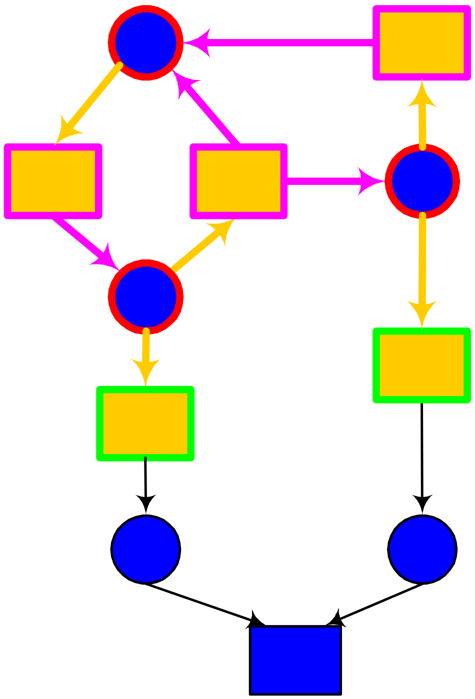
Siphons

A set R of places is a siphon if $^{\bullet}R \subseteq R^{\bullet}$.
A siphon is a proper siphon if it is not empty.



Siphons

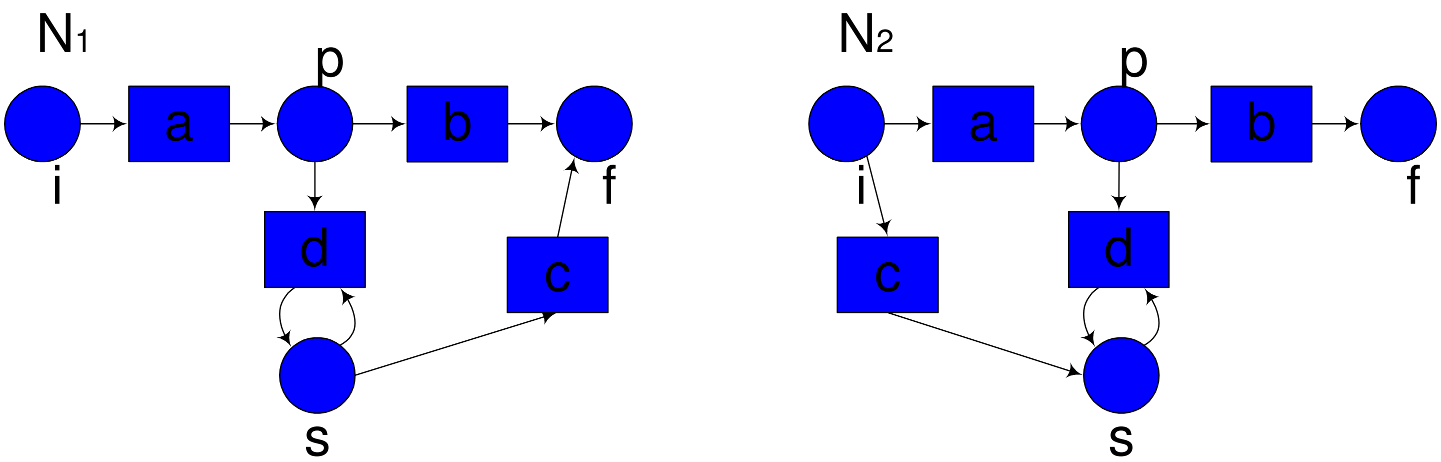
A set R of places is a siphon if ${}^\bullet R \subseteq R^\bullet$.
A siphon is a proper siphon if it is not empty.



Unmarked siphons remain unmarked

Non-redundancy criterion

- A WF-net has no redundant places iff $P \setminus \{i\}$ contains no proper siphon.
- A WF-net has no redundant places iff it has no redundant transitions.

