Verifying Generalized Soundness for Workflow Nets

Olivia Oanea
(joint work with Kees van Hee, Natalia Sidorova and Marc Voorhoeve)

Architecture of Information Systems
Department of Mathematics and Computer Science
Technical University Eindhoven

June 8, 2006
ProSe
Generalized soundness for Workflow nets: Motivation

Preserving correctness of (WF) nets by refinement
Generalized soundness for Workflow nets: Motivation

Preserving correctness of (WF) nets by refinement

[Diagram of a Workflow net]
A Petri net $N = (P, T, F)$ is a workflow net (WF-net) iff:

1. $N$ has two special places: $i$ — the initial place with $i^\bullet = \emptyset$, and the final place $f$ with $f^\bullet = \emptyset$.
2. Every node $n \in (P \cup T)$ is on a path from $i$ to $f$.

Generalized soundness for WF nets

A WF-net $N$ is generalized sound iff for all $k \in \mathbb{N}$, all markings reachable from $k \cdot \vec{i}$ terminate properly, i.e. $m \overset{*}{\rightarrow} k \cdot \vec{f}$
A Petri net \( N = (P, T, F) \) is a Workflow net (WF-net) iff:

1. \( N \) has two special places: \( i \) — the initial place with \( \bullet i = \emptyset \), and the final place \( f \) with \( f^\bullet = \emptyset \).
2. Every node \( n \in (P \cup T) \) is on a path from \( i \) to \( f \).

Generalized soundness for WF nets

A WF-net \( N \) is generalized sound iff for all \( k \in \mathbb{N} \), all markings reachable from \( k \cdot i \) terminate properly, i.e. \( m \vdash^* k \cdot f \)
1. Old procedure for deciding generalized soundness for WF nets

2. New Decision procedure for the generalized soundness of BWF-nets

3. Practical Application of the Decision Procedure
Decidability

Generalized soundness problem for Workflow nets is decidable

K. van Hee, N. Sidorova, and M. Voorhoeve.

Main ideas

- A WF-net $N$ is generalized sound iff a certain BWF-net $N'$ can be derived from it and $N'$ is generalized sound.
- Verifying generalized soundness on $N'$ is reduced to a finite number of proper termination checks in $N'$. 
Batch Workflow nets

**Trap**
A subset of places $Q$ is called a \textit{trap} if $Q^\bullet \subseteq \bullet Q$.

**Siphon**
A subset $Q \subseteq P$ is called a \textit{siphon} if $\bullet Q \subseteq Q^\bullet$.

---

**Definition**
A Batch Workflow net (BWF-net) $N$ is a WF-net having the following properties:

1. every non-empty siphon of $N$ contains $i$;
2. every non-empty trap of $N$ contains $f$.
Old decision procedure for the generalized soundness of BWF-nets

Facts

1. $m \xrightarrow{\sigma} m'$ implies $m' = m + F \cdot \sigma$
2. $m \xrightarrow{\sigma} m'$ implies $\mathcal{I} \cdot m = \mathcal{I} \cdot m'$, where $\mathcal{I}$ is the matrix having place invariants as rows

If $N$ is generalized sound then

1. $\mathcal{I} \cdot \bar{i} = \mathcal{I} \cdot \bar{f}$ since $\bar{i} \xrightarrow{*} \bar{f}$
2. $\mathcal{I} \cdot x = \bar{0}$ has only the trivial solution on $\mathbb{N}^P$. otherwise if $x > \bar{0} \Rightarrow x \xrightarrow{*} \bar{0}$ — false since $t^* \neq \emptyset$

Generalized soundness $\iff$ proper termination of

- $\mathcal{R} = \bigcup_{k \in \mathbb{N}} \mathcal{R}(k \cdot \bar{i}) = \bigcup_{k \in \mathbb{N}} \{k \cdot \bar{i} + F \cdot \nu | \nu \in \mathbb{N}^T \} \cap \mathbb{N}^P$
- $\mathcal{G} = \bigcup_{k \in \mathbb{N}} \mathcal{G}_k$, where $\mathcal{G}_k = \{k \cdot \bar{i} + F \cdot \nu | \nu \in \mathbb{Z}^T \} \cap \mathbb{N}^P$
  all markings $m \in \mathcal{G}_k$ have the same $i$-weight $w(m) = k$
Old decision procedure for the generalized soundness of BWF-nets

