Finite Equational Bases for CCS with Restriction

Paul van Tilburg
paul@luon.net

Supervisor: Bas Luttik

Department of Mathematics and Computer Science
Eindhoven University of Technology

June 5, 2007

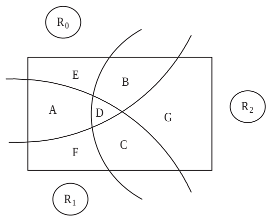
Outline

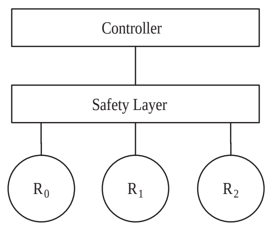
Introduction

- CCS: Calculus of Communicating Systems – a process algebra
- Developed by Robert Milner in late seventies
- Process Algebras:
  - Describing processes
  - Formal language
  - Transition systems
- Other process algebras: CSP and ACP

- Axiomatisations – starting point proving properties of modelled process

An Application





Process Terms

Definitions

$\mathcal{A}$: Set of actions ($move(r)$, $send(d)$, a, b, c)

$\mathcal{V}$: Set of variables (x, y, z)

$\mathcal{T}$: Set of terms, generated by:

$$\mathrm{T} ::= 0 \mid x \mid a.\mathrm{T} \mid \mathrm{T}+\mathrm{T} \mid \mathrm{T} \parallel \mathrm{T} \mid \mathrm{T} \setminus H$$

$(a.0,\ a.0 + b.0,\ a.(x \setminus \{a\}) \parallel b.y,\ p,\ q,\ r)$

$\mathcal{T}^{\circ}$: Closed terms – terms without variables

Process Term Semantics

Operational Rules

$$1\ \frac{}{a.p \xrightarrow{a} p} \qquad 2\ \frac{p \xrightarrow{a} p'}{p+q \xrightarrow{a} p'} \qquad 3\ \frac{q \xrightarrow{a} q'}{p+q \xrightarrow{a} q'}$$

Example

- Receiver $R = \sum_{d \in D} recv(d).comm(d).R$

$$R \xrightarrow{recv(e)} comm(e).R \xrightarrow{comm(e)} R$$

- Sender: $S = \sum_{d \in D} \overline{comm(d)}.send(d).S$

$$S \xrightarrow{\overline{comm(f)}} send(f).S \xrightarrow{send(f)} S$$

Process Term Semantics – Parallelism

Operational Rules

$$5\ \frac{p \xrightarrow{a} p'}{p \parallel q \xrightarrow{a} p' \parallel q} \qquad 6\ \frac{q \xrightarrow{a} q'}{p \parallel q \xrightarrow{a} p \parallel q'} \qquad 7\ \frac{p \xrightarrow{a} p' \quad q \xrightarrow{\bar{a}} q'}{p \parallel q \xrightarrow{\tau} p' \parallel q'}$$

Example

- Sender-Receiver $R \parallel S$

$$R \parallel S \xrightarrow{recv(d)} comm(d).R \parallel \overline{comm(d)}.send(d).S$$

$$\xrightarrow{\tau} R \parallel send(d).S \xrightarrow{send(d)} R \parallel S$$

- However:

$$R \parallel S \xrightarrow{\overline{comm(d)}} R \parallel send(d).S \text{ or } comm(d).R \parallel S \xrightarrow{comm(d)} R \parallel S$$

Process Term Semantics – Restriction

Operational Rule

$$4\ \frac{p \xrightarrow{a} p' \quad a, \bar{a} \notin H}{p \setminus H \xrightarrow{a} p' \setminus H}$$

Example

- Block communication actions with $H = \{comm(d) \mid d \in D\}$
- Sender-Receiver $(R \parallel S) \setminus H$

$$(R \parallel S) \setminus H \xrightarrow{recv(d)} (comm(d).R \parallel \overline{comm(d)}.send(d).S) \setminus H$$

$$\xrightarrow{\tau} (R \parallel send(d).S) \setminus H \xrightarrow{send(d)} (R \parallel S) \setminus H$$

Bisimulation

Definition
Bisimulation $p \leftrightarrows q$:
if $p \xrightarrow{a} p'$, then there exists q' such that $q \xrightarrow{a} q'$ and $p' \leftrightarrows q'$

Examples

- Bisimilar: $a.\mathbf{0} + a.\mathbf{0} \leftrightarrows a.\mathbf{0}$
- Not bisimilar: $door.(lady.\mathbf{0} + tiger.\mathbf{0}) \not\leftrightarrows door.lady.\mathbf{0} + door.tiger.\mathbf{0}$.

