A formalization of the spi calculus in Coq

Sébastien Briais

École Normale Supérieure de Lyon

INRIA-Microsoft Research
2007, November 29th
Orsay, FRANCE

Plan

Plan

The spi calculus

- Extension of the pi calculus that incorporates cryptographic messages [AG98]
- To model and study cryptographic protocols.

Syntax

- Countably infinite set of *names*.
  Communication channels, nonces, atomic data, ...
- Messages

$$M, N \quad ::= \quad x \mid (M \,.\, N) \mid \mathrm{Enc}^{\mathrm{s}}_{N} M$$

- Expressions

$$\begin{aligned} E, F \quad ::= \quad & x \mid (E \,.\, F) \mid \mathrm{Enc}^{\mathrm{s}}_{F} E \\ \mid \quad & \pi_1 (E) \mid \pi_2 (E) \mid \mathrm{Dec}^{\mathrm{s}}_{F} E \end{aligned}$$

- Guards

$$\phi \quad ::= \quad [E = F] \mid [E : \boldsymbol{N}]$$

Syntax (continued)

- Processes

$$\begin{array}{rcl} P, Q & ::= & \mathbf{0} \mid E(x).P \mid \overline{E}\langle F \rangle.P \\ & \mid & \phi P \mid (\nu x)\, P \\ & \mid & P \mid Q \mid P + Q \mid !\, P \end{array}$$

Syntax (continued)

- Processes

$$\begin{array}{rcl} P, Q & ::= & \mathbf{0} \mid E(x).P \mid \overline{E}\langle F \rangle.P \\ & \mid & \phi P \mid (\nu x)\, P \\ & \mid & P \mid Q \mid P + Q \mid !\, P \end{array}$$

Late LTS

$$\textsc{Input}\ \frac{}{E(x).P}$$

Evaluation of expressions and guards

- Expressions :

$$
\begin{array}{rcll}
\mathbf{e}_c(a) & := & a & \\
\mathbf{e}_c(\mathrm{Enc}^s_F E) & := & \mathrm{Enc}^s_N M & \text{if } \mathbf{e}_c(E) = M \in \boldsymbol{M} \\
 & & & \text{and } \mathbf{e}_c(F) = N \in \boldsymbol{M} \\
\mathbf{e}_c((E_1 \,.\, E_2)) & := & (M_1 \,.\, M_2) & \text{if } \mathbf{e}_c(E_1) = M_1 \in \boldsymbol{M} \\
 & & & \text{and } \mathbf{e}_c(E_2) = M_2 \in \boldsymbol{M} \\
\mathbf{e}_c(\mathrm{Dec}^s_F E) & := & M & \text{if } \mathbf{e}_c(E) = \mathrm{Enc}^s_N M \in \boldsymbol{M} \\
 & & & \text{and } \mathbf{e}_c(F) = N \in \boldsymbol{M} \\
\mathbf{e}_c(\pi_1\,(E)) & := & M_1 & \text{if } \mathbf{e}_c(E) = (M_1 \,.\, M_2) \in \boldsymbol{M} \\
\mathbf{e}_c(\pi_2\,(E)) & := & M_2 & \text{if } \mathbf{e}_c(E) = (M_1 \,.\, M_2) \in \boldsymbol{M} \\
\mathbf{e}_c(E) & := & \bot & \text{otherwise}
\end{array}
$$

Late LTS

$$\textsc{Input}\ \frac{\mathbf{e}_c(E) = a \in \boldsymbol{N}}{E(x).P}$$

Late LTS

$$\textsc{Input}\ \frac{\mathbf{e}_c(E) = a \in \boldsymbol{N}}{E(x).P \xrightarrow{a}}$$

Syntax (continued)

- Processes

$$\begin{array}{rcl} P, Q & ::= & \mathbf{0} \mid E(x).P \mid \overline{E}\langle F\rangle.P \\ & \mid & \phi P \mid (\nu x)\, P \\ & \mid & P \mid Q \mid P + Q \mid !\, P \end{array}$$

- Agents

$$\begin{array}{rcl} A & ::= & P \\ & \mid & (x)P \\ & \mid & (\nu \tilde{z})\, \langle M\rangle P \quad \text{where } \{\tilde{z}\} \subseteq \mathsf{n}(M) \end{array}$$

Syntax (continued)

- Processes

$$\begin{array}{rcl} P, Q & ::= & \mathbf{0} \mid E(x).P \mid \overline{E}\langle F \rangle.P \\ & \mid & \phi P \mid (\nu x)\, P \\ & \mid & P \mid Q \mid P + Q \mid !\, P \end{array}$$

- Agents

$$\begin{array}{rcl} A & ::= & P \\ & \mid & (x)P \\ & \mid & (\nu \tilde{z})\, \langle M \rangle P \quad \text{where } \{\tilde{z}\} \subseteq \mathsf{n}(M) \end{array}$$

Late LTS

$$\textsc{Input}\ \frac{\mathbf{e}_c(E) = a \in \boldsymbol{N}}{E(x).P \xrightarrow{a} (x)P}$$

Late LTS

$$\textsc{Input}\ \frac{\mathbf{e}_c(E) = a \in \boldsymbol{N}}{E(x).P \xrightarrow{a} (x)P} \qquad \textsc{Output}\ \frac{\mathbf{e}_c(E) = a \in \boldsymbol{N} \qquad \mathbf{e}_c(F) = M \in \boldsymbol{M}}{\overline{E}\langle F\rangle.P \xrightarrow{\overline{a}} \langle M\rangle P}$$

Late LTS

$$\textsc{Input}\ \frac{\mathbf{e}_c(E) = a \in \boldsymbol{N}}{E(x).P \xrightarrow{a} (x)P} \qquad \textsc{Output}\ \frac{\mathbf{e}_c(E) = a \in \boldsymbol{N} \qquad \mathbf{e}_c(F) = M \in \boldsymbol{M}}{\overline{E}\langle F\rangle.P \xrightarrow{\overline{a}} \langle M\rangle P}$$

$$\textsc{Close-l}\ \frac{P \xrightarrow{a} F \qquad Q \xrightarrow{\overline{a}} C}{P \mid Q \xrightarrow{\tau}}$$

Operations on agents

- Pseudo-application :
  If $F = (x)P$ is an abstraction and
  $C = (\nu \tilde{z})\,\langle M \rangle Q$ is a concretion (with $\{\tilde{z}\} \cap \mathrm{fn}(P) = \emptyset$),
  then $F \bullet C := (\nu \tilde{z})\,(P\{^{M}/_{x}\} \mid Q)$

Late LTS

$$\textsc{Input}\ \frac{\mathbf{e}_c(E) = a \in \boldsymbol{N}}{E(x).P \xrightarrow{a} (x)P} \qquad \textsc{Output}\ \frac{\mathbf{e}_c(E) = a \in \boldsymbol{N} \qquad \mathbf{e}_c(F) = M \in \boldsymbol{M}}{\overline{E}\langle F\rangle.P \xrightarrow{\overline{a}} \langle M\rangle P}$$

$$\textsc{Close-l}\ \frac{P \xrightarrow{a} F \qquad Q \xrightarrow{\overline{a}} C}{P \mid Q \xrightarrow{\tau} F \bullet C}$$

Late LTS

$$\textsc{Input}\ \frac{\mathbf{e}_c(E) = a \in \boldsymbol{N}}{E(x).P \xrightarrow{a} (x)P} \qquad \textsc{Output}\ \frac{\mathbf{e}_c(E) = a \in \boldsymbol{N} \qquad \mathbf{e}_c(F) = M \in \boldsymbol{M}}{\overline{E}\langle F\rangle.P \xrightarrow{\overline{a}} \langle M\rangle P}$$

$$\textsc{Close-L}\ \frac{P \xrightarrow{a} F \qquad Q \xrightarrow{\overline{a}} C}{P \mid Q \xrightarrow{\tau} F \bullet C} \qquad \textsc{IfThen}\ \frac{P \xrightarrow{\mu} P'}{\phi P \xrightarrow{\mu} P'}$$

