Verifying Generalized Soundness for Workflow Nets

Kees van Hee, Olivia Oanea, Natalia Sidorova and Marc Voorhoeve

Department of Mathematics and Computer Science
Technical University Eindhoven
The Netherlands

PSI 2006
A Petri net \( N = (P, T, F) \) is a \textit{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 \( f \).
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 $\cdot i = \emptyset$, 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 problem for Workflow nets

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 $f$. 

![Workflow net diagram]
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 $f$. 

**Proper termination (1-soundness)**
A WF-net $N$ is generalized sound iff for all $k \in \mathbb{N}$, all markings reachable from $k \cdot \bar{f}$ terminate properly, i.e. $m \rightarrow^* k \cdot \bar{f}$. 
Generalized soundness problem for Workflow nets

A WF-net \( N \) is generalized sound iff for all \( k \in \mathbb{N} \), all markings reachable from \( k \cdot \tilde{f} \) terminate properly, i.e. \( m \xrightarrow{*} k \cdot \tilde{f} \).
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
Outline

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


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 $N'$.
Trap
A subset of places $Q$ is called a *trap* if $Q^* \subseteq \bullet Q$.

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

**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 \( m' = m + F \cdot \vec{\sigma} \)

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. \( \mathcal{I} \cdot \vec{i} = \mathcal{I} \cdot \vec{f} \) since \( \vec{i} \xrightarrow{*} \vec{f} \)

2. \( \mathcal{I} \cdot x = \vec{0} \) has only the trivial solution on \( \mathbb{N}^P \).
   otherwise if \( x > \vec{0} \Rightarrow x \xrightarrow{*} \vec{0} \) — false since \( t^\bullet \neq \emptyset \) for all \( t \)

### Generalized soundness \(\Leftrightarrow\) proper termination of

- \( \mathcal{R} = \bigcup_{k \in \mathbb{N}} \mathcal{R}(k \cdot \vec{i}) = \bigcup_{k \in \mathbb{N}} \{ k \cdot \vec{i} + F \cdot \vec{v} | \vec{v} \in \mathbb{N}^T \} \cap \mathbb{N}^P \)

- \( \mathcal{G} = \bigcup_{k \in \mathbb{N}} \mathcal{G}_k \), where \( \mathcal{G}_k = \{ k \cdot \vec{i} + F \cdot \vec{v} | \vec{v} \in \mathbb{Z}^T \} \cap \mathbb{N}^P \)
  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 $\Gamma \subseteq \mathcal{G}$

- $\mathcal{H} = \{ a \cdot \vec{i} + F \cdot \vec{v} | a \in \mathbb{Q}^+, \, \vec{v} \in \mathbb{Q}^T \} \cap (\mathbb{Q}^+)^P$ 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 $\mathcal{I} \cdot \vec{i} = \mathcal{I} \cdot \vec{f}$
2. Check whether $\mathcal{I} \cdot \vec{x} = \vec{0}$ 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 $\iff$ 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 $\Gamma$
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 $x = \bar{0}$
- $\mathcal{H} = \{ a \cdot \bar{i} + F \cdot \nu \mid a \in \mathbb{Q}^{+}, \nu \in \mathbb{Q}^{T} \} \cap (\mathbb{Q}^{+})^{P} = (A + B) \cap \{ \bar{i}, \bar{f}, \bar{a}, \bar{b} \}$, $A = \{ \bar{i} \}$ 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
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 $B = \{ \pm(3 \cdot \vec{a} + \vec{b} - \vec{i}), \pm(\vec{a} + \vec{b}), \pm(\vec{i} - \vec{a} - 3 \cdot \vec{b}) \}$

$E = \{ \vec{i}, \vec{f}, \vec{a}, \vec{b} \}$

$E_G = \{ \vec{i}, \vec{f}, 8 \cdot \vec{a}, 8 \cdot \vec{b} \}$

$|\Gamma| = 44$
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 \( m \leq (\max_i \{ e_i \}, \ldots, \max_i \{ e_{P_i} \}) \).
3. \( \Upsilon \subseteq \Gamma \).
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. \( \Upsilon \subseteq \Gamma \).
New Decision Procedure