Generalized soundness $\iff$ proper termination of a finite $\Gamma \subseteq \mathcal{G}$

- $\mathcal{H} = \{ a \cdot \bar{i} + F \cdot v | a \in \mathbb{Q}^+, v \in \mathbb{Q}^T \} \cap (\mathbb{Q}^+)^P$ is a convex polyhedral cone and has a finite set of generators $E = \{ e_1, \ldots, e_n \}$;
- $E_G = \{ e^1, \ldots, e^n \} \in \mathcal{G}$ is the set of rescaled generators in $\mathcal{G}$;
- $\Gamma = \{ \sum_i \alpha_i \cdot e^i \leq 1 \} \cap \mathcal{G}$ is the set of markings (integer points) of the polytope having as generators $E_G$

Decision Procedure

1. Check whether $\mathcal{I} \cdot \bar{i} = \mathcal{I} \cdot \bar{f}$
2. Check whether $\mathcal{I} \cdot x = \bar{0}$ has only the trivial solution on $\mathbb{N}^P$.
3. Check proper termination for $\Gamma$
Old decision procedure for the generalized soundness of BWF-nets

**Generalized soundness** ⇔ **proper termination of a finite** $\Gamma \subseteq \mathcal{G}$

- $\mathcal{H} = \{ a \cdot \bar{t} + F \cdot v | a \in \mathbb{Q}^+, v \in \mathbb{Q}^T \} \cap (\mathbb{Q}^+)^P$ is a convex polyhedral cone and has a finite set of generators $E = \{ e_1, \ldots, e_n \}$;
- $E_G = \{ e^1, \ldots, e^n \} \in \mathcal{G}$ is the set of rescaled generators in $\mathcal{G}$;
- $\Gamma = \{ \sum_i \alpha_i \cdot e^i \leq 1 \} \cap \mathcal{G}$ is the set of markings (integer points) of the polytope having as generators $E_G$.

**Decision Procedure**

1. Check whether $\mathcal{I} \cdot \bar{t} = \mathcal{I} \cdot \bar{f}$
2. Check whether $\mathcal{I} \cdot x = \bar{0}$ has only the trivial solution on $\mathbb{N}^P$.
3. Check proper termination for $\Gamma$. 
Computing $\Gamma$ - example

$\Gamma$ is very large

- $(4, 1, 1, 4) \cdot \vec{i} = (4, 1, 1, 4) \cdot \vec{f}$
- $(4, 1, 1, 4) \cdot x = \vec{0}$ implies $x = \vec{0}$
- $\mathcal{H} = \{ a \cdot \vec{i} + F \cdot v | a \in \mathbb{Q}^+, v \in \mathbb{Q}^T \} \cap (\mathbb{Q}^+)^P = (A + B) \cap \{ \vec{i}, \vec{f}, \vec{a}, \vec{b} \}$,
  $A = \{ \vec{i} \}$ and $B = \{ \pm(3 \cdot \vec{a} + \vec{b} - \vec{i}), \pm(\vec{a} + \vec{b}), \pm(\vec{i} - \vec{a} - 3 \cdot \vec{b}) \}$
- $E = \{ \vec{i}, \vec{f}, \vec{a}, \vec{b} \}$
- $E_G = \{ \vec{i}, \vec{f}, 8 \cdot \vec{a}, 8 \cdot \vec{b} \}$
- $|\Gamma| = 44$
Computing $\Gamma$ - example

- $(4, 1, 1, 4) \cdot \bar{i} = (4, 1, 1, 4) \cdot \bar{f}$
- $(4, 1, 1, 4) \cdot x = \bar{0}$ implies $x = \bar{0}$
- $\mathcal{H} = \{a \cdot \bar{i} + F \cdot \bar{v} | a \in \mathbb{Q}^+, \bar{v} \in (\mathbb{Q}^T)^P \} \cap (\mathbb{Q}^P)^P = (A + B) \cap \{\bar{i}, \bar{f}, \bar{a}, \bar{b}\}$,
  $A = \{\bar{i}\}$ and $B = \{\pm(3 \cdot \bar{a} + \bar{b} - \bar{i}), \pm(\bar{a} + \bar{b}), \pm(\bar{i} - \bar{a} - 3 \cdot \bar{b})\}$
- $E = \{\bar{i}, \bar{f}, \bar{a}, \bar{b}\}$
- $E_G = \{\bar{i}, \bar{f}, 8 \cdot \bar{a}, 8 \cdot \bar{b}\}$
- $|\Gamma| = 44$