Process Algebra

- Elements, called *processes*: $\mathcal{P}^\circ / \leftrightarrows$
- Operators:
  - $\mathbf{0}$
  - $a.\_$ (for each $a \in \mathcal{A}$)
  - $\_ \setminus H$ (for each $H \subset \mathcal{A}$)
  - $\_ + \_$
- Axioms:

$$\begin{array}{lll} \text{(A1)} & x + y & \approx y + x \\ \text{(A2)} & (x+y)+z & \approx x + (y+z) \\ \text{(A3)} & x + x & \approx x \\ \text{(A4)} & x + \mathbf{0} & \approx x \end{array}$$

Equational Theory

Definitions

- *Process equation* $p \approx q$: pair of process terms
- *Valid* process equation $p \approx q$: $[\![p]\!]_\nu = [\![q]\!]_\nu$ for all $\nu : \mathcal{V} \to \mathbf{P}$
- *Equational base*: set of valid equations from which all other valid equations can be derived
- An equational base is *finite* if it contains a finite number of axioms

Equational Theory (2)

Definition
Soundness:
if $p \approx q$ derivable, then $p \leftrightarrows q$

Example
$x + x \approx x$

Definition
Completeness:
if $p \leftrightarrows q$, then $p \approx q$ derivable

Example
$a.y + a.y \leftrightarrows a.y$

Problem Statement

- Find axiomatisation for CCS with parallelism and restriction
- Prove soundness
- Prove completeness
  - $p \leftrightarrows q$ means $[\![p]\!]_\nu = [\![q]\!]_\nu$ for all ν
  - show that if $p \not\approx q$, then there exists $*$ such that $[\![p]\!]_* \neq [\![q]\!]_*$
  - $*$ is called a *distinguishing valuation*
- Incremental steps

A Basic Process Algebra: $\mathbf{P}$

- Process terms $\mathcal{P}$:

$$\mathrm{P} \ ::= \ \mathbf{0} \ | \ x \ | \ a.\mathrm{P} \ | \ \mathrm{P} + \mathrm{P} \ | \ \mathrm{P} \setminus H$$

- Process algebra $\mathbf{P}$:
  - Based on $\mathcal{P} / \leftrightarrows$
  - Basic fragment of CCS
  - No parallelism, just restriction

Equational Base for $\mathbf{P}$

$$\begin{array}{llll} \text{(A1)} & x + y & \approx y + x & \\ \text{(A2)} & (x+y)+z & \approx x + (y+z) & \\ \text{(A3)} & x + x & \approx x & \\ \text{(A4)} & x + \mathbf{0} & \approx x & \\ & & & \\ \text{(D1)} & \mathbf{0} \setminus H & \approx \mathbf{0} & \\ \text{(D2)} & a.x \setminus H & \approx \mathbf{0} & \text{if } a \in H \\ \text{(D3)} & a.x \setminus H & \approx a.(x \setminus H) & \text{otherwise} \\ \text{(D4)} & (x+y) \setminus H & \approx x \setminus H + y \setminus H & \\ & & & \\ \text{(DX1)} & x \setminus \emptyset & \approx x & \\ \text{(DX2)} & x \setminus \mathcal{A} & \approx \mathbf{0} & \\ \text{(DX3)} & (x \setminus H) \setminus J & \approx x \setminus (H \cup J) & \end{array}$$

Normal Forms

- *Normal form*: most basic, “smallest” form of a process term (6 is the normal form of $3 + 3$, or $1 + 5$, etc.)
- Result of applying axioms that work towards small terms
- Simplifies completeness proof
  - If p has normal form s and q normal form t, then $s \approx p \approx q \approx t$
  - Indicates smallest elements to investigate

Normal Forms (2)

Normal forms of $\mathcal{P}$, generated by

$$N ::= 0 \mid a.N \mid N + N \mid x \setminus H$$

Remark
$H \neq \mathcal{A}$, but H can be $\emptyset$

Examples
$0 \quad a.0 + b.0 \quad a.x \quad x \setminus \{b\}$

Definition
Simple normal forms: $a.N$ and $x \setminus H$

Distinguishing Valuation

Valuation requirements

- Distinguish between simple normal forms $a.t$ and $x \setminus H$
- Based on notion of *length*:
  longest number of steps a process can take
  - A variable can not take any steps
  - Length of $a.0 + a.a.0$ is 2;
    length of $a.(x \setminus \{b\})$ is 1
  - Bisimulation preserves length
- Should have properties that survive restriction

Distinguishing Valuation $*_m$

Definition
For each $x \in \mathcal{V}$, and $m \geq 1$, injective function $\lceil\cdot\rceil : \mathcal{V} \to (\mathbb{N} - \{0\})$:

$$*_m(x) = \sum_{a \in \mathcal{A}} a.\psi_{\lceil x \rceil \cdot m} \text{ with } \psi_i = \sum_{a \in \mathcal{A}} a^i.0$$