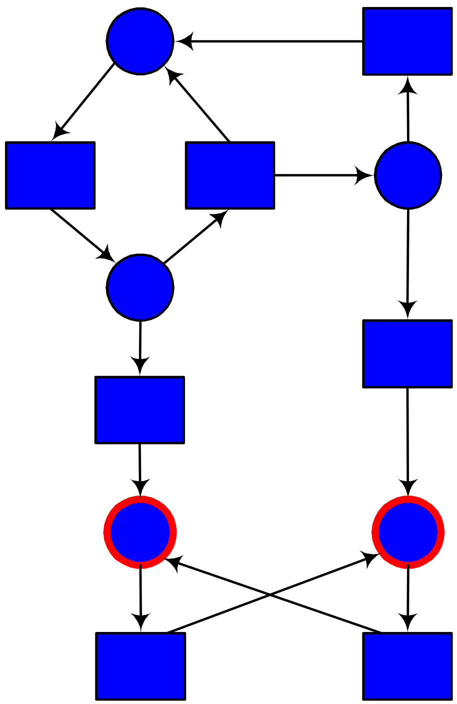


Traps

A set R of places is a trap if $R^\bullet \subseteq {}^\bullet R$.
A trap is a proper trap if it is not empty.

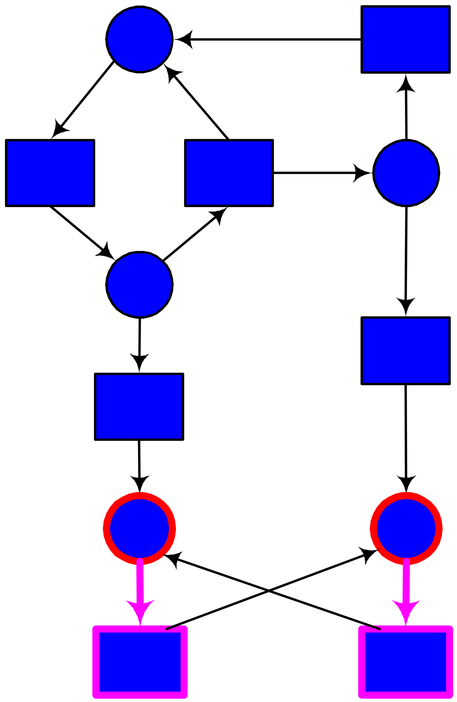
Traps

A set R of places is a trap if $R^\bullet \subseteq {}^\bullet R$.
A trap is a proper trap if it is not empty.



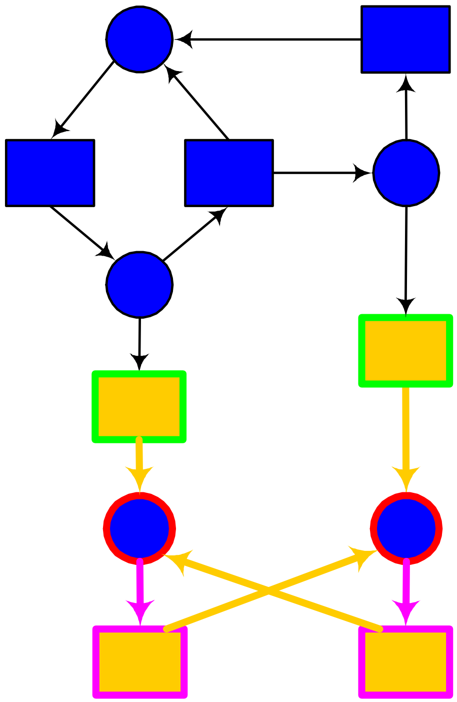
Traps

A set R of places is a trap if $R^\bullet \subseteq {}^\bullet R$.
A trap is a proper trap if it is not empty.



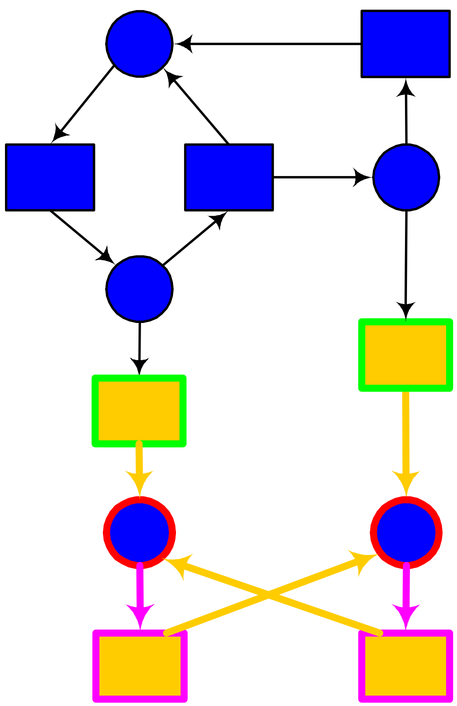
Traps

A set R of places is a trap if $R^\bullet \subseteq {}^\bullet R$.
A trap is a proper trap if it is not empty.



