Automating Soundness Proofs

Muck van Weerdenburg

SOS'08 July 6, 2008

Outline

- Introduction
  - The(/My) Problem
  - Process Language Development
  - Typical Soundness Proofs
- Automation
  - Approach
  - Translation to FOL
  - Proving
  - Proof of Concept
- Future Work
  - Universal Quantification
  - Additional Logic Rules

The(/My) Problem

In the last 5 years I made several process languages.

Each time, the same tasks have to be done.

Some task are “essential”; require actual thinking.

But some are tedious and straightforward (boring).

Process Language Development - Syntax

For illustration we use a simple process language.

CCS without parallelism:

- deadlock 0
- action prefix $a.\_$
- alternative composition $\_ + \_$

Process Language Development - SOS

$$\frac{}{a.p \xrightarrow{a} p}$$

$$\frac{p \xrightarrow{a} p'}{p + q \xrightarrow{a} p'} \qquad \frac{q \xrightarrow{a} q'}{p + q \xrightarrow{a} q'}$$

Process Language Development - Relation

We say to processes p and q are equivalent if...

...there is a relation R relating p and q...

such that if $p' R q'$, then

- $q' R p'$, and
- forall a and p'' with $p' \xrightarrow{a} p''$, there is a q'' with $q' \xrightarrow{a} q''$ and $p'' R q''$

We write $p \underline{\leftrightarrow} q$ iff p and q are equivalent.

Process Language Development - Equalities (Axioms)

We think we have the following equalities between processes.

$$x + y \;\underline{\leftrightarrow}\; y + x$$

$$x + (y + z) \;\underline{\leftrightarrow}\; (x + y) + z$$

$$x + x \;\underline{\leftrightarrow}\; x$$

$$x + 0 \;\underline{\leftrightarrow}\; x$$

Typical Soundness Proofs - Relation (revisited)

We say to processes p and q are equivalent if...

...there is a relation R relating p and q...

such that if $p' \; R \; q'$, then

- $q' \; R \; p'$, and
- forall a and p'' with $p' \xrightarrow{a} p''$, there is a q'' with $q' \xrightarrow{a} q''$ and $p'' \; R \; q''$

We write $p \underline{\leftrightarrow} q$ iff p and q are equivalent.

Typical Soundness Proofs

A soundness proof typically follows the following lines.

We first define a relation that should be the witness of the equality.

For $x + 0 \underline{\leftrightarrow} x$ this could be:

$$R = \{\langle p+0, p\rangle, \langle p, p+0\rangle, \langle p, p\rangle \ : \ \mathrm{true}\}$$

Typical Soundness Proofs

Assume p and q with $p\ R\ q$.

This means there is a r such that:

- $p = r + 0$ and $q = r$, or
- $p = r$ and $q = r + 0$, or
- $p = r$ and $q = r$

Let us consider the first case.

Typical Soundness Proofs

We have $p = r + 0$ and $q = r$.

Second transfer condition says:

Assume a and p' such that $p \xrightarrow{a} p'$.

This means:

- $r \xrightarrow{a} p'$, or
- $0 \xrightarrow{a} p'$

Again, let us consider the first case.

Typical Soundness Proofs

We have $p = r + 0$, $q = r$ and $r \xrightarrow{a} p'$.

We must find a q' such that $q \xrightarrow{a} q'$ and $p' \; R \; q'$.

As $q = r$ and we have $r \xrightarrow{a} p'$, take $q' = p'$.

Etc...

Typical Soundness Proofs

These proofs (almost) always follow these lines:

- Deconstruct assumptions.
- Construct desired conclusions.

Very little intelligence is required in this process.

We (have to) do these proofs again and again for each new theory/language.

Outline

Approach of Automation

We want to translate the problem to first-order logic...

...and use a prover to solve it.

(All automatically.)

Translation to FOL - SOS

We assume all rules have a conclusion of the form $P(f(\ldots),\ldots)$.

Then we simply interpret a rule $\dfrac{P_1,\ldots,P_N}{Q}$ as

$$\forall_{x_1,x_2,\ldots}(P_1 \wedge \ldots \wedge P_n \Rightarrow Q)$$

(This requires a complete/well-defined specification.)

Then we can easily define, for each P and f, a definition for $P(f(\ldots),\ldots)$.

