Contracts for Mobile Processes

Giuseppe Castagna   Luca Padovani

Laboratoire PPS, CNRS, Université Paris Diderot

Istituto di Scienze e Tecnologie dell'Informazione, Università di Urbino

CONCUR 2009

Outline

1. Motivation
   - Protocols and processes
   - Contracts and mobile systems
2. Contracts
   - Syntax
   - Semantics
3. Results
4. Concluding remarks

Protocols and processes

Session types

- prescriptions on the use of channels

$$u : \sigma, v : \tau, \cdots \vdash P$$

Contracts

- overall process behavior

$$u : \mathtt{Ch}, v : \mathtt{Ch}, \cdots \vdash P : T$$

Summary

- both are behavioral types
- σ = projection of T on u

What session types and contracts are for

Characterizing well-formed systems

- the system eventually terminates
- the system never deadlocks

Characterizing well-typed processes

- sent messages have the correct/expected type
- messages sent/delivered in the right order

Reasoning about processes by means of their type

- refactoring processes
- searching for services

A problem of abstraction

Session types	Contracts
$\texttt{?Int.?Int.}(\texttt{!Real} \oplus \texttt{!Error})$	$a.a.(\overline{b} \oplus \overline{c})$
$\texttt{?}(\texttt{!Bool.!Bool})$	a ?

A natural candidate

Contracts without channel passing $\Rightarrow$ CCS

Contracts with channel passing $\Rightarrow$ π-calculus

A problem of abstraction

Session types	Contracts
$\texttt{?Int.?Int.}(\texttt{!Real} \oplus \texttt{!Error})$	$a.a.(\overline{b} \oplus \overline{c})$
$\texttt{?}(\texttt{!Bool.!Bool})$	a ?

A natural candidate

Contracts without channel passing $\Rightarrow$ **CCS**

Contracts with channel passing $\Rightarrow$ π-calculus

An example

process $\texttt{store}?(x).x?(y : \texttt{Item}).$
$\quad\texttt{if}\ y\ \text{is in stock}$
$\quad\quad\texttt{then}\ \texttt{bank}!(\!|x|\!)$
$\quad\quad\texttt{else}\ x!\langle \text{available}(y)\rangle$

$$\Downarrow$$

contract $\texttt{store}?(x).x?\texttt{Item}.(\texttt{bank}!x.\texttt{1} \oplus x!\texttt{Date}.\texttt{1})$

An example

process $\boxed{\texttt{store}?(x)}.x?(y:\texttt{Item}).$
$\texttt{if}\ y$ is in stock
$\texttt{then}\ \boxed{\texttt{bank}!(\!|x|\!)}$
$\texttt{else}\ x!\langle\text{available}(y)\rangle$

$$\Downarrow$$

contract $\boxed{\texttt{store}?(x)}.x?\texttt{Item}.(\boxed{\texttt{bank}!x}.1 \oplus x!\texttt{Date}.1)$

An example

process $\texttt{store?}(x).\boxed{x\texttt{?}(y : \texttt{Item})}.$

$\quad\quad \texttt{if}\ y$ is in stock

$\quad\quad\quad \texttt{then bank!}\llparenthesis x \rrparenthesis$

$\quad\quad\quad \texttt{else}\ \boxed{x!\langle \text{available}(y) \rangle}$

$$\Downarrow$$

contract $\texttt{store?}(x).\boxed{x\texttt{?Item}}.(\texttt{bank!}x.\mathbf{1} \oplus \boxed{x!\texttt{Date}}.\mathbf{1})$

Some typing rules

v-send

$$\frac{\Gamma \vdash e : t \qquad \Gamma \vdash P : T}{\Gamma \vdash \alpha!e.P : \alpha!t.T}$$

v-recv

$$\frac{\Gamma, x : t \vdash P : T}{\Gamma \vdash \alpha?(x : t).P : \alpha?t.T}$$

c-send

$$\frac{\Gamma \vdash P : T}{\Gamma \vdash \alpha!(\!|\beta|\!).P : \alpha!\beta.T}$$

c-recv

$$\frac{\Gamma, x : \mathtt{Ch} \vdash P : T}{\Gamma \vdash \alpha?(\!|x|\!).P : \alpha?(x).T}$$