Evaluation of expressions and guards

- Expressions :

$$
\begin{array}{rcll}
\mathbf{e}_c(a) & := & a & \\
\mathbf{e}_c(\mathrm{Enc}^s_F E) & := & \mathrm{Enc}^s_N M & \text{if } \mathbf{e}_c(E) = M \in \boldsymbol{M} \\
 & & & \text{and } \mathbf{e}_c(F) = N \in \boldsymbol{M} \\
\mathbf{e}_c((E_1 \,.\, E_2)) & := & (M_1 \,.\, M_2) & \text{if } \mathbf{e}_c(E_1) = M_1 \in \boldsymbol{M} \\
 & & & \text{and } \mathbf{e}_c(E_2) = M_2 \in \boldsymbol{M} \\
\mathbf{e}_c(\mathrm{Dec}^s_F E) & := & M & \text{if } \mathbf{e}_c(E) = \mathrm{Enc}^s_N M \in \boldsymbol{M} \\
 & & & \text{and } \mathbf{e}_c(F) = N \in \boldsymbol{M} \\
\mathbf{e}_c(\pi_1\,(E)) & := & M_1 & \text{if } \mathbf{e}_c(E) = (M_1 \,.\, M_2) \in \boldsymbol{M} \\
\mathbf{e}_c(\pi_2\,(E)) & := & M_2 & \text{if } \mathbf{e}_c(E) = (M_1 \,.\, M_2) \in \boldsymbol{M} \\
\mathbf{e}_c(E) & := & \bot & \text{otherwise}
\end{array}
$$

- Guards :

$$
\begin{array}{rcll}
\mathbf{e}([E = F]) & := & \textbf{true} & \text{si } \mathbf{e}_c(E) = \mathbf{e}_c(F) = M \in \boldsymbol{M} \\
\mathbf{e}([E : \boldsymbol{N}]) & := & \textbf{true} & \text{si } \mathbf{e}_c(E) = a \in \boldsymbol{N} \\
\mathbf{e}(\phi) & := & \textbf{false} & \text{dans les autres cas}
\end{array}
$$

Late LTS

$$\textsc{Input}\ \frac{\mathbf{e}_c(E) = a \in \boldsymbol{N}}{E(x).P \xrightarrow{a} (x)P} \qquad \textsc{Output}\ \frac{\mathbf{e}_c(E) = a \in \boldsymbol{N} \qquad \mathbf{e}_c(F) = M \in \boldsymbol{M}}{\overline{E}\langle F\rangle.P \xrightarrow{\overline{a}} \langle M\rangle P}$$

$$\textsc{Close-L}\ \frac{P \xrightarrow{a} F \qquad Q \xrightarrow{\overline{a}} C}{P \mid Q \xrightarrow{\tau} F \bullet C} \qquad \textsc{IfThen}\ \frac{P \xrightarrow{\mu} P'}{\phi P \xrightarrow{\mu} P'}\ \mathbf{e}(\phi) = \mathbf{true}$$

Late LTS

$$\textsc{Input}\ \frac{\mathbf{e}_c(E) = a \in \boldsymbol{N}}{E(x).P \xrightarrow{a} (x)P} \qquad \textsc{Output}\ \frac{\mathbf{e}_c(E) = a \in \boldsymbol{N} \qquad \mathbf{e}_c(F) = M \in \boldsymbol{M}}{\overline{E}\langle F\rangle.P \xrightarrow{\overline{a}} \langle M\rangle P}$$

$$\textsc{Close-l}\ \frac{P \xrightarrow{a} F \qquad Q \xrightarrow{\overline{a}} C}{P \mid Q \xrightarrow{\tau} F \bullet C} \qquad \textsc{IfThen}\ \frac{P \xrightarrow{\mu} P'}{\phi P \xrightarrow{\mu} P'}\ \mathbf{e}(\phi) = \mathbf{true}$$

$$\textsc{Res}\ \frac{P \xrightarrow{\mu} A}{(\nu z)\, P \xrightarrow{\mu} (\nu z)\, A}\ z \notin \mathrm{n}(\mu) \qquad \textsc{Par-l}\ \frac{P \xrightarrow{\mu} A}{P \mid Q \xrightarrow{\mu} A \mid Q}$$

+ Sum, Rep- et Alpha.

Bisimulations

Two processes are bisimilar if they can play the same transitions, i.e. they obey the game

$$P$$
$$\mathcal{R}$$
$$Q$$

Bisimulations

Two processes are bisimilar if they can play the same transitions, i.e. they obey the game

$$\begin{array}{ccc} P & \xrightarrow{\mu} & P' \\ \mathcal{R} & & \\ Q & & \end{array}$$

Bisimulations

Two processes are bisimilar if they can play the same transitions, i.e. they obey the game

$$\begin{array}{ccc} P & \xrightarrow{\mu} & P' \\ \mathcal{R} & & \\ Q & \overset{\mu}{\dashrightarrow} & Q' \end{array}$$

Bisimulations

Two processes are bisimilar if they can play the same transitions, i.e. they obey the game

$$\begin{array}{ccc} P & \xrightarrow{\mu} & P' \\ \mathcal{R} & & \mathcal{R} \\ Q & \overset{\mu}{\dashrightarrow} & Q' \end{array}$$

Bisimulations

Two processes are bisimilar if they can play the same transitions, i.e. they obey the game

$$\begin{array}{ccc} P & \xrightarrow{\mu} & P' \\ \mathcal{R} & & \mathcal{R} \\ Q & \overset{\mu}{\dashrightarrow} & Q' \end{array}$$

and

$$\begin{array}{ccc} Q & \xrightarrow{\mu} & Q' \\ \mathcal{R} & & \mathcal{R} \\ P & \overset{\mu}{\dashrightarrow} & P' \end{array}$$

Bisimulations in the spi calculus

- Bisimulations of pi calculus are too fine-grained.
- Indeed, if $P(c, M) := (\nu k)\, \overline{c}\langle \mathrm{Enc}^{\mathrm{s}}_k M \rangle . \mathbf{0}$ (where $k \notin \{c\} \cup \mathrm{n}(M)$)
  Requiring an exact match between actions makes distinguish $P(c, M)$ from $P(c, N)$.
  Whereas these two processes should be considered equivalent (since the encryption key is not disclosed to the environment).
- Environment-sensitive bisimulations : extend the notion of bisimulation with a data structure to encode environment knowledge.
- Framed bisimulation (Abadi, Gordon), alley bisimulation (Boreale et al.), hedged bisimulation (Borgström, Nestmann), ...

The attacker knowledge represented as hedges

- A hedge $h \in \boldsymbol{H}$ is a finite set of pairs of messages.
- Intuitively $(M, N) \in h$ means that M and N are indistinguishable.

Late hedged bisimulation

A symmetric hedged relation $\mathcal{R}$

Hedged relations

- A hedged relation $\mathcal{R}$ is a subset of $\boldsymbol{H} \times \boldsymbol{P} \times \boldsymbol{P}$ such that whenever $(h, P, Q) \in \mathcal{R}$, we have $\mathrm{fn}(P) \subseteq \mathrm{n}(\pi_1(h))$ and $\mathrm{fn}(Q) \subseteq \mathrm{n}(\pi_2(h))$.
- A hedged relation $\mathcal{R}$ is *symmetric* if whenever $(h, P, Q) \in \mathcal{R}$ we have $(h^{-1}, Q, P) \in \mathcal{R}$.