Translation to FOL - SOS (revisited)

$$\frac{}{a.p \xrightarrow{a} p}$$

$$\frac{p \xrightarrow{a} p'}{p + q \xrightarrow{a} p'} \qquad \frac{q \xrightarrow{a} q'}{p + q \xrightarrow{a} q'}$$

Translation to FOL - SOS

In our example this means we get the following:

$$0 \xrightarrow{a} x \stackrel{def}{=} \text{false}$$

$$a.x \xrightarrow{a} y \stackrel{def}{=} \exists_p (x = p \ \wedge \ y = p \ \wedge \ \text{true})$$

$$\begin{aligned} x + y \xrightarrow{a} z \stackrel{def}{=} & \ \exists_{p,p',q} (x = p \ \wedge \ y = q \ \wedge \ z = p' \ \wedge \ p \xrightarrow{a} p') \\ \vee & \ \exists_{p,q,q'} (x = p \ \wedge \ y = q \ \wedge \ z = q' \ \wedge \ q \xrightarrow{a} q') \end{aligned}$$

Translation to FOL - Relation

To formulate the relation we get the following:

$$
\begin{aligned}
\mathrm{is\_rel}(R) \stackrel{def}{=} \forall_{p,q}(R(p,q) \Rightarrow\ & \\
& R(q,p) \wedge \\
& \forall a, p'(p \xrightarrow{a} p' \Rightarrow \exists q'(q \xrightarrow{a} q' \ \wedge\ R(p',q'))))
\end{aligned}
$$

(Note this is not quite first-order, but can easily be formulated as such.)

Translation to FOL - Equalities

An equality e such as $x + 0 \underline{\leftrightarrow} x$ is represented by:

$$\begin{array}{rcl} R_e(x, y) & \stackrel{def}{=} & \exists z(x = z + 0 \wedge y = z) \\ & \vee & \exists z(x = z \wedge y = z + 0) \\ & \vee & \exists z(x = z \wedge y = z) \end{array}$$

(This represents $\{\langle p+0, p\rangle, \langle p, p+0\rangle, \langle p, p\rangle \ : \ \mathrm{true}\}$.)

Proving

To prove the soundness of $x + 0 \underline{\leftrightarrow} x$...

...we take all previous definition as axioms in a logic system...

..and construct a proof for $\mathrm{is_rel}(R_e)$.

Proving

We use a standard sequent logic with the rule.

Except that we replace

$$\frac{\Gamma \vDash \varphi[t/x], \Delta}{\Gamma \vDash \exists_x(\varphi), \Delta} \quad \text{with} \quad \frac{\Gamma \vDash \varphi[X/x], \Delta}{\Gamma \vDash \exists_x(\varphi), \Delta}$$

where X is a meta-variable.

This allows delay of specific instantiation.

Proving

In proving $x + y \leftrightarrow y + x$:

$$\dfrac{\dfrac{\dfrac{?}{x \xrightarrow{a} x' \models y \xrightarrow{a} Z, x \xrightarrow{a} Z}}{x \xrightarrow{a} x' \models y + x \xrightarrow{a} Z} \qquad \dfrac{?}{x \xrightarrow{a} x' \models x' \; R \; Z}}{\dfrac{x \xrightarrow{a} x' \models y + x \xrightarrow{a} Z \;\wedge\; x' \; R \; Z}{x \xrightarrow{a} x' \models \exists_z (y + x \xrightarrow{a} z \;\wedge\; x' \; R \; z)}}$$

Proving

In proving $x + y \leftrightarroweq y + x$:

$$\dfrac{\dfrac{\dfrac{}{x \xrightarrow{a} x' \models y \xrightarrow{a} x', x \xrightarrow{a} x'}}{x \xrightarrow{a} x' \models y + x \xrightarrow{a} x'} \qquad \dfrac{\cdots}{x \xrightarrow{a} x' \models x' \; R \; x'}}{\dfrac{x \xrightarrow{a} x' \models y + x \xrightarrow{a} x' \;\wedge\; x' \; R \; x'}{x \xrightarrow{a} x' \models \exists_z (y + x \xrightarrow{a} z \;\wedge\; x' \; R \; z)}}$$

Proof of Concept

With this method we proved soundess of axiomatisations of

- CCS (without parallelism),
- $BPA_{\delta\epsilon}$ (has termination predicate),
- BPA^* (iteration),
- ACP (parallelism).

Prototype at http://www.win.tue.nl/˜mweerden/soundness/.

Outline

Universal Quantification

Universal quantification in assumptions:

$$\frac{\Gamma, \forall_x(\varphi), \varphi[t/x] \vDash \Delta}{\Gamma, \forall_x(\varphi) \vDash \Delta}$$

You have to be smart about the values you want to instantiate.

With this: proofs for symmetry, transitivity and congruence?

Additional Logic Rules

Discrete-time languages (with time-transition $\mapsto$):

$$\frac{x \mapsto x', y \mapsto y'}{x + y \mapsto x' + y'}$$

Proving $x + x = x$ results in:

$$x \mapsto x', x \mapsto y' \vDash x \mapsto Z \ \wedge \ x' + y' \ R \ Z$$

Here you could use a rule like $\frac{y = z}{x \mapsto y, x \mapsto z}$.

Additional Logic Rules

Rules for substitution.

Summation:

$$\frac{p[t/x] \xrightarrow{a} p'}{\Sigma_x p \xrightarrow{a} p'}$$

Recursion:

$$\frac{p[\mu X.p/X] \xrightarrow{a} p'}{\mu X.p \xrightarrow{a} p'}$$

Extensions

Proof generation.

Automatic relation expansion.

Induction.

...

Thank you for your attention!