Some typing rules

v-send

$$\frac{\Gamma \vdash e : t \qquad \Gamma \vdash P : T}{\Gamma \vdash \alpha!e.P : \alpha!t.T}$$

v-recv

$$\frac{\Gamma, x : t \vdash P : T}{\Gamma \vdash \alpha?(x : t).P : \alpha?t.T}$$

c-send

$$\frac{\Gamma \vdash P : T}{\Gamma \vdash \alpha!(\!|\beta|\!).P : \alpha!\beta.T}$$

c-recv

$$\frac{\Gamma, x : \mathtt{Ch} \vdash P : T}{\Gamma \vdash \alpha?(\!|x|\!).P : \alpha?(x).T}$$

undecidable $\longrightarrow$ decidable

Outline

Syntax

failure, success

$$T ::= 0 \mid 1 \qquad \pi.T \mid T + T \mid T \oplus T \qquad T|T \mid (\nu a)T$$

$$\pi ::= \alpha?f \mid \alpha!f \mid \alpha!(a)$$

$$f ::= x \mid (x) \mid a \mid \texttt{Int} \mid \texttt{Bool} \mid \cdots$$

Infinite behaviors = infinite terms

- regularity $\qquad X = c?\texttt{Int}.X$
- boundedness $\qquad X = a?(x).(c!x.1 \mid X)$

Syntax

dynamic operators

$$T \;::=\; 0 \mid 1 \qquad \pi.T \mid T + T \mid T \oplus T \qquad T|T \;\mid\; (\boldsymbol{\nu} a)T$$

$$\pi \;::=\; \alpha?f \mid \alpha!f \mid \alpha!(a)$$

$$f \;::=\; x \mid (x) \mid a \mid \texttt{Int} \mid \texttt{Bool} \mid \cdots$$

Infinite behaviors = infinite terms

- regularity $X = c?\texttt{Int}.X$
- boundedness $X = a?(x).(c!x.1 \mid X)$

Syntax

systems

$$T ::= 0 \mid 1 \qquad \pi.T \mid T + T \mid T \oplus T \qquad T|T \mid (\nu a)T$$

$$\pi ::= \alpha?f \mid \alpha!f \mid \alpha!(a)$$

$$f ::= x \mid (x) \mid a \mid \mathtt{Int} \mid \mathtt{Bool} \mid \cdots$$

Infinite behaviors = infinite terms

- regularity $\qquad X = c?\mathtt{Int}.X$
- boundedness $\qquad X = a?(x).(c!x.1 \mid X)$

Syntax

$$T \quad ::= \quad 0 \mid 1 \qquad \pi.T \mid T + T \mid T \oplus T \qquad T|T \;\mid\; (\boldsymbol{\nu} a)T$$

$$\pi \quad ::= \quad \alpha?f \mid \alpha!f \mid \alpha!(a)$$

prefixes

$$f \quad ::= \quad x \mid (x) \mid a \mid \mathtt{Int} \mid \mathtt{Bool} \mid \cdots$$

Infinite behaviors = infinite terms

- regularity $\qquad X = c?\mathtt{Int}.X$
- boundedness $\qquad X = a?(x).(c!x.1 \mid X)$

Syntax

$$T ::= 0 \mid 1 \qquad \pi.T \mid T + T \mid T \oplus T \qquad T|T \mid (\boldsymbol{\nu} a)T$$

$$\pi ::= \alpha?f \mid \alpha!f \mid \alpha!(a)$$

$$f ::= x \mid (x) \mid a \mid \mathtt{Int} \mid \mathtt{Bool} \mid \cdots$$

patterns = sets of values and names + binders

Infir

- regularity $X = c?\mathtt{Int}.X$
- boundedness $X = a?(x).(c!x.1 \mid X)$

Syntax

$$T ::= 0 \mid 1 \qquad \pi.T \mid T + T \mid T \oplus T \qquad T|T \mid (\nu a)T$$

$$\pi ::= \alpha?f \mid \alpha!f \mid \alpha!(a)$$

$$f ::= x \mid (x) \mid a \mid \mathtt{Int} \mid \mathtt{Bool} \mid \cdots$$

Infinite behaviors = infinite terms

- regularity $X = c?\mathtt{Int}.X$
- boundedness $X = a?(x).(c!x.1 \mid X)$

Labeled operational semantics

$$1 \xrightarrow{\checkmark} 1$$

$$\frac{m \in f \rightsquigarrow \sigma}{c?f.T \xrightarrow{c?m} T\sigma} \qquad \frac{m \in f}{c!f.T \longrightarrow c!m.T} \qquad c!m.T \xrightarrow{c!m} T$$

Example

$$c!\texttt{Int}.1 \mid c?\texttt{Real}.1 \xrightarrow{20 \in \texttt{Int}} c!20.1 \mid c?\texttt{Real}.1 \xrightarrow{20 \in \texttt{Real} \rightsquigarrow \emptyset} 1 \mid 1 \xrightarrow{\checkmark}$$

Contracts as behavioral types

Systems

$$S \stackrel{\text{def}}{=} T_1 \mid T_2 \mid \cdots \mid T_n$$

1. when is a system well-formed?
2. when is a process well-typed?
3. when are two types equal?