Late hedged bisimulation

A symmetric hedged relation $\mathcal{R}$ is a *(strong) late hedged bisimulation* if whenever $(h, P, Q) \in \mathcal{R}$, we have that

1. if $P \xrightarrow{\tau} P'$ then
   there exists Q' such that $Q \xrightarrow{\tau} Q'$ and $(h, P', Q') \in \mathcal{R}$

Late hedged bisimulation

A symmetric hedged relation $\mathcal{R}$ is a *(strong) late hedged bisimulation* if whenever $(h, P, Q) \in \mathcal{R}$, we have that

1. if $P \xrightarrow{\tau} P'$ then
   there exists Q' such that $Q \xrightarrow{\tau} Q'$ and $(h, P', Q') \in \mathcal{R}$
2. if $P \xrightarrow{a} (x)P'$ (with $x \notin \mathrm{n}(\pi_1(h))$)
   then

Late hedged bisimulation

A symmetric hedged relation $\mathcal{R}$ is a *(strong) late hedged bisimulation* if whenever $(h, P, Q) \in \mathcal{R}$, we have that

1. if $P \xrightarrow{\tau} P'$ then
   there exists Q' such that $Q \xrightarrow{\tau} Q'$ and $(h, P', Q') \in \mathcal{R}$
2. if $P \xrightarrow{a} (x)P'$ (with $x \notin \mathrm{n}(\pi_1(h))$)
   then
   there exist y and Q' such that $Q \xrightarrow{a} (y)Q'$
   (with $y \notin \mathrm{n}(\pi_2(h))$)

Late hedged bisimulation

A symmetric hedged relation $\mathcal{R}$ is a *(strong) late hedged bisimulation* if whenever $(h, P, Q) \in \mathcal{R}$, we have that

1. if $P \xrightarrow{\tau} P'$ then
   there exists Q' such that $Q \xrightarrow{\tau} Q'$ and $(h, P', Q') \in \mathcal{R}$
2. if $P \xrightarrow{a} (x)P'$ (with $x \notin \mathrm{n}(\pi_1(h))$)
   and $(a, b) \in h$ then
   there exist y and Q' such that $Q \xrightarrow{b} (y)Q'$
   (with $y \notin \mathrm{n}(\pi_2(h))$)

Late hedged bisimulation

A symmetric hedged relation $\mathcal{R}$ is a *(strong) late hedged bisimulation* if whenever $(h, P, Q) \in \mathcal{R}$, we have that

1. if $P \xrightarrow{\tau} P'$ then
   there exists Q' such that $Q \xrightarrow{\tau} Q'$ and $(h, P', Q') \in \mathcal{R}$
2. if $P \xrightarrow{a} (x)P'$ (with $x \notin \mathrm{n}(\pi_1(h))$)
   and $(a, b) \in h$ then
   there exist y and Q' such that $Q \xrightarrow{b} (y)Q'$
   (with $y \notin \mathrm{n}(\pi_2(h))$)
   and for all M
   we have $(h, P'\{^M/_x\}, Q'\{^M/_y\}) \in \mathcal{R}$.

The attacker knowledge represented as hedges

- A hedge $h \in \boldsymbol{H}$ is a finite set of pairs of messages.
- Intuitively $(M, N) \in h$ means that M and N are indistinguishable.
- The synthesis $\mathcal{S}(h)$ of a hedge h

$$\textsc{Syn-Inc}\ \frac{(M, N) \in h}{(M, N) \in \mathcal{S}(h)}$$

$$\textsc{Syn-Enc-s}\ \frac{(M_1, N_1) \in \mathcal{S}(h) \qquad (M_2, N_2) \in \mathcal{S}(h)}{(\mathrm{Enc}^{\mathrm{s}}_{M_2} M_1, \mathrm{Enc}^{\mathrm{s}}_{N_2} N_1) \in \mathcal{S}(h)}$$

$$\textsc{Syn-Pair}\ \frac{(M_1, N_1) \in \mathcal{S}(h) \qquad (M_2, N_2) \in \mathcal{S}(h)}{((M_1 \,.\, M_2), (N_1 \,.\, N_2)) \in \mathcal{S}(h)}$$

Late hedged bisimulation

A symmetric hedged relation $\mathcal{R}$ is a *(strong) late hedged bisimulation* if whenever $(h, P, Q) \in \mathcal{R}$, we have that

1. if $P \xrightarrow{\tau} P'$ then
   there exists Q' such that $Q \xrightarrow{\tau} Q'$ and $(h, P', Q') \in \mathcal{R}$
2. if $P \xrightarrow{a} (x)P'$ (with $x \notin \mathrm{n}(\pi_1(h))$)
   and $(a, b) \in h$ then
   there exist y and Q' such that $Q \xrightarrow{b} (y)Q'$
   (with $y \notin \mathrm{n}(\pi_2(h))$)
   and for all $(M, N) \in \mathcal{S}(h)$
   we have $(h, P'\{^M/_x\}, Q'\{^N/_y\}) \in \mathcal{R}$.

Late hedged bisimulation

A symmetric hedged relation $\mathcal{R}$ is a *(strong) late hedged bisimulation* if whenever $(h, P, Q) \in \mathcal{R}$, we have that

1. if $P \xrightarrow{\tau} P'$ then
   there exists Q' such that $Q \xrightarrow{\tau} Q'$ and $(h, P', Q') \in \mathcal{R}$
2. if $P \xrightarrow{a} (x)P'$ (with $x \notin \mathrm{n}(\pi_1(h))$)
   and $(a, b) \in h$ then
   there exist y and Q' such that $Q \xrightarrow{b} (y)Q'$
   (with $y \notin \mathrm{n}(\pi_2(h))$)
   and for all B and (M, N) such that $h \vdash_B (M, N)$
   we have $(h \cup B, P'\{^M/_x\}, Q'\{^N/_y\}) \in \mathcal{R}$.

Late hedged bisimulation

A symmetric hedged relation $\mathcal{R}$ is a *(strong) late hedged bisimulation* if whenever $(h, P, Q) \in \mathcal{R}$, we have that

1. if $P \xrightarrow{\tau} P'$ then
   there exists Q' such that $Q \xrightarrow{\tau} Q'$ and $(h, P', Q') \in \mathcal{R}$
2. if $P \xrightarrow{a} (x)P'$ (with $x \notin \mathrm{n}(\pi_1(h))$)
   and $(a, b) \in h$ then
   there exist y and Q' such that $Q \xrightarrow{b} (y)Q'$
   (with $y \notin \mathrm{n}(\pi_2(h))$)
   and for all B and (M, N) such that $h \vdash_B (M, N)$
   we have $(h \cup B, P'\{^M/_x\}, Q'\{^N/_y\}) \in \mathcal{R}$.
3. if $P \xrightarrow{\overline{a}} (\nu \tilde{c}) \langle M \rangle P'$ (with $\{\tilde{c}\} \cap \mathrm{n}(\pi_1(h)) = \emptyset$)
   and $(a, b) \in h$ then
   there exist $\left\{\tilde{d}\right\}$, Q' and N such that $Q \xrightarrow{\overline{b}} (\nu \tilde{d}) \langle N \rangle Q'$
   (with $\left\{\tilde{d}\right\} \cap \mathrm{n}(\pi_2(h)) = \emptyset$)

Late hedged bisimulation

A symmetric hedged relation $\mathcal{R}$ is a *(strong) late hedged bisimulation* if whenever $(h, P, Q) \in \mathcal{R}$, we have that

1. if $P \xrightarrow{\tau} P'$ then
   there exists Q' such that $Q \xrightarrow{\tau} Q'$ and $(h, P', Q') \in \mathcal{R}$
2. if $P \xrightarrow{a} (x)P'$ (with $x \notin \mathrm{n}(\pi_1(h))$)
   and $(a, b) \in h$ then
   there exist y and Q' such that $Q \xrightarrow{b} (y)Q'$
   (with $y \notin \mathrm{n}(\pi_2(h))$)
   and for all B and (M, N) such that $h \vdash_B (M, N)$
   we have $(h \cup B, P'\{^M/_x\}, Q'\{^N/_y\}) \in \mathcal{R}$.
3. if $P \xrightarrow{\overline{a}} (\nu \tilde{c}) \langle M \rangle P'$ (with $\{\tilde{c}\} \cap \mathrm{n}(\pi_1(h)) = \emptyset$)
   and $(a, b) \in h$ then
   there exist $\left\{\tilde{d}\right\}$, Q' and N such that $Q \xrightarrow{\overline{b}} (\nu \tilde{d}) \langle N \rangle Q'$
   (with $\left\{\tilde{d}\right\} \cap \mathrm{n}(\pi_2(h)) = \emptyset$)
   and $(h \cup \{(M, N)\}, P', Q') \in \mathcal{R}$.