Traps

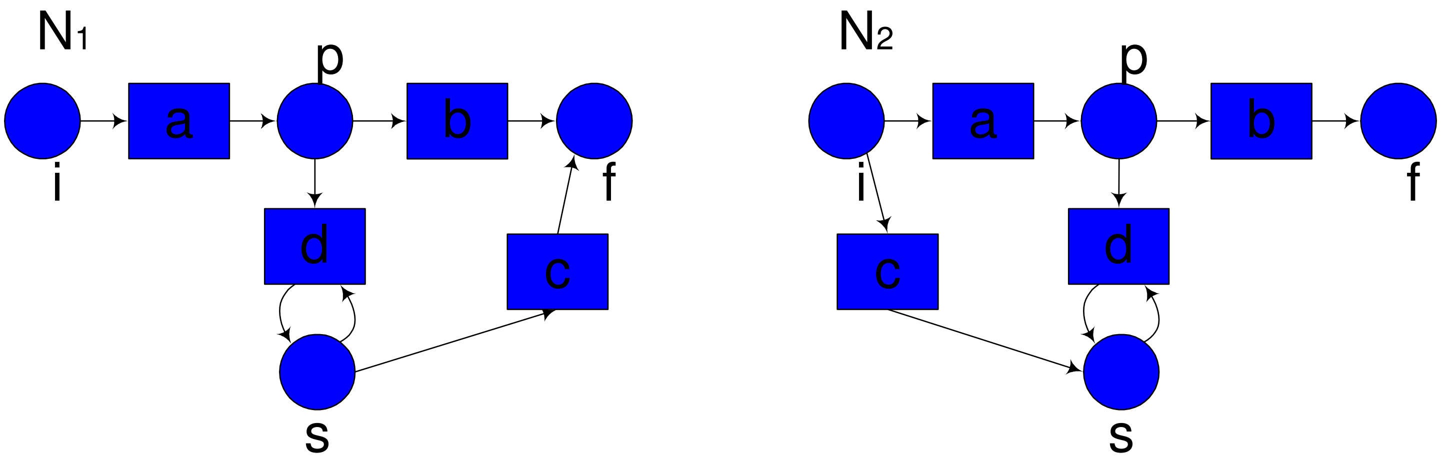
A set R of places is a trap if $R^\bullet \subseteq {}^\bullet R$.
A trap is a proper trap if it is not empty.



Marked traps remain marked.

Non-persistency criterion

- A WF-net has no persistent places iff $P \setminus \{f\}$ contains no proper trap.

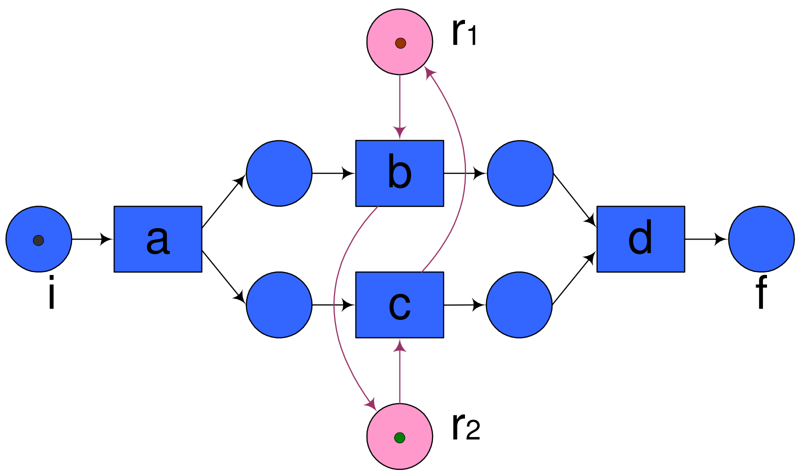


A check for structural correctness

- Check that the production net has no redundant places and no persistent places;
- Check that all resource places are redundant and persistent;
- Check whether resources are independent of each other, if necessary: resource r is independent of other resources iff place r is redundant and persistent in the net with all other resource places removed.