$\Gamma$ is very large
Reducing the number of proper termination checks

Lemma

\[ m > m', \ m \in G_i, \ m' \in G_j \implies i > j \]

Theorem

Let \( \Upsilon \) is the set of minimal markings of \( G^+ = G - G_0 \). Then:

1. \( N \) is generalized sound iff every marking \( m \in \Upsilon \) terminates properly.
2. Each marking \( m \in \Upsilon \) satisfies \( m \leq (\max_i \{ e_i^1 \}, \ldots, \max_i \{ e_i^{|P|} \}) \).
3. \( \Upsilon \subseteq \Gamma \).
Reducing the number of proper termination checks

Lemma

\[ m > m', \ m \in \mathcal{G}_i, \ m' \in \mathcal{G}_j \implies i > j \]

Theorem

Let \( \Upsilon \) is the set of minimal markings of \( \mathcal{G}^+ = \mathcal{G} - \mathcal{G}_0 \). Then:

1. \( \mathcal{N} \) is generalized sound iff every marking \( m \in \Upsilon \) terminates properly.
2. Each marking \( m \in \Upsilon \) satisfies \( m \leq (\max_i\{e_i^1\}, \ldots, \max_i\{e_i^{|P|}\}) \).
3. \( \Upsilon \subseteq \Gamma \).
New Decision Procedure

- Check whether $\mathcal{I} \cdot \overline{i} = \mathcal{I} \cdot \overline{f}$
- Check whether $\mathcal{I} \cdot x = \overline{0}$ for $x \in (\mathbb{Q}^+)^P$ has only trivial solution
- Check proper termination for a finite minimal set of markings of $\mathcal{G}$:
  1. Find a set of generators $E$ of the polyhedral cone $\mathcal{H}$
  2. Compute the set of rescaled generators $- E_G$
  3. Find a set of minimal markings $\Upsilon$ of $\mathcal{G}$:
     $$\Upsilon = \min \{ m | m \in G^+ \land m \leq M \}$$
     where $M = (\max_i \{ e^i_1 \}, \ldots, \max_i \{ e^i_{|P|} \})$
  4. Check proper termination for all markings of $\Upsilon$ using a backward reachability algorithm
Backward reachability check

**Input:*** \( N = (P, T, F), \Upsilon, J = \{ w(m) \mid m \in \Upsilon \} \)

**Output:*** “the BWF-net is sound” or “the BWF-net is not sound, \( m, k \)” where \( m \in \mathcal{G}_k \), \( m \xrightarrow{*} k \cdot \bar{f} \) and \( k = \min \{ \ell \mid m \in \Upsilon : \ell \cdot \bar{i} \xrightarrow{\sigma} m \xrightarrow{*} \ell \cdot \bar{f} \} \)

for \( j \in J \) do

\[ B_j = \{ j \cdot \bar{f} \} \]

repeat

\[ B_j = B_j \cup \{ m - F_t \mid \forall p \in P : m(p) \geq F(p, t) \land m \in B_j \land t \in T \} \]

until a fixpoint is reached or \( \Upsilon_j \subseteq B_j \)

if \( \Upsilon_j \not\subseteq B_j \) then

pick \( m \in \Upsilon_j \setminus B_j; \ell = 1 \)

loop

if \( (j + \ell) \cdot \bar{i} \in B_{j+\ell} \) then

return (“the BWF-net is not sound”, \( m, j + \ell \) )

end

else \( \ell++ \)

loop

end

end

return (“the BWF-net is sound”)
Backward reachability check

**Input:** \( N = (P, T, F), \gamma, J = \{w(m) \mid m \in \gamma\} \)

**Output:** “the BWF-net is sound” or “the BWF-net is not sound, \( m, k \)” where \( m \in g_k, m \not\to k \cdot \bar{f} \) and \( k = \min\{\ell \mid m \in \gamma : \ell \cdot \bar{i} \not\to m \not\to \ell \cdot \bar{f}\} \)