Analysis of a hedge

- The analysis $\mathcal{A}(h)$ is the smallest hedge that is closed by $\mathrm{analz}(\cdot)$.

$$\textsc{Ana-Inc}\ \frac{(M,N) \in h}{(M,N) \in \mathrm{analz}(h)}$$

$$\textsc{Ana-Dec-s}\ \frac{(\mathrm{Enc}^{s}_{M_2} M_1, \mathrm{Enc}^{s}_{N_2} N_1) \in \mathrm{analz}(h) \qquad (M_2,N_2) \in \mathcal{S}(h)}{(M_1,N_1) \in \mathrm{analz}(h)}$$

$$\textsc{Ana-Fst}\ \frac{((M_1 \,.\, M_2),(N_1 \,.\, N_2)) \in \mathrm{analz}(h)}{(M_1,N_1) \in \mathrm{analz}(h)}$$

$$\textsc{Ana-Snd}\ \frac{((M_1 \,.\, M_2),(N_1 \,.\, N_2)) \in \mathrm{analz}(h)}{(M_2,N_2) \in \mathrm{analz}(h)}$$

Late hedged bisimulation

A symmetric hedged relation $\mathcal{R}$ is a *(strong) late hedged bisimulation* if whenever $(h, P, Q) \in \mathcal{R}$, we have that

1. if $P \xrightarrow{\tau} P'$ then
   there exists Q' such that $Q \xrightarrow{\tau} Q'$ and $(h, P', Q') \in \mathcal{R}$
2. if $P \xrightarrow{a} (x)P'$ (with $x \notin \mathrm{n}(\pi_1(h))$)
   and $(a, b) \in h$ then
   there exist y and Q' such that $Q \xrightarrow{b} (y)Q'$
   (with $y \notin \mathrm{n}(\pi_2(h))$)
   and for all B and (M, N) such that $h \vdash_B (M, N)$
   we have $(h \cup B, P'\{^M/_x\}, Q'\{^N/_y\}) \in \mathcal{R}$.
3. if $P \xrightarrow{\overline{a}} (\nu \tilde{c}) \langle M \rangle P'$ (with $\{\tilde{c}\} \cap \mathrm{n}(\pi_1(h)) = \emptyset$)
   and $(a, b) \in h$ then
   there exist $\left\{\tilde{d}\right\}$, Q' and N such that $Q \xrightarrow{\overline{b}} (\nu \tilde{d}) \langle N \rangle Q'$
   (with $\left\{\tilde{d}\right\} \cap \mathrm{n}(\pi_2(h)) = \emptyset$)
   and $(\mathcal{A}(h \cup \{(M, N)\}), P', Q') \in \mathcal{R}$.

Irreducibles

- $\mathcal{I}(h)$ is the smallest hedge such that $\mathcal{S}(\mathcal{I}(h)) = \mathcal{S}(\mathcal{A}(h))$.
- A hedge h is irreducible iff $\mathcal{I}(h) = h$

Late hedged bisimulation

A symmetric hedged relation $\mathcal{R}$ is a *(strong) late hedged bisimulation* if whenever $(h, P, Q) \in \mathcal{R}$, we have that

1. if $P \xrightarrow{\tau} P'$ then
   there exists Q' such that $Q \xrightarrow{\tau} Q'$ and $(h, P', Q') \in \mathcal{R}$
2. if $P \xrightarrow{a} (x)P'$ (with $x \notin \mathrm{n}(\pi_1(h))$)
   and $(a, b) \in h$ then
   there exist y and Q' such that $Q \xrightarrow{b} (y)Q'$
   (with $y \notin \mathrm{n}(\pi_2(h))$)
   and for all B and (M, N) such that $h \vdash_B (M, N)$
   we have $(h \cup B, P'\{^M/_x\}, Q'\{^N/_y\}) \in \mathcal{R}$.
3. if $P \xrightarrow{\overline{a}} (\nu \tilde{c}) \langle M \rangle P'$ (with $\{\tilde{c}\} \cap \mathrm{n}(\pi_1(h)) = \emptyset$)
   and $(a, b) \in h$ then
   there exist $\left\{\tilde{d}\right\}$, Q' and N such that $Q \xrightarrow{\overline{b}} (\nu \tilde{d}) \langle N \rangle Q'$
   (with $\left\{\tilde{d}\right\} \cap \mathrm{n}(\pi_2(h)) = \emptyset$)
   and $(\mathcal{I}(h \cup \{(M, N)\}), P', Q') \in \mathcal{R}$.

Irreducibles, consistency

- $\mathcal{I}(h)$ is the smallest hedge such that $\mathcal{S}(\mathcal{I}(h)) = \mathcal{S}(\mathcal{A}(h))$.
- A hedge h is irreducible iff $\mathcal{I}(h) = h$
- A hedge h is consistent iff :
  Whenever $(M, N) \in h$
  - $M \in \boldsymbol{N} \iff N \in \boldsymbol{N}$
  - whenever $(M', N') \in h : M = M' \iff N = N'$
  - $M \neq (M_1 \,.\, M_2)$ and $N \neq (N_1 \,.\, N_2)$
  - if $M = \mathrm{Enc}^{\mathrm{s}}_{M_2} M_1$ then $(M_2, N_2) \notin \mathcal{S}(h)$
  - if $N = \mathrm{Enc}^{\mathrm{s}}_{N_2} N_1$ then $(M_2, N_2) \notin \mathcal{S}(h)$
- A consistent hedge is irreducible.

Hedged relations

- A hedged relation $\mathcal{R}$ is a subset of $\boldsymbol{H} \times \boldsymbol{P} \times \boldsymbol{P}$ such that whenever $(h, P, Q) \in \mathcal{R}$, we have $\mathrm{fn}(P) \subseteq \mathrm{n}(\pi_1(h))$ and $\mathrm{fn}(Q) \subseteq \mathrm{n}(\pi_2(h))$.
- A hedged relation $\mathcal{R}$ is *symmetric* if whenever $(h, P, Q) \in \mathcal{R}$ we have $(h^{-1}, Q, P) \in \mathcal{R}$.
- A hedged relation $\mathcal{R}$ is *consistent* if whenever $(h, P, Q) \in \mathcal{R}$, we have that h is a consistent hedge.

Late hedged bisimulation

A symmetric consistent hedged relation $\mathcal{R}$ is a *(strong) late hedged bisimulation* if whenever $(h, P, Q) \in \mathcal{R}$, we have that

1. if $P \xrightarrow{\tau} P'$ then
   there exists Q' such that $Q \xrightarrow{\tau} Q'$ and $(h, P', Q') \in \mathcal{R}$
2. if $P \xrightarrow{a} (x)P'$ (with $x \notin \mathrm{n}(\pi_1(h))$)
   and $(a, b) \in h$ then
   there exist y and Q' such that $Q \xrightarrow{b} (y)Q'$
   (with $y \notin \mathrm{n}(\pi_2(h))$)
   and for all B and (M, N) such that $h \vdash_B (M, N)$
   we have $(h \cup B, P'\{^M/_x\}, Q'\{^N/_y\}) \in \mathcal{R}$.
3. if $P \xrightarrow{\overline{a}} (\nu \tilde{c}) \langle M \rangle P'$ (with $\{\tilde{c}\} \cap \mathrm{n}(\pi_1(h)) = \emptyset$)
   and $(a, b) \in h$ then
   there exist $\left\{\tilde{d}\right\}$, Q' and N such that $Q \xrightarrow{\overline{b}} (\nu \tilde{d}) \langle N \rangle Q'$
   (with $\left\{\tilde{d}\right\} \cap \mathrm{n}(\pi_2(h)) = \emptyset$)
   and $(\mathcal{I}(h \cup \{(M, N)\}), P', Q') \in \mathcal{R}$.