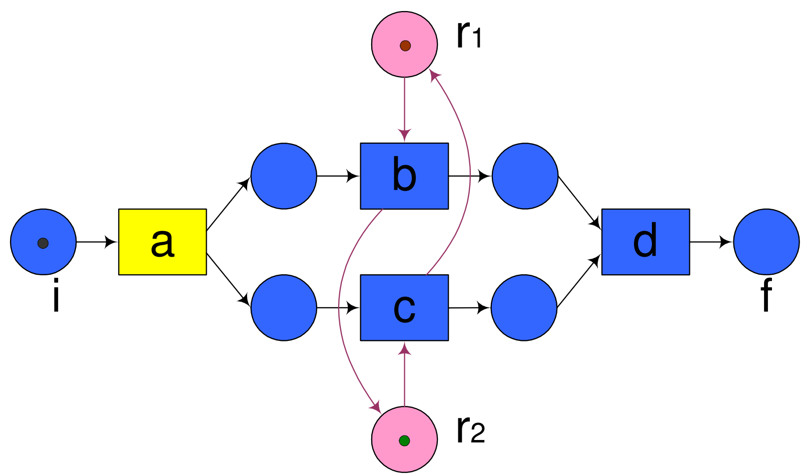
A check for structural correctness

- Check that the production net has no redundant places and no persistent places;
- Check that all resource places are redundant and persistent;
- Check whether resources are independent of each other, if necessary: resource r is independent of other resources iff place r is redundant and persistent in the net with all other resource places removed.



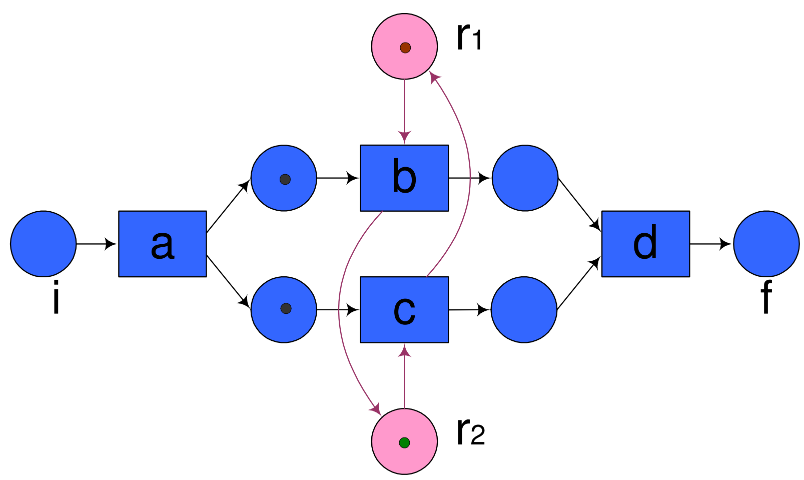
A check for structural correctness

- Check that the production net has no redundant places and no persistent places;
- Check that all resource places are redundant and persistent;
- Check whether resources are independent of each other, if necessary: resource r is independent of other resources iff place r is redundant and persistent in the net with all other resource places removed.



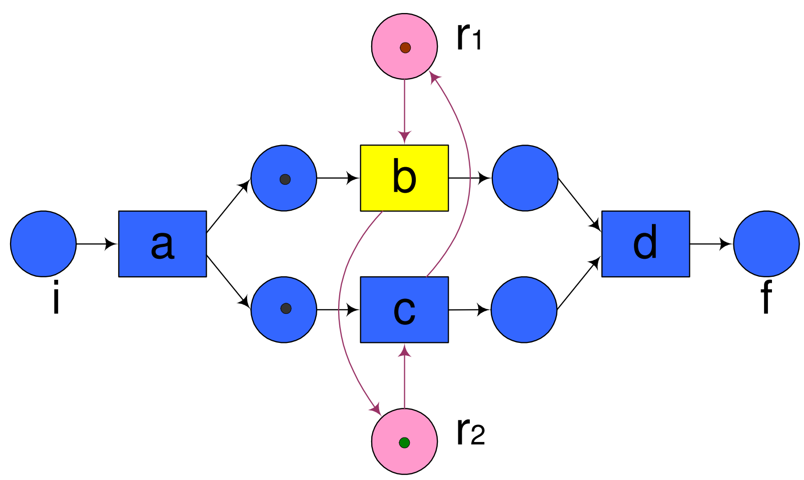
A check for structural correctness

- Check that the production net has no redundant places and no persistent places;
- Check that all resource places are redundant and persistent;
- Check whether resources are independent of each other, if necessary: resource r is independent of other resources iff place r is redundant and persistent in the net with all other resource places removed.