for \( j \in J \) do
\[
B_j = \{j \cdot \bar{f}\}; \\
\text{repeat} \\
\quad B_j = B_j \cup \{m - F_t \mid \forall p \in P : m(p) \geq F(p, t) \land m \in B_j \land t \in T\} \\
\text{until a fixpoint is reached or } \gamma_j \not\subseteq B_j; \\
\text{if } \gamma_j \not\subseteq B_j \text{ then} \\
\quad \text{pick } m \in \gamma_j \setminus B_j; \ell = 1; \\
\text{loop} \\
\quad \text{if } (j + \ell) \cdot \bar{i} \in B_{j+\ell} \text{ then} \\
\quad\quad \text{return}(“the BWF-net is not sound”, \( m, j + \ell \)) \\
\quad\quad \text{end} \\
\quad\text{else } \ell++ \\
\text{loop} \\
\text{end} \\
\text{return}(“the BWF-net is sound”)
Backward reachability check

**Input:** \( N = (P, T, F), \gamma, J = \{w(m) \mid m \in \gamma\} \)

**Output:** “the BWF-net is sound” or “the BWF-net is not sound, \( m, k \)” where \( m \in g_k, m \not\rightarrow^* k \cdot \bar{f} \) and \( k = \min\{\ell \mid m \in \gamma : \ell \cdot \bar{i} \not\rightarrow m \not\rightarrow^* \ell \cdot \bar{f}\} \)

for \( j \in J \) do
  \( B_j = \{j \cdot \bar{f}\}; \)
  repeat
    \( B_j = B_j \cup \{m - F_t \mid \forall p \in P : m(p) \geq F(p, t) \land m \in B_j \land t \in T\} \)
  until a fixpoint is reached or \( \gamma_j \subseteq B_j \);
  if \( \gamma_j \not\subseteq B_j \) then
    pick \( m \in \gamma_j \setminus B_j; \ell = 1; \)
    loop
      if \( (j + \ell) \cdot \bar{i} \in B_{j+\ell} \) then
        return(“the BWF-net is not sound”, \( m, j + \ell \) )
      end
    else \( \ell++ \)
    loop
  end
return(“the BWF-net is sound”)
Example

\[ E = \{ \bar{i}, \bar{f}, \bar{a}, \bar{b} \} \]
\[ E_G = \{ \bar{i}, \bar{f}, 8 \cdot \bar{a}, 8 \cdot \bar{b} \} \]

\[ \Upsilon = \{ 8 \cdot \bar{a}, 8 \cdot \bar{b}, \bar{a} + 3 \cdot \bar{b}, 3 \cdot \bar{a} + \bar{b}, \bar{i}, \bar{f} \} \]

\[ |\Upsilon| = 6; |\Gamma| = 44 \]

- \( 8 \cdot \bar{b} \in \mathcal{R}(2 \cdot \bar{i}) \)
- \( 8 \cdot \bar{b} \not\rightarrow 2 \cdot \bar{f} \)
Example

\[ E = \{ \bar{i}, \bar{f}, \bar{a}, \bar{b} \} \]
\[ E_G = \{ \bar{i}, \bar{f}, 8 \cdot \bar{a}, 8 \cdot \bar{b} \} \]

\[ \Upsilon = \{ 8 \cdot \bar{a}, 8 \cdot \bar{b}, \bar{a} + 3 \cdot \bar{b}, 3 \cdot \bar{a} + \bar{b}, \bar{i}, \bar{f} \} \]

\[ |\Upsilon| = 6; |\Gamma| = 44 \]

- \[ 8 \cdot \bar{b} \in \mathcal{R}(2 \cdot \bar{i}) \]
- \[ 8 \cdot \bar{b} \not\rightarrow 2 \cdot \bar{f} \]
Example

\[ E = \{ \bar{i}, \bar{f}, \bar{a}, \bar{b} \} \]
\[ E_G = \{ \bar{i}, \bar{f}, 8 \cdot \bar{a}, 8 \cdot \bar{b} \} \]

\[ \Upsilon = \{ 8 \cdot \bar{a}, 8 \cdot \bar{b}, \bar{a} + 3 \cdot \bar{b}, 3 \cdot \bar{a} + \bar{b}, \bar{i}, \bar{f} \} \]

\[ |\Upsilon| = 6; \quad |\Gamma| = 44 \]