A word on open hedged bisimulation

- We have defined an open variant of late hedged bisimulation, following Sangiorgi's idea of open bisimulation in the pi calculus.
- The idea is to move the instantiation of input names before the transitions take place.
- For instance, the input clause roughly becomes

$$(\sigma, \rho) \rhd_B se \quad \begin{array}{cccc} P & P\sigma & \xrightarrow{a} & (x)P' \quad P' \\ | & & & | \\ Q & Q\rho & \xrightarrow{b} & (y)Q' \quad Q' \end{array}$$

A symbolic LTS

- *Idea :* record without checking the conditions needed to enable transitions.

$$\text{S-INPUT}\ \frac{}{E(x).P \xrightarrow[\{[E:\boldsymbol{N}]\}]{\mathbf{e}_a(E)} (x)P}$$

$$\text{S-OUTPUT}\ \frac{}{\overline{E}\langle F\rangle.P \xrightarrow[\{[E:\boldsymbol{N}],[F:\boldsymbol{M}]\}]{\overline{\mathbf{e}_a(E)}} \langle \mathbf{e}_a(F)\rangle P}$$

Abstract (or symbolic) evaluation of expressions :

$$
\begin{array}{rcll}
\mathbf{e}_{\mathrm{a}}(a) & := & a & \text{if } a \in \boldsymbol{N} \\
\mathbf{e}_{\mathrm{a}}(\mathrm{Enc}^{\mathrm{s}}_{F} E) & := & \mathrm{Enc}^{\mathrm{s}}_{\mathbf{e}_{\mathrm{a}}(F)} \mathbf{e}_{\mathrm{a}}(E) & \\
\mathbf{e}_{\mathrm{a}}((E \,.\, F)) & := & (\mathbf{e}_{\mathrm{a}}(E) \,.\, \mathbf{e}_{\mathrm{a}}(F)) & \\
\mathbf{e}_{\mathrm{a}}(\mathrm{Dec}^{\mathrm{s}}_{F} E) & := & E_1 & \text{if } \mathbf{e}_{\mathrm{a}}(E) = \mathrm{Enc}^{\mathrm{s}}_{E_2} E_1 \\
 & & \mathrm{Dec}^{\mathrm{s}}_{\mathbf{e}_{\mathrm{a}}(F)} \mathbf{e}_{\mathrm{a}}(E) & \text{otherwise} \\
\mathbf{e}_{\mathrm{a}}(\pi_1\,(E)) & := & E_1 & \text{if } \mathbf{e}_{\mathrm{a}}(E) = (E_1 \,.\, E_2) \\
 & & \pi_1\,(\mathbf{e}_{\mathrm{a}}(E)) & \text{otherwise} \\
\mathbf{e}_{\mathrm{a}}(\pi_2\,(E)) & := & E_2 & \text{if } \mathbf{e}_{\mathrm{a}}(E) = (E_1 \,.\, E_2) \\
 & & \pi_2\,(\mathbf{e}_{\mathrm{a}}(E)) & \text{otherwise}
\end{array}
$$

$$\text{S-INPUT}\ \frac{}{E(x).P \xrightarrow[\{[E:\boldsymbol{N}]\}]{\mathbf{e}_a(E)} (x)P}$$

$$\text{S-OUTPUT}\ \frac{}{\overline{E}\langle F\rangle.P \xrightarrow[\{[E:\boldsymbol{N}],[F:\boldsymbol{M}]\}]{\overline{\mathbf{e}_a(E)}} \langle \mathbf{e}_a(F)\rangle P}$$

$$\text{S-GUARD}\ \frac{P \xrightarrow[c]{\mu} A}{\phi P \xrightarrow[c\&\{\phi\}]{\mu} A}$$

$$\text{S-CLOSE-L}\ \frac{P \xrightarrow[c_1]{E} F \qquad Q \xrightarrow[c_2]{\overline{E'}} C}{P \mid Q \xrightarrow[\{[E=E']\}\&c_1\&c_2]{\tau} F \bullet C}$$

$$\text{S-INPUT}\ \frac{}{E(x).P \xmapsto[\{[E:\boldsymbol{N}]\}]{\mathbf{e}_a(E)} (x)P}$$

$$\text{S-OUTPUT}\ \frac{}{\overline{E}\langle F\rangle.P \xmapsto[\{[E:\boldsymbol{N}],[F:\boldsymbol{M}]\}]{\overline{\mathbf{e}_a(E)}} \langle \mathbf{e}_a(F)\rangle P}$$

$$\text{S-GUARD}\ \frac{P \overset{\mu}{\underset{c}{\longmapsto}} A}{\phi P \xmapsto[c\&\{\phi\}]{\mu} A}$$

$$\text{S-CLOSE-L}\ \frac{P \overset{E}{\underset{c_1}{\longmapsto}} F \qquad Q \overset{\overline{E'}}{\underset{c_2}{\longmapsto}} C}{P \mid Q \xmapsto[\{[E=E']\}\&c_1\&c_2]{\tau} F \bullet C}$$

$$\text{S-RES}\ \frac{P \overset{\mu}{\underset{c}{\longmapsto}} A}{(\nu z)\, P \xmapsto[\nu_+(z,c)]{\mu} (\nu z)\, A}\ z \notin \mathrm{n}(\mu)$$

Plan

Why formalize in Coq ?

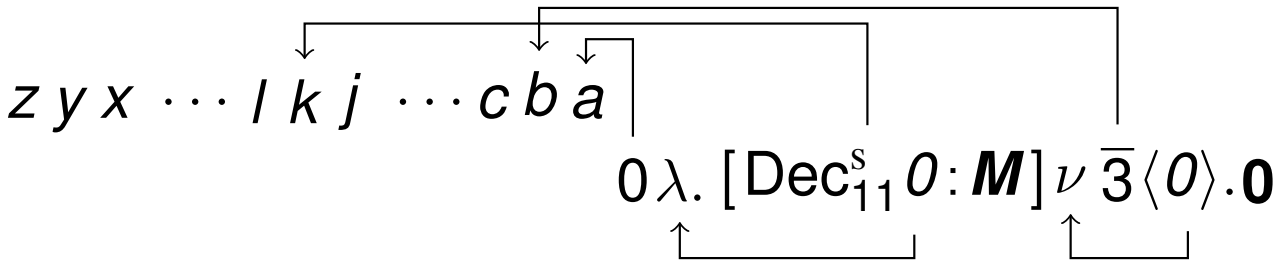
- *Dream :*
  extract a certified (correct by construction) bisimulation checker.
- Validate hand-written proofs.
- Provide an interactive framework to reason formally about cryptographic protocols within the spi calculus model.
- It's fun !

Representation of binders

- There exist several techniques to encode binders :
  - de Bruijn indices
  - locally nameless
  - higher-order abstract syntax
  - nominal
- We have chosen de Bruijn representation.

de Bruijn representation

Representation of $a(x).[\mathrm{Dec}_k^{\mathrm{S}} x : \boldsymbol{M}](\nu l)\,\overline{b}\langle l\rangle.\mathbf{0}$:

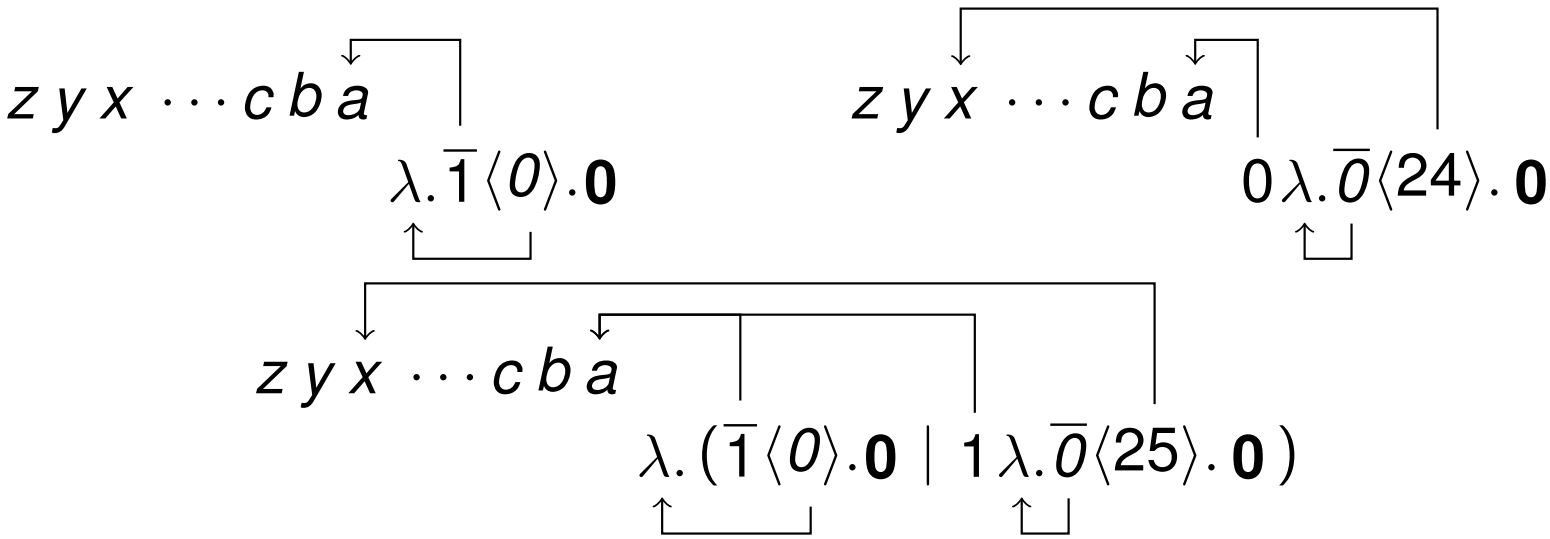


Operations on de Bruijn indices

- Parametrised by the binding depth d
- $\mathrm{mem}_d(i, t)$ returns **true** iff i is free in t
- $\mathrm{lift}_d(k, t)$ makes room for k new binders in t
  Used in parallel composition of an agent and a process :

$$\begin{aligned}(\lambda.P) \mid Q &:= \lambda.(P \mid \mathrm{lift}_0(1, Q)) \\ (\nu^k\langle F\rangle P) \mid Q &:= \nu^k\langle F\rangle(P \mid \mathrm{lift}_0(k, Q))\end{aligned}$$

For instance :



- $\mathrm{swap}_d(k, t)$ makes a circular permutation of the k first indices in t
- $\mathrm{low}_d(t)$ removes the first index
- Used in restriction of an agent :

$$\begin{array}{rcll} \boldsymbol{\nu}(\lambda.P) & := & \lambda.\nu\,\mathrm{swap}_0(1, P) & \\ \boldsymbol{\nu}(\nu^k\langle F\rangle P) & := & \nu^{k+1}\langle F\rangle P & \text{if } \mathrm{mem}_k(0, F) = \textbf{true} \\ & := & \nu^k\langle \mathrm{low}_k(F)\rangle\nu\,\mathrm{swap}_0(k, P) & \text{otherwise} \end{array}$$

- $\mathrm{lsubst}_d(k, \overline{E}, t)$ substitutes the $|\overline{E}|$ first indices with the corresponding expression of $\overline{E}$ in t. The k first indices are bound in $\overline{E}$.

$$(\lambda.P) \bullet (\nu^k\langle F\rangle Q) \quad := \quad \nu^k(\mathrm{lsubst}_0(k, F, P) \mid Q)$$

Concretely in Coq

- We have several types :
  names, messages, expressions, guards, processes, agents.
- These “de Bruijn” operations should be defined for each of these types, i.e. :

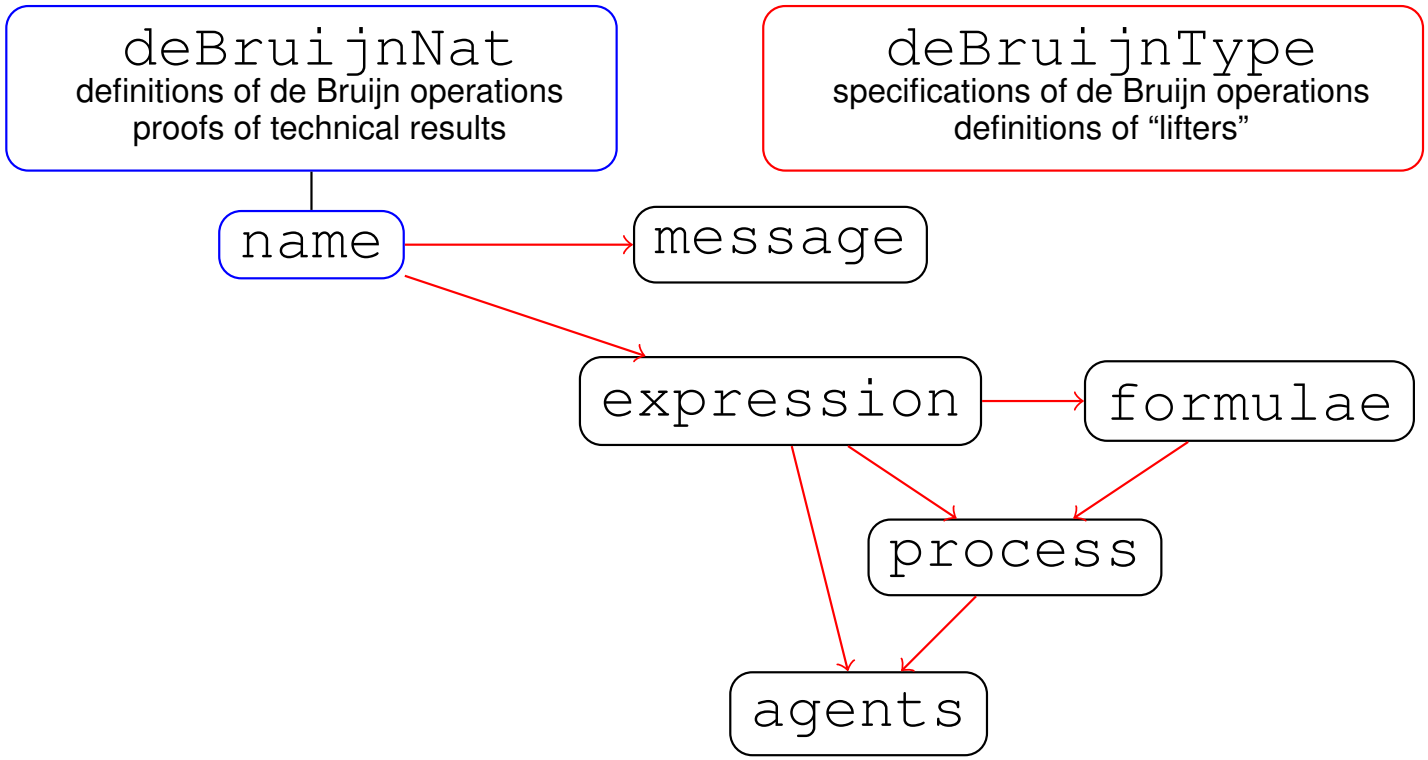
```
Definition name_lift
    (d:nat) (k:nat) (x:name) : name := ...
Definition message_lift ...
Definition expression_lift ...
Definition formula_lift ...
Definition process_lift ...
Definition agent_lift ...
```

  and similarly for the other operations.
- Moreover, several (about 60) facts hold for these operations.
  For instance, we have :

$$\forall d, d', k, k', x : \mathrm{swap}_{d+k+d'}(k', \mathrm{lift}_d(k, x)) = \mathrm{lift}_d(k, \mathrm{swap}_{d+d'}(k', x))$$

Abstracting de Bruijn indices

- Not scalable and tedious.
- Instead, define and specify all these operations on names
- and lift them to other types thanks to “good” iterators.
- In practice :



Asbtracting the LTS

- There are several LTS to define.
- Some properties are shared (for instance, structural congruence preserves the semantics)
- These LTS all follow the same pattern.
- Instead of defining each LTS separately, we make a functor and thus defer the definition of the semantics to the definitions of the semantics of actions.