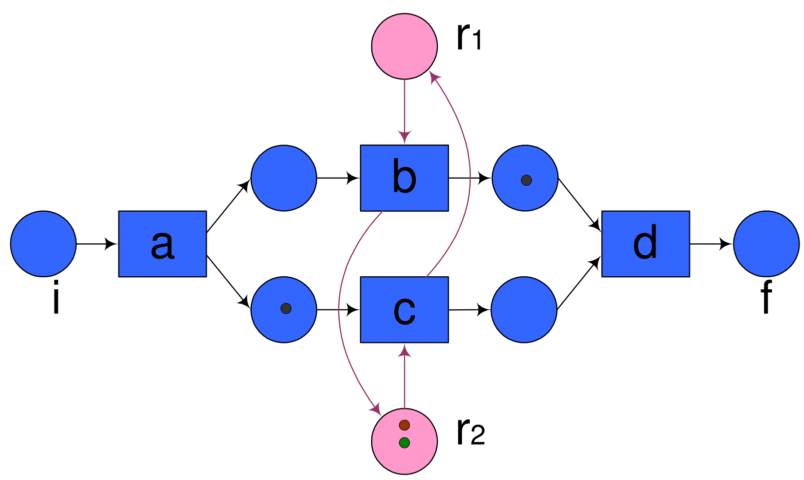
A check for structural correctness

- Check that the production net has no redundant places and no persistent places;
- Check that all resource places are redundant and persistent;
- Check whether resources are independent of each other, if necessary: resource r is independent of other resources iff place r is redundant and persistent in the net with all other resource places removed.



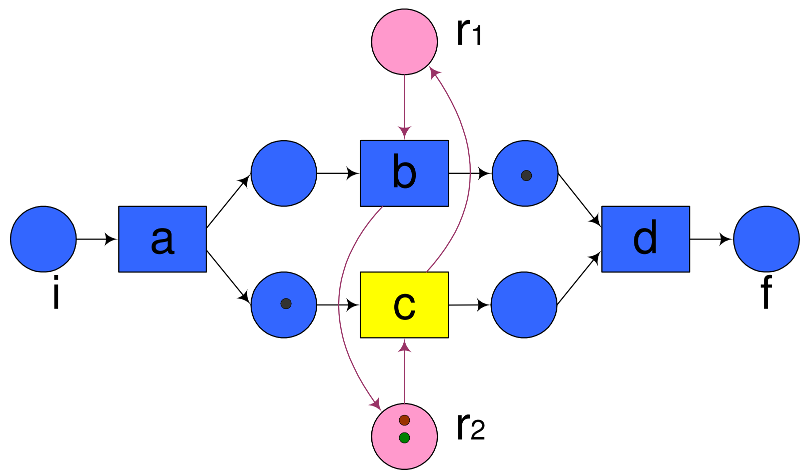
A check for structural correctness

- Check that the production net has no redundant places and no persistent places;
- Check that all resource places are redundant and persistent;
- Check whether resources are independent of each other, if necessary: resource r is independent of other resources iff place r is redundant and persistent in the net with all other resource places removed.



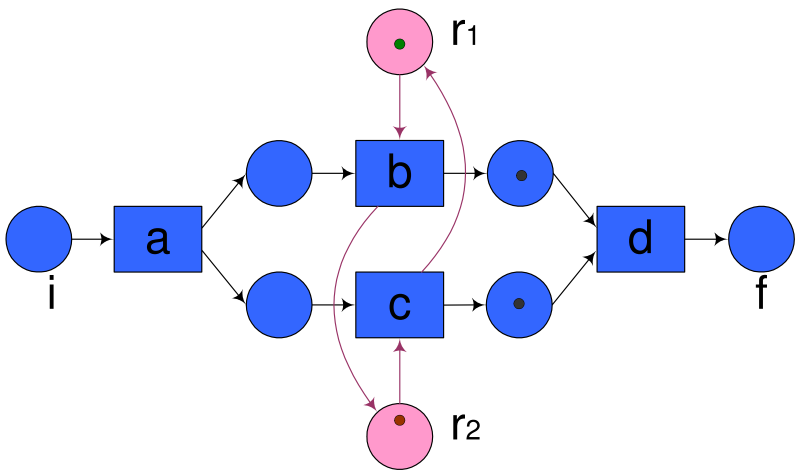
A check for structural correctness

- Check that the production net has no redundant places and no persistent places;
- Check that all resource places are redundant and persistent;
- Check whether resources are independent of each other, if necessary: resource r is independent of other resources iff place r is redundant and persistent in the net with all other resource places removed.