- \[ 8 \cdot \bar{b} \in \mathcal{R}(2 \cdot \bar{i}) \]
- \[ 8 \cdot \bar{b} \not\rightarrow 2 \cdot \bar{f} \]
Example

\[ E = \{ \bar{i}, \bar{f}, \bar{a}, \bar{b} \} \]
\[ E_G = \{ \bar{i}, \bar{f}, 8 \cdot \bar{a}, 8 \cdot \bar{b} \} \]

\[ \Upsilon = \{ 8 \cdot \bar{a}, 8 \cdot \bar{b}, \bar{a} + 3 \cdot \bar{b}, 3 \cdot \bar{a} + \bar{b}, \bar{i}, \bar{f} \} \]

\[ |\Upsilon| = 6; \ |\Gamma| = 44 \]

- \[ 8 \cdot \bar{b} \in \mathcal{R}(2 \cdot \bar{i}) \]
- \[ 8 \cdot \bar{b} \not\rightarrow 2 \cdot \bar{f} \]
Example

\[ E = \{ \bar{i}, \bar{f}, \bar{a}, \bar{b} \} \]
\[ E_G = \{ \bar{i}, \bar{f}, 8 \cdot \bar{a}, 8 \cdot \bar{b} \} \]

\[ \Upsilon = \{ 8 \cdot \bar{a}, 8 \cdot \bar{b}, \bar{a} + 3 \cdot \bar{b}, 3 \cdot \bar{a} + \bar{b}, \bar{i}, \bar{f} \} \]

\[ |\Upsilon| = 6; \ |\Gamma| = 44 \]

- \[ 8 \cdot \bar{b} \in \mathcal{R}(2 \cdot \bar{i}) \]
- \[ 8 \cdot \bar{b} \not\rightarrow 2 \cdot \bar{f} \]
Example

\[ E = \{ \bar{i}, \bar{f}, \bar{a}, \bar{b} \} \]
\[ E_G = \{ \bar{i}, \bar{f}, 8 \cdot \bar{a}, 8 \cdot \bar{b} \} \]

\[ \Upsilon = \{ 8 \cdot \bar{a}, 8 \cdot \bar{b}, \bar{a} + 3 \cdot \bar{b}, 3 \cdot \bar{a} + \bar{b}, \bar{i}, \bar{f} \} \]

\[ |\Upsilon| = 6; \quad |\Gamma| = 44 \]

- \( 8 \cdot \bar{b} \in \mathcal{R}(2 \cdot \bar{i}) \)
- \( 8 \cdot \bar{b} \not\rightarrow 2 \cdot \bar{f} \)
### Implementation and experimental results

Parma Polyhedra Library - PPL for finding $\gamma$

### Results

| File name | Description | $|P|/|T|$ | Size($\gamma$) | Time       |
|-----------|-------------|---------|----------------|------------|
| consm     | sound       | 23/27   | $75 (\gamma = E_g)$ | 19909 ms   |
| smwf      | sound       | 18/22   | $70 (\gamma = E_g)$ | 8005 ms    |
| ref       | sound       | 12/12   | $14 (\gamma = E_g)$ | 131 ms     |
| smp       | sound       | 9/10    | trivial ($\gamma = E_g$) | 16 ms      |
| soundm    | sound       | 9/9     | $10 (\gamma = E_g)$ | 26 ms      |
| snotws    | sound       | 7/8     | $7 (\gamma = E_g)$ | 9 ms       |
| snet      | sound       | 9/6     | $10 (\gamma = E_g)$ | 48 ms      |
| sound     | sound       | 6/6     | $6 (\gamma = E_g)$ | 9 ms       |
| fcs       | not sound   | 7/5     | $3 (\gamma = E_g)$ | 5 ms       |
| snet2     | 1 sound     | 5/6     | trivial ($\gamma = E_g$) | 5 ms      |
| soundp    | sound       | 5/5     | 6               | 7 ms       |
| exn2      | 1 not 2-sound | 4/3   | 4                | 8 ms       |
We give an improved procedure for verifying generalized soundness that:

- reduces the number of proper termination checks
- gives a counterexample in case the net is not sound

Future work:

- optimize the algorithm
- investigate the use of the algorithm for checking soundness in a compositional way
- verification of temporal logic properties of Petri nets (not necessarily WF-nets) using such a reduction technique
- build sound by construction nets in a hierarchical manner