Example
Choose $\mathcal{A} = \{a, b\}, \mathcal{V} = \{x\}, t = a.a.0 + x \setminus \{b\}$
Length of t is 2, choose $m = 2$, $\lceil x \rceil = 1$. Then:

$$*_m(x) = a.\psi_2 + b.\psi_2 = a.(a.a.0 + b.b.0) + b.(a.a.0 + b.b.0),$$

and $[\![t]\!]_{*_m} = a.a.0 + a.a.a.0$

Distinguishing Valuation $*_m$ (2)

Example
Given $*_m(x) = a.(a.a.0 + b.b.0) + b.(a.a.0 + b.b.0)$, consider:

$$[\![x \setminus \{a\} + x \setminus \{b\}]\!]_{*_m} = b.b.b.0 + a.a.a.0$$
$$[\![x \setminus \emptyset]\!]_{*_m} = a.(a.a.0 + b.b.0) + b.(a.a.0 + b.b.0)$$

Valuation properties

- If m is chosen equal or greater than the length, difference between prefix and variable can be detected
- Because of injective function variable can be found
- Residual shows which restriction was applied

Adding Parallelism: $\mathbf{P}_F$

- Process terms $\mathcal{P}_F$:

$$P ::= 0 \mid x \mid a.P \mid P + P \mid P \setminus H \mid P \mathbin{|\!\lfloor} P \mid P \parallel P$$

- Process algebra $\mathbf{P}_F$:
  - Based on $\mathbf{P}$
  - Parallelism
  - Interleaving, no communication

Equational Base for $\mathbf{P}_F$

(L1)	$0 \mathbin{\|\!\lfloor} x$	≈ 0
(L2)	$a.x \mathbin{\|\!\lfloor} y$	$\approx a.(x \parallel y)$
(L3)	$(x + y) \mathbin{\|\!\lfloor} z$	$\approx x \mathbin{\|\!\lfloor} z + y \mathbin{\|\!\lfloor} z$
(L4)	$(x \mathbin{\|\!\lfloor} y) \mathbin{\|\!\lfloor} z$	$\approx x \mathbin{\|\!\lfloor} (y \parallel z)$
(L5)	$x \mathbin{\|\!\lfloor} 0$	$\approx x$
(D5)	$(x \mathbin{\|\!\lfloor} y) \setminus H$	$\approx x \setminus H \mathbin{\|\!\lfloor} y \setminus H$
(M)	$x \parallel y$	$\approx x \mathbin{\|\!\lfloor} y + y \mathbin{\|\!\lfloor} x$

Normal Forms

Normal forms of $\mathcal{P}_F$, generated by

$$N ::= 0 \mid a.N \mid N + N \mid (x \setminus H) \mathbin{|\!\lfloor} N$$

Examples
$x \mathbin{|\!\lfloor} (a.0 + b.0) \quad a.(x \setminus \{a\}) \mathbin{|\!\lfloor} (y \setminus \{b\})$

Definition
Simple normal forms: $a.N$ and $(x \setminus H) \mathbin{|\!\lfloor} N$

Distinguishing Valuation

Valuation requirements

- Distinguish between simple normal forms $a.t$ and $(x \setminus H) \mathbin{|\!\lfloor} t$
- Length is not enough, consider $a.a.0$ and $a.0 \mathbin{|\!\lfloor} a.0$
- Based on notion of *branching degree*:
  number of choices for a process that lead to another unique process
  - Branching degree of $a.0 + a.(0 + 0) + a.a.0$ is 2
  - Branching degree respects bisimulation

Adding Communication: $\mathbf{P}_H$

- Process terms:

$$\mathrm{P} ::= \mathbf{0} \mid x \mid a.\mathrm{P} \mid \mathrm{P}+\mathrm{P} \mid \mathrm{P}\setminus H \mid \mathrm{P} \mathbin{⌊\!⌊} \mathrm{P} \mid \mathrm{P} \,|\, \mathrm{P} \mid \mathrm{P} \parallel \mathrm{P}$$

- Operational rules:

$$\frac{p \xrightarrow{a} p' \quad q \xrightarrow{\bar{a}} q'}{p \mid q \xrightarrow{\tau} p' \parallel q'} \qquad \frac{p \xrightarrow{a} p' \quad q \xrightarrow{\bar{a}} q'}{p \parallel q \xrightarrow{\tau} p' \parallel q'}$$