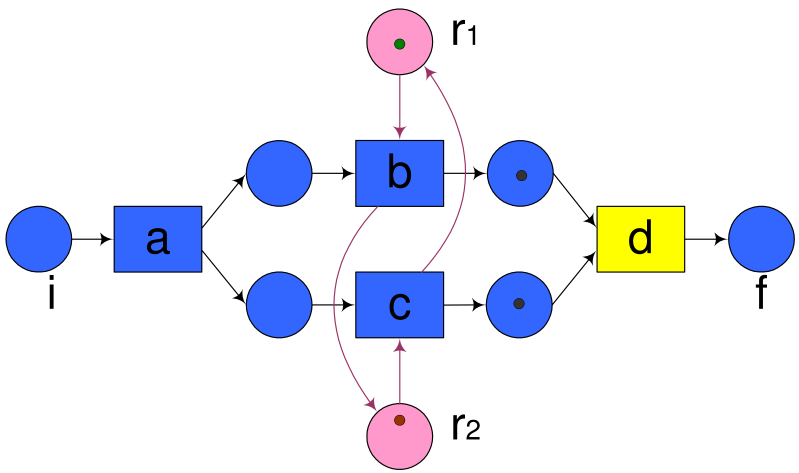
A check for structural correctness

- Check that the production net has no redundant places and no persistent places;
- Check that all resource places are redundant and persistent;
- Check whether resources are independent of each other, if necessary: resource r is independent of other resources iff place r is redundant and persistent in the net with all other resource places removed.



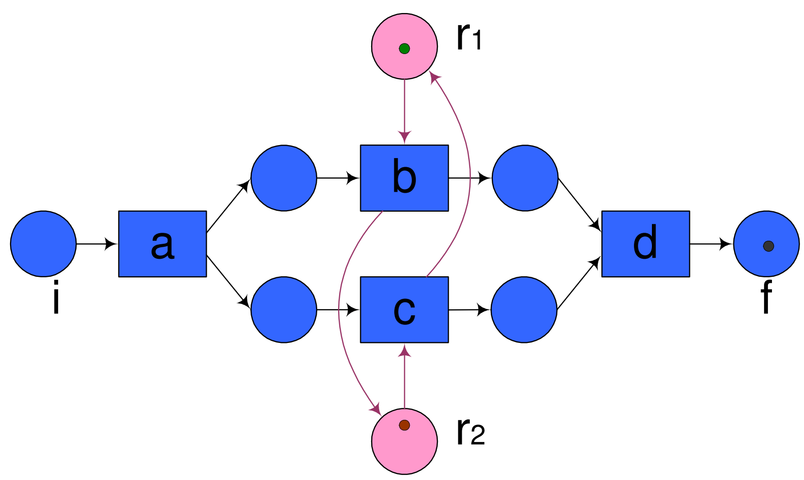
A check for structural correctness

- Check that the production net has no redundant places and no persistent places;
- Check that all resource places are redundant and persistent;
- Check whether resources are independent of each other, if necessary: resource r is independent of other resources iff place r is redundant and persistent in the net with all other resource places removed.



A check for structural correctness

- Check that the production net has no redundant places and no persistent places;
- Check that all resource places are redundant and persistent;
- Check whether resources are independent of each other, if necessary: resource r is independent of other resources iff place r is redundant and persistent in the net with all other resource places removed.



Soundness

Desired property: proper completion

Soundness for WF-nets:

A WF-net N with initial and final places i and f resp. is *k-sound* for $k \in \mathbb{N}$ iff $[f^k]$ is reachable from all markings m from $\mathcal{M}(N, [i^k])$.

A WF-net is (generalised) sound iff it is k-sound for every natural k.