Actions

We rely on a set of actions $\mathcal{A}$ and several functions to manipulate them :

- mkSil : $\mathcal{A}$ (silent)
- mkInp : $\boldsymbol{E} \to \mathcal{A} \cup \{\bot\}$ (input)
- mkOutp : $\boldsymbol{E} \times \boldsymbol{E} \to (\mathcal{A} \times \boldsymbol{E}) \cup \{\bot\}$ (output)
- mkRes : $\mathcal{A} \to \mathcal{A} \cup \{\bot\}$ (restriction)
- mkIf : $\boldsymbol{F} \times \mathcal{A} \to \mathcal{A} \cup \{\bot\}$ (guard)
- mkInt : $\mathcal{A} \times \mathcal{A} \to \mathcal{A} \cup \{\bot\}$ (interact)

The functor that defines an LTS

$$\textsc{Silent}\ \frac{}{\tau.P \xrightarrow{\text{mkSil}} P} \qquad \textsc{Input}\ \frac{\text{mkInp}(E) = \alpha \in \mathcal{A}}{E\lambda.P \xrightarrow{\alpha} \lambda.P}$$

$$\textsc{Output}\ \frac{\text{mkOutp}(E, F) = (\alpha, M) \in \mathcal{A} \times \boldsymbol{E}}{\overline{E}\langle F\rangle.P \xrightarrow{\alpha} \langle M\rangle P}$$

$$\textsc{Close-L}\ \frac{P \xrightarrow{\alpha} F \qquad Q \xrightarrow{\beta} C \qquad \text{mkInt}(\alpha, \beta) = \gamma \in \mathcal{A}}{P \mid Q \xrightarrow{\gamma} F \bullet C}$$

$$\textsc{Res}\ \frac{P \xrightarrow{\alpha} A \qquad \text{mkRes}(\alpha) = \beta \in \mathcal{A}}{\nu P \xrightarrow{\beta} \boldsymbol{\nu} A}$$

$$\textsc{IfThen}\ \frac{P \xrightarrow{\alpha} P' \qquad \mathsf{mkIf}(\phi, \alpha) = \beta \in \mathcal{A}}{\phi P \xrightarrow{\beta} P'}$$

$$\textsc{Par-L}\ \frac{P \xrightarrow{\alpha} A}{P \mid Q \xrightarrow{\alpha} A \mid Q}$$

$$\textsc{Sum-L}\ \frac{P \xrightarrow{\alpha} A}{P + Q \xrightarrow{\alpha} A}$$

$$\textsc{Rep-act}\ \frac{P \xrightarrow{\alpha} A}{!\,P \xrightarrow{\alpha} A \mid !\,P}$$

$$\textsc{Rep-close}\ \frac{P \xrightarrow{\alpha} F \qquad P \xrightarrow{\beta} C \qquad \mathsf{mkInt}(\alpha, \beta) = \gamma \in \mathcal{A}}{!\,P \xrightarrow{\gamma} (F \bullet C) \mid !\,P}$$

Plan

Structural congruence preserves the semantics

- $P \equiv Q$ iff P and Q represents intuitively the same process. For instance, $(\boldsymbol{P}, +, \mathbf{0})$ and $(\boldsymbol{P}, \mid, \mathbf{0})$ are monoids.
- We extend this definition to agents.
- A classical result is

Theorem

If $P \equiv Q$ and $P \xrightarrow{\mu} A$ then there exists B such that $A \equiv B$ and $Q \xrightarrow{\mu} B$.

- This theorem constitutes a good crash test for our formalization.
- With our definitions, we show

Theorem

If the set of actions $\mathcal{A}$ and the functions mkSil, mkInp, . . . *satisfy some conditions, then if $P \equiv Q$ and $P \xrightarrow{\alpha} A$ then there exist β and B such that $A \equiv B$, $Q \xrightarrow{\beta} B$ and $\alpha = \beta$.*

- This theorem constitutes a good crash test for our formalization.
- With our definitions, we show

Theorem

If the set of actions $\mathcal{A}$ and the functions mkSil, mkInp, ... *satisfy some conditions, then if $P \equiv Q$ and $P \xrightarrow{\alpha} A$ then there exist β and B such that $A \equiv B$, $Q \xrightarrow{\beta} B$ and $\alpha = \beta$.*

- Thanks to our formalization, we have noticed that this result does not hold for the symbolic LTS !
  Indeed let $P := (\nu x)\, x(z).\mathbf{0}$ and $Q := \overline{y}\langle k\rangle.\mathbf{0}$.
  We have $P \mid Q \equiv (\nu x)\,(x(z).\mathbf{0} \mid \overline{y}\langle k\rangle.\mathbf{0})$.
  $P \mid Q$ cannot perform any internal transition whereas
  $(\nu x)\,(x(z).\mathbf{0} \mid \overline{y}\langle k\rangle.\mathbf{0}) \xrightarrow[(\nu x)\,\{[x:\boldsymbol{N}],[y:\boldsymbol{N}],[x=y],[k:\boldsymbol{M}]\}]{\tau} (\nu x)\,(\mathbf{0} \mid \mathbf{0})$

Back to the analysis

- We defined the analysis $\mathcal{A}(h)$ as being the smallest hedge that is closed by $\mathrm{analz}(\cdot)$.
- In Coq, this requires some work to show that this definition makes sense.
- To show the existence, we exhibit a multiset that stricly decreases.
- Fortunately, the CoLoR library shows that multiset ordering is well-founded.

A small example of bisimulation

- Define $P(c, M) := (\nu k)\,\overline{c}\langle \mathrm{Enc}^{\mathrm{s}}_k M\rangle.\,\mathbf{0}$ where $k \notin \{c\} \cup \mathrm{n}(M)$.
- We show that for any c, M and N we have $P(c, M) \dot{\sim}^{h}_{\mathrm{LH}} P(c, N)$ where $h = \mathcal{I}(\{(c, c), (M, M), (N, N)\})$.
- In Coq : $P(c, M) := \nu\overline{\mathrm{lift}_0(1, c)}\langle \mathrm{Enc}^{\mathrm{s}}_0 \mathrm{lift}_0(1, M)\rangle.\,\mathbf{0}$.
- We exhibit a late hedged bisimulation.

$$
\begin{aligned}
\mathcal{R} := \quad & \{(h_0(c, M, N), P(c, M), P(c, N))\} \\
\cup \quad & \{(h_1(c, M, N, k, k), \mathbf{0}, \mathbf{0}) \mid k \notin \mathrm{n}(h_0(c, M, N))\} \\
\cup \quad & \{(h_0(c, N, M), P(c, N), P(c, M))\} \\
\cup \quad & \{(h_1(c, N, M, k, k), \mathbf{0}, \mathbf{0}) \mid k \notin \mathrm{n}(h_0(c, N, M))\}
\end{aligned}
$$

where

$$
\begin{aligned}
h_0(c, M, N) &:= \mathcal{I}(\{(c, c), (M, M), (N, N)\}) \\
h_1(c, M, N, k, l) &:= h_0 \cup \{(\mathrm{Enc}^{\mathrm{s}}_k M, \mathrm{Enc}^{\mathrm{s}}_k N)\}
\end{aligned}
$$

A variant

- Define $Q(c, M, M') = (\nu k)\, \overline{c}\langle \mathrm{Enc}^{\mathrm{s}}_k M \rangle.\overline{c}\langle \mathrm{Enc}^{\mathrm{s}}_k M' \rangle.\, \mathbf{0}$ where $k \notin \{c\} \cup \mathrm{n}(M, M')$.
- We show that for any c, M, N and N', if $N \neq N'$ then there is no hedge h such that $(c, c) \in h$ (i.e. the channel c is known by the attacker) and $Q(c, M, M) \dot{\sim}^{h}_{\mathrm{LH}} Q(c, N, N')$.
- The proof proceeds by contradiction.

$$h' := \mathcal{I}(h \cup \{\mathrm{Enc}^{\mathrm{s}}_k M, \mathrm{Enc}^{\mathrm{s}}_l N\})$$
$$h'' := \mathcal{I}(h' \cup \{\mathrm{Enc}^{\mathrm{s}}_k M, \mathrm{Enc}^{\mathrm{s}}_l N'\})$$

At some point, we have that h'' is consistent. This implies necessarily that $N = N'$.

Conclusion

- We have formalized an “abstract” pi calculus

$$\begin{array}{rcl} P, Q & ::= & \mathbf{0} \mid P + Q \\ & \mid & P \mid Q \mid !\,P \\ & \mid & (\nu x)\,P \mid \tau.P \\ & \mid & E(x).P \mid \overline{E}\langle F \rangle.P \\ & \mid & \phi P \end{array}$$

- and have made two instantiations of this calculus : monadic pi calculus and spi calculus.
- We have shown a general theorem about structural congruence.
- We have defined 3 LTS for the spi calculus and studied their properties.
  We have thus fixed several errors in the handwritten proofs.
- We have formalized the hedges and then defined late hedged bisimulation in Coq.