Participant satisfaction

Definition

$T \triangleleft S$ if $T \mid S \Longrightarrow T' \mid S'$ and $T' \not\longrightarrow$ implies

- $T' \xrightarrow{\mu_1}$ and $S' \xRightarrow{\mu_2}$
- $\mu_1 \# \mu_2$ $\qquad (c!m \# c?m,\ \checkmark \# \checkmark)$

for some μ_1 and μ_2

Examples

- $c!\mathtt{Int}.1 \triangleleft c?\mathtt{Real}.1$
- $c!\mathtt{Real}.1 \not\triangleleft c?\mathtt{Int}.1$
  $c!\mathtt{Real}.1 \mid c?\mathtt{Int}.1 \longrightarrow c!\sqrt{2}.1 \mid c?\mathtt{Int}.1$ stuck

Well-formed systems

$$S \stackrel{\text{def}}{=} T_1 \mid T_2 \mid \cdots \mid T_n$$

Definition

S is *well formed* if $\quad T_k \triangleleft \prod_{i \in \{1,\dots,n\} \setminus \{k\}} T_i \quad$ for every $1 \le k \le n$

Examples

- $c!\mathtt{Int}.\mathtt{1} \mid c?\mathtt{Real}.\mathtt{1}$ is well formed
- $c!\mathtt{Real}.\mathtt{1} \mid c?\mathtt{Int}.\mathtt{1}$ is ill formed

Well-typed participant

Definition

T is *viable* if $T \mid S$ is well formed for some S

Example

$$
\begin{aligned}
T &\stackrel{\text{def}}{=} c?\mathtt{Int}.1 + c?\mathtt{Bool}.0 \\
S &\stackrel{\text{def}}{=} c?\mathtt{Int}.0 + c?\mathtt{Bool}.1
\end{aligned}
$$

- T is viable
- S is viable
- $T \oplus S$ is not viable

Example: global order on channels

$$P \stackrel{\text{def}}{=} a?(\!|x|\!).b?(\!|y|\!).x!3.x?(z : \texttt{Int}).y!\texttt{true}.0$$

$$P' \stackrel{\text{def}}{=} a?(\!|x|\!).b?(\!|y|\!).x!3.y!\texttt{true}.x?(z : \texttt{Int}).0$$

$$Q \stackrel{\text{def}}{=} a!(c).b!(d).c?(z : \texttt{Int}).d?(z : \texttt{Bool}).c!5.0$$

$$Q' \stackrel{\text{def}}{=} a!(c).b!(d).c?(z : \texttt{Int}).c!5.d?(z' : \texttt{Bool}).0$$

- deadlock because of cyclic dependency
- $T_P \mid T_Q$ ill-formed (**not viable!**)

Example: global order on channels

$$P \stackrel{\text{def}}{=} a?\llparenthesis x \rrparenthesis.b?\llparenthesis y \rrparenthesis.x!3.x?(z : \mathtt{Int}).y!\mathtt{true}.0$$

$$P' \stackrel{\text{def}}{=} a?\llparenthesis x \rrparenthesis.b?\llparenthesis y \rrparenthesis.x!3.y!\mathtt{true}.x?(z : \mathtt{Int}).0$$

$$Q \stackrel{\text{def}}{=} a!(c).b!(d).c?(z : \mathtt{Int}).d?(z : \mathtt{Bool}).c!5.0$$

$$Q' \stackrel{\text{def}}{=} a!(c).b!(d).c?(z : \mathtt{Int}).c!5.d?(z' : \mathtt{Bool}).0$$

- imposing global order
- $T_P \mid T_{Q'}$ well-formed

Example: global order on channels

$$P \stackrel{\text{def}}{=} a?(\!|x|\!).b?(\!|y|\!).x!3.x?(z : \texttt{Int}).y!\texttt{true}.0$$

$$P' \stackrel{\text{def}}{=} a?(\!|x|\!).b?(\!|y|\!).x!3.y!\texttt{true}.x?(z : \texttt{Int}).0$$

$$Q \stackrel{\text{def}}{=} a!(c).b!(d).c?(z : \texttt{Int}).d?(z : \texttt{Bool}).c!5.0$$

$$Q' \stackrel{\text{def}}{=} a!(c).b!(d).c?(z : \texttt{Int}).c!5.d?(z' : \texttt{Bool}).0$$

- global order is not necessary
- $T_{P'} \mid T_Q$ well-formed

Example: linearity

$$a?(\!|x|\!).b?(\!|y|\!).x!(\!|y|\!).x?(z : \mathtt{Int}).y!\mathtt{true}.0$$

$$a!(c).b!(d).c?(\!|z|\!).c!5.z?(z' : \mathtt{Bool}).0$$

Subcontract

Definition

$T \preceq S$ if $T \mid R$ well formed implies $S \mid R$ well formed for every R

Examples

- $T \oplus S \preceq T$
- $\pi.T + \pi.S \approx \pi.(T \oplus S)$

  ... very much like the *must* preorder ...
- $0 \preceq T$

$\preceq$ is not a precongruence

$$0 \preceq T$$

Definition (strong subcontract)

Let $\sqsubseteq$ be the largest precongruence included in $\preceq$

Theorem

If T is viable, then $T \preceq S$ iff $T \sqsubseteq S$

- $T \sqsubseteq 0$ iff T is not viable
- if $1 + T \sqsubseteq T$, then T is well formed
- $\pi.0 \sqsubseteq \pi.T$

Outline

On progress

Theorem

If $\vdash P : T$ *and* T *w.f. and* $P \stackrel{\tau}{\Longrightarrow} Q \stackrel{\tau}{\not\longrightarrow}$, *then* Q *has succeeded*

- success = "no pending actions"

On decidability

Proposition

- *well-formedness*
- *viability*
- *subcontract*

are decidable provided that $c!f$ *matches finitely many names*

If a name is sent:

- either it is fresh $c!(a)$
- or it is a public name $c!a$
- or it was received earlier $c?(\mathsf{x}) \cdots d!\mathsf{x}$

Outline

Session types and contracts: a comparison

- optimistic *vs* conservative
- global *vs* compositional

	Session types	Contracts
structuring	++	−−
analysis	−−	++

Concluding remarks

Contributions

1. contracts for processes with channel mobility
2. straightforward solution to global progress
   (of bounded systems)

Our wish list

- algorithms (almost done)
- choreographic specifications
- expressiveness

Concluding remarks

Contributions

1. contracts for processes with channel mobility
2. straightforward solution to global progress (of bounded systems)

Our wish list

- algorithms (almost done)
- choreographic specifications
- expressiveness

Thank you.

Regular does not mean finite-state

- unbounded participants
- unbounded buffers
- state encoded within processes

$$\begin{array}{rcl} P(x : \mathtt{Int}) & = & \mathtt{deposit?}(y : \mathtt{Int}).P(x+y) \\ & & + \, \mathtt{withdraw?}(y : \mathtt{Int}).P(\max\{0, x-y\}) \end{array}$$

$$P(0)$$

$$\begin{array}{rcl} P & = & c?(x : \mathtt{Int}).(\mathtt{deposit?}(y : \mathtt{Int}).c!\langle x+y \rangle.P \\ & & + \, \mathtt{withdraw?}(y : \mathtt{Int}).c!\langle \max\{0, x-y\} \rangle.P) \\ Q & = & c?(x : \mathtt{Int}).c!\langle x \rangle.Q \end{array}$$

$$(\boldsymbol{\nu} c)(P \mid c!\langle 0 \rangle.Q)$$

Simulating asynchrony

input

$$\frac{\Gamma \vdash \alpha : \mathtt{Ch} \qquad \Gamma, x : t \vdash P : T}{\Gamma \vdash \alpha?(x : t).P : \alpha?t.T + \alpha?\neg t.0}$$

c-recv

$$\frac{\Gamma \vdash \alpha : \mathtt{Ch} \qquad \Gamma, x : \mathtt{Ch} \vdash P : T}{\Gamma \vdash \alpha?(\!|x|\!).P : \alpha?(x).T + \alpha?\neg\mathtt{Ch}.0}$$