True concurrency equivalence semantics: an overview

Vashti Galpin

vcg@dcs.ed.ac.uk

Department of Computer Science
University of Edinburgh
Scotland

Introduction

- concurrency
- process domains
  - Petri nets
  - event structures
  - labelled transition systems
- process algebras
  - syntax
  - operational semantics
  - equivalence semantics
- true concurrency equivalence semantics

$$a \mid b \neq a.b.nil + b.a.nil$$

Labelled transition system $(\mathcal{S}, \mathcal{A}, \{\stackrel{a}{\to} \subseteq \mathcal{S} \times \mathcal{S} \mid a \in \mathcal{A}\})$

- $\mathcal{S}$ – set of states
- $\mathcal{A}$ – set of transition labels, actions
- relations $\stackrel{a}{\to}$ describe which transitions occur between states.
- write $s \stackrel{a}{\to} s'$ for $(s, s') \in \stackrel{a}{\to}$
- no structure on states or actions – pure labelled transition system

CCS syntax $P ::= nil \mid \alpha.P \mid P + P \mid P|P \mid P\backslash L \mid P[f]$

- $\alpha \in Act = \{a, b, c, \ldots, \overline{a}, \overline{b}, \overline{c}, \ldots\} \cup \tau$
- $L \subset \mathcal{L} = \{a, b, c, \ldots, \overline{a}, \overline{b}, \overline{c}, \ldots\}$
- f, relabelling function such that $f(\overline{\ell}) = \overline{f(\ell)}$ and $f(\tau) = \tau$
- $\mathcal{P}$ denotes the set of processes generated by this syntax

Operational semantics for CCS

(ST1)	$\alpha.P \stackrel{\alpha}{\to} P$	$\alpha \in Act$	
(ST2)	$P \stackrel{\alpha}{\to} P'$	implies	$P + Q \stackrel{\alpha}{\to} P'$ $Q + P \stackrel{\alpha}{\to} P'$
(ST3)	$P \stackrel{\alpha}{\to} P'$	implies	$P \mid Q \stackrel{\alpha}{\to} P' \mid Q$ $Q \mid P \stackrel{\alpha}{\to} Q \mid P'$
(ST4)	$P \stackrel{\alpha}{\to} P'$, $Q \stackrel{\overline{\alpha}}{\to} Q'$	implies	$P \mid Q \stackrel{\tau}{\to} P' \mid Q'$
(ST5)	$P \stackrel{\alpha}{\to} P'$	implies	$P[f] \stackrel{f(\alpha)}{\to} P'[f]$
(ST6)	$P \stackrel{\alpha}{\to} P'$	implies	$P\backslash L \stackrel{\alpha}{\to} P'\backslash L \quad \alpha, \overline{\alpha} \notin L$

Examples

$$
\begin{array}{lll}
a.nil + \overline{a}.nil & \stackrel{a}{\rightarrow} & nil \\
 & \stackrel{\overline{a}}{\rightarrow} & nil \\
a.nil \mid \overline{a}.nil & \stackrel{a}{\rightarrow} & nil \mid