Generalised soundness is decidable
[van Hee, Sidorova, Voorhoeve 2004]

TU/e

Soundness of RCWF-nets

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

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

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

Soundness of RCWF-nets

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

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

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

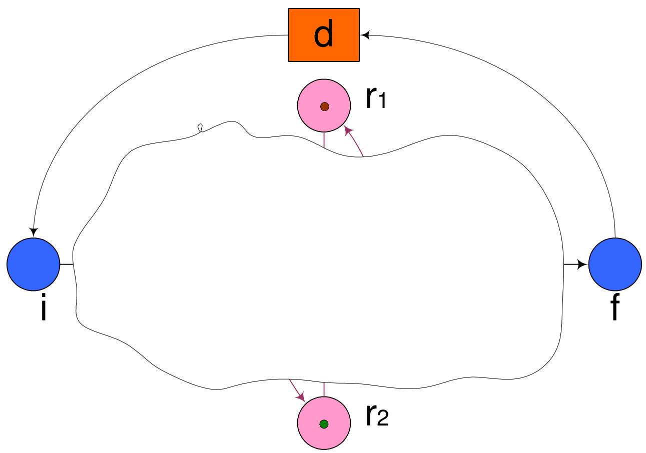
(1) If N is k-sound, the underlying production WF-net N_p is k-sound as well.
(2) If N is sound, N_p is sound, too.

Soundness and transition invariants

Soundness includes the requirement to work correctly for all “large” markings.

$$\Downarrow$$

Every transition invariant of the closure of the production net is a transition invariant of the the closure of the RCWF-net N.



Soundness and transition invariants

Check that $\forall x \in \mathbb{Z}^T : F'_p \cdot x = 0 \Leftrightarrow F' \cdot x = 0.$

If not, the net is not sound.

If yes, then if no deadlock or livelock occurs due to the lack of resources, then the RCWF-net terminates properly:

Let N be an RCWF-net such that its production net N_p has no redundant transitions, and for the closure nets N' and N'_p holds $\forall x \in \mathbb{Z}^T : F'_p \cdot x = 0 \Leftrightarrow F' \cdot x = 0$.
Then for any $k \in \mathbb{N}, m_r \in \mathbb{N}^{P_r}, m' \in \mathbb{N}^P$,
$k[i] + m_r \xrightarrow{*} k[f] + m'$ implies $m_r = m'$.

TU/e

Soundness and place invariants

Let N be a sound RCWF-net and r is a resource place. Then there exists a place invariant $I \in \mathcal{I}$ such that $I(i) = I(f) = 0$ and $I(r) \neq 0$.

An additional characterization of resource independence:

In a sound net, all invariants satisfy $I(i) = I(f)$.

Decompose a linear space $\mathcal{I}$ of all place invariants into the subspaces $\mathcal{I}_P$, the production invariants, and $\mathcal{I}_R$, the resource invariants satisfying $I(i) = I(f) = 0$.

If the resources are independent, $\mathcal{I}_R$ is decomposable into subspaces $\mathcal{I}_r$ for $r \in P_r$ such that $\forall I \in \mathcal{I}_r, q \in P_r : I(q) \neq 0 \iff q = r$.

TU/e

Soundness and place invariants

A desirable property for RCWF-nets with independent resources is the existence of bases for the $\mathcal{I}_r$ having nonnegative coefficients (i.e. resources can only become available when released after being claimed earlier).

RCWF-nets with this property are connected to the S^4PR nets of [Colom2003].

Conclusion

- (Non-)redundancy and (non-)persistency: simple structural correctness checks with the use of traps and siphons
- Soundness of the production net is necessary for the soundness of the RCWF-net
- Transition invariants of the closure of the production net are the same as of the closure of the RCWF-net, if the RCWF-net is sound. (guarantee for the resource conservation)
- Soundness implies the existence of a resource place invariant for every resource place, which relates sound RCWF-nets to S^4PR nets [Colom2003].

Related works

- Colom, Ezpeleta, Martinez, Silva, Turuel et al.
  Flexible manufacturing systems: the key issue is the construction of appropriate schedules.
- Barkaoui&Petrucci:
  Nets with shared resources: structural soundness corresponds approx. to the existence of k, m_r such that the net is (k, m_r) sound.

Future work

- We gave only necessary conditions for soundness. What are sufficient conditions?
- Is the soundness problem decidable for RCWF-nets?
- What are the structural patterns for building sound-by-construction RCWF-nets?