- Process algebra $\mathbf{P}_H$:
  - Based on $\mathbf{P}_F$
  - Communication

Equational Base for $\mathbf{P}_H$

(C1)	$\mathbf{0} \mid x$	$\approx \mathbf{0}$	
(C2)	$a.x \mid b.y$	$\approx \tau.(x \parallel y)$	if $b = \bar{a}$
(C3)	$a.x \mid b.y$	$\approx \mathbf{0}$	otherwise
(C4)	$(x+y) \mid z$	$\approx x \mid z + y \mid z$	
(C5)	$x \mid y$	$\approx y \mid x$	
(C6)	$(x \mid y) \mid z$	$\approx x \mid (y \mid z)$	
(C7)	$(x \mathbin{⌊\!⌊} y) \mid z$	$\approx (x \mid z) \mathbin{⌊\!⌊} y$	
(M)	$x \parallel y$	$\approx x \mathbin{⌊\!⌊} y + y \mathbin{⌊\!⌊} x + x \mid y$	
(H)	$x \mid (y \mid z)$	$\approx \mathbf{0}$	

Problems with Normal Forms

No distribution axiom for $\mathbin{⌊\!⌊}$

$$\text{(D5)} \quad (x \mathbin{⌊\!⌊} y)\setminus H \approx x\setminus H \mathbin{⌊\!⌊} y\setminus H$$

Example

Choose $H = \{b\}$, $x = a.b.\mathbf{0}$, $y = \bar{b}.c.\mathbf{0}$, then

- Left-hand side:
  $(a.b.\mathbf{0} \mathbin{⌊\!⌊} \bar{b}.c.\mathbf{0})\setminus\{b\} \approx a.((b.\mathbf{0} \parallel \bar{b}.c.\mathbf{0})\setminus\{b\}) \approx a.\tau.c.\mathbf{0}$
- Right-hand side:
  $(a.b.\mathbf{0}\setminus\{b\}) \mathbin{⌊\!⌊} (\bar{b}.c.\mathbf{0}\setminus\{b\}) \approx a.\mathbf{0} \mathbin{⌊\!⌊} \mathbf{0} \approx a.\mathbf{0}$

However, $a.\tau.c.\mathbf{0} \not\approx a.\mathbf{0}$

Problems with Normal Forms (2)

No distribution axiom for $|$

$$\text{(D6)} \quad (x \mid y)\setminus H \approx (x\setminus H) \mid (y\setminus H)$$

Example

Choose $H = \{a\}$, $x = a.b.\mathbf{0}$, $y = \bar{a}.c.\mathbf{0}$, then

- Left-hand side:
  $(a.b.\mathbf{0} \mid \bar{a}.c.\mathbf{0})\setminus\{a\} \approx \tau.(b.c.\mathbf{0} + c.b.\mathbf{0})$
- Right-hand side:
  $(a.b.\mathbf{0}\setminus\{a\}) \mid (\bar{a}.c.\mathbf{0}\setminus\{a\}) \approx \mathbf{0}$

However, $\tau.(b.c.\mathbf{0} + c.b.\mathbf{0}) \not\approx \mathbf{0}$

Alphabet Axioms

- Pushing restrictions partially inward
- Based on the *alphabet* of a process

$$\begin{array}{ll}
\text{(DL1)} & (x \mathbin{⌊\!⌊} (y\setminus H))\setminus H \approx (x\setminus H) \mathbin{⌊\!⌊} (y\setminus H) \\
\text{(DL2)} & ((x\setminus H) \mathbin{⌊\!⌊} y)\setminus H \approx (x\setminus H) \mathbin{⌊\!⌊} (y\setminus H) \\
\\
\text{(DC1)} & (x \mid (y\setminus H))\setminus H \approx (x\setminus H) \mid (y\setminus H)
\end{array}$$

Example

$(x \mid y\setminus\{a\})\setminus\{a,b\} \approx (x\setminus\{a\} \mid y\setminus\{a\})\setminus\{b\}$

Problems with Normal Forms and Distinguishing

- Number and structure of normal forms not known yet
- Difficult to determine which restriction has which effect in:

$$\Big(\big(((x\setminus H \mathbin{⌊\!⌊} y)\setminus J) \mathbin{⌊\!⌊} z\big)\setminus M\Big) \mathbin{⌊\!⌊} p$$

- Are there more alphabet axioms?

Concluding Remarks

Results

- Established equational base for $\mathbf{P}$; proved sound & complete
- Established equational base for $\mathbf{P}_F$; proved sound & complete
- Explored equational base for $\mathbf{P}_H$:
  - Identified problems
  - Proposed possible solutions

Thanks

Bas Luttik, Jos Baeten, Michel Reniers, the department, family & friends

Questions

Any Questions?