Future work

- Continue the formalization until realizing our dream of having a correct-by-construction bisimulation checker.
- Test our de Bruijn “library” on other formalisms (POPLMark ?)
- Develop tactics to ease reasoning in Coq with our framework.

Thanks !
Questions ?

Bibliography


D. Sangiorgi
A Theory of Bisimulation for the π-calculus.


M. Abadi and A. Gordon
A Calculus for Cryptographic Protocols : The Spi Calculus


J. Borgström, S. Briais and U. Nestmann
Symbolic Bisimulations in the Spi Calculus


S. Briais and U. Nestmann
Open Bisimulation, Revisited

Operations on agents

- Pseudo-application :
  If $F = (x)P$ is an abstraction and
  $C = (\nu\tilde{z})\,\langle M\rangle Q$ is a concretion (with $\{\tilde{z}\} \cap \mathrm{fn}(P) = \emptyset$),
  then $F \bullet C := (\nu\tilde{z})\,(P\{^{M}/_{x}\} \mid Q)$
- Restriction :

$$
\begin{array}{rcll}
(\boldsymbol{\nu} x)\,P & := & (\nu x)\,P & \\
(\boldsymbol{\nu} x)\,((y)P) & := & (y)(\nu x)\,P & \text{if } y \neq x \\
(\boldsymbol{\nu} x)\,((\nu\tilde{z})\,\langle M\rangle P) & := & (\nu\tilde{z})\,\langle M\rangle(\nu x)\,P & \text{if } y \notin \{\tilde{z}\} \text{ and } x \notin \mathrm{n}(M) \\
(\boldsymbol{\nu} x)\,((\nu\tilde{z})\,\langle M\rangle P) & := & (\nu x\tilde{z})\,\langle M\rangle P & \text{if } y \notin \{\tilde{z}\} \text{ and } x \in \mathrm{n}(M)
\end{array}
$$

- Parallel composition :

$$
\begin{array}{rcll}
((x)P) \mid Q & := & (x)(P \mid Q) & \text{if } x \notin \mathrm{fn}(Q) \\
((\nu\tilde{z})\,\langle M\rangle P) \mid Q & := & (\nu\tilde{z})\,\langle M\rangle(P \mid Q) & \text{if } \{\tilde{z}\} \cap \mathrm{fn}(Q) = \emptyset
\end{array}
$$

Possible inputs

Let $h \in \boldsymbol{H}$, $(M, N) \in \boldsymbol{M} \times \boldsymbol{M}$
Let $B \subseteq \boldsymbol{N} \times \boldsymbol{N}$ a consistent hedge such that

- $\pi_1(B) \cap \mathrm{n}(\pi_1(h)) = \emptyset$
- $\pi_2(B) \cap \mathrm{n}(\pi_2(h)) = \emptyset$

i.e. the names of B are fresh component-wise w.r.t. those of h.
We write $h \vdash_B (M, N)$ if

- $\forall (b_1, b_2) \in B : b_1 \in \mathrm{n}(M) \vee b_2 \in \mathrm{n}(N)$
- $(M, N) \in \mathcal{S}(h \cup B)$

A LTS that collects type constraints

$$\text{NC-SILENT}\ \frac{}{\tau.P \xrightarrow[\emptyset]{\tau} P} \qquad \text{NC-INPUT}\ \frac{\mathbf{e}_c(E) = a \in \boldsymbol{N}}{E(x).P \xrightarrow[\{a\}]{a} (x)P}$$

$$\text{NC-OUTPUT}\ \frac{\mathbf{e}_c(E) = a \in \boldsymbol{N} \qquad \mathbf{e}_c(F) = M \in \boldsymbol{M}}{\overline{E}\langle F\rangle.P \xrightarrow[\{a\}]{\overline{a}} \langle M\rangle P}$$

$$\text{NC-IFTHEN}\ \frac{P \xrightarrow[S]{\mu} A}{\phi P \xrightarrow[S \cup \mathbf{nc}(\phi)]{\mu} A}\ \mathbf{e}(\phi) = \mathbf{true}$$

where $\mathbf{nc}([E : \boldsymbol{N}]) := \{\mathbf{e}_c(E)\}$ and $\mathbf{nc}([E = F]) := \emptyset$.

Theorem

The two semantics are equivalent :

1. *If* $P \xrightarrow{\mu} A$ *there exists* $S \subseteq \boldsymbol{N}$ *such that* $P \xrightarrow[S]{\mu} A$.
2. *If* $P \xrightarrow[S]{\mu} A$ *then* $P \xrightarrow{\mu} A$.

Lemma

If $P \xrightarrow[S]{\mu} A$ *and* $\sigma : \boldsymbol{N} \to \boldsymbol{M}$ *is a substitution such that* $S\sigma \subseteq \boldsymbol{N}$ *then* $P\sigma \xrightarrow[S\sigma]{\mu\sigma} A\sigma$.

A symbolic LTS

- *Idea :* record without checking the conditions needed to enable transitions.
- A transition constraint has the form $(\nu\tilde{z})\,\Phi$ where Φ is a finite set of guards and $\tilde{z}$ is a finite set of names that occur in Φ, i.e. $\{\tilde{z}\} \subseteq \mathrm{n}(\Phi)$
- Composition of constraints :
  - Conjunction of $c_1 = (\nu\tilde{z_1})\,\Phi_1$ and $c_2 = (\nu\tilde{z_2})\,\Phi_2$

$$c_1 \;\&\; c_2 := (\Phi_1 \cup \Phi_2)$$

  - Restriction of name x.
    If $c = (\nu\tilde{z})\,\Phi$ and $x \notin \{\tilde{z}\}$:

$$\begin{array}{rcll} \nu_+(x, c) & := & (\nu x\tilde{z})\,\Phi & \text{if } x \in \mathrm{fn}(c) \\ & := & c & \text{otherwise} \end{array}$$

Define $>_o$ as being the smallest precongruence on expressions that satisfies :

- $\pi_1((E_1 . E_2)) >_o E_1$ if $\mathbf{e}_c(E_2) \neq \perp$
- $\pi_2((E_1 . E_2)) >_o E_2$ if $\mathbf{e}_c(E_1) \neq \perp$
- $\mathrm{Dec}^s_{E_2}\mathrm{Enc}^s_{E_2} E_1 >_o E_1$ if $\mathbf{e}_c(E_2) \neq \perp$

Extend this relation to agents in :

- $A >_o^= B$ (A, B are concrete agents)
- $A >_o^{\mathbf{e}} B$ (A is symbolic, B is concrete)

(two ways to handle concretions)

Theorem

Let $P, Q \in \mathbf{P}$ and assume that $P >_o Q$.

1. *If $P \xrightarrow[s]{\mu} A$ then $Q \xrightarrow[s]{\mu} B$ and $A >_o^= B$*
2. *If $Q \xrightarrow[s]{\mu} B$ then $P \xrightarrow[s]{\mu} A$ and $A >_o^= B$*

Theorem

Let $P, Q \in \mathbf{P}$ and $\sigma : \mathbf{N} \to \mathbf{M}$ a substitution.

1. *If $P \xmapsto[c]{\mu_s} A$ and $\mathbf{e}(c\sigma) = \mathbf{true}$ then $P\sigma \xrightarrow[\mathbf{nc}(c\sigma)]{\mathbf{e}_c(\mu_s\sigma)} B$ with $A\sigma >_o^{\mathbf{e}} B$*
2. *If $P\sigma \xrightarrow[s]{\mu} B$ then $P \xmapsto[c]{\mu_s} A$ with $\mathbf{e}(c\sigma) = \mathbf{true}$, $\mathbf{nc}(c\sigma) = S$, $\mathbf{e}_c(\mu_s\sigma) = \mu$ and $A\sigma >_o^{\mathbf{e}} B$*