- Check whether $\mathcal{I} \cdot \vec{i} = \mathcal{I} \cdot \vec{f}$
- Check whether $\mathcal{I} \cdot \vec{x} = \vec{0}$ for $\vec{x} \in (\mathbb{Q}^+)^P$ 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 $\mathcal{G}$:
     \[
     \Upsilon = \min\{m | m \in \mathcal{G}^+ \land m \leq M\}
     \]
     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 \( m \in G_k \), \( m \xrightarrow{\ast} k \cdot \bar{f} \) and \( k = \min\{\ell \mid m \in \Upsilon: \ell \cdot i \overset{\sigma}{\rightarrow} m \not\xrightarrow{\ast} \ell \cdot \bar{f}\} \)

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
return(“the BWF-net is sound”)
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

\[
m \in G_k, \quad m \not\rightarrow^* k \cdot \bar{f} \quad \text{and} \quad k = \min\{\ell \mid m \in \Upsilon : \ell \cdot i \not\rightarrow m \not\rightarrow \ell \cdot \bar{f}\}
\]

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 \( \Upsilon_j \subseteq B_j \);

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

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

loop

| if \( (j + \ell) \cdot i \in B_{j+\ell} \) 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:** \( N = (P, T, F), \) \( \Upsilon, J = \{ w(m) \mid m \in \Upsilon \} \)

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

**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 \( \Upsilon_j \subseteq B_j \);

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

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

**loop**

\[
\text{if } (j + \ell) \cdot \bar{i} \in B_{j+\ell} \text{ then return("the BWF-net is not sound", } m, j + \ell) \]

**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

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

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

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

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

$E = \{\bar{i}, \bar{f}, \bar{a}, \bar{b}\}$

$E_G = \{\bar{i}, \bar{f}, 8 \cdot \bar{a}, 8 \cdot \bar{b}\}$

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

$|\Upsilon| = 6; |\Gamma| = 44$

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

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

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

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

- \( 8 \cdot \bar{b} \in \mathcal{R}(2 \cdot \bar{i}) \)
- \( 8 \cdot \bar{b} \not\rightarrow 2 \cdot \bar{f} \)
Parma Polyhedra Library - PPL for finding \( \Upsilon \)

### Results

| File name | Description     | \(|P|/|T| | Size(\(\Upsilon\)) | Time     |
|-----------|-----------------|-----------------|-------------------|----------|
| consm     | sound           | 23/27           | 75 (\(\Upsilon = E_g\)) | 19909 ms |
| smwf      | sound           | 18/22           | 70 (\(\Upsilon = E_g\)) | 8005 ms  |
| ref       | sound           | 12/12           | 14 (\(\Upsilon = E_g\)) | 131 ms   |
| smp       | sound           | 9/10            | 9 (\(\Upsilon = E_g\))  | 16 ms    |
| soundm    | sound           | 9/9             | 10 (\(\Upsilon = E_g\)) | 26 ms    |
| snotws    | sound           | 7/8             | 7 (\(\Upsilon = E_g\))  | 9 ms     |
| snet      | sound           | 9/6             | 10 (\(\Upsilon = E_g\)) | 48 ms    |
| sound     | sound           | 6/6             | 6 (\(\Upsilon = E_g\))  | 9 ms     |
| fcs       | sound           | 7/5             | 6 (\(\Upsilon = E_g\))  | 5 ms     |
| snet2     | 1 not 2-sound   | 5/6             | 5 (\(\Upsilon = E_g\))  | 5 ms     |
| soundp    | sound           | 5/5             | 6                  | 7 ms     |
| exn2      | 1 not 2-sound   | 4/3             | 6                  | 8 ms     |
Conclusions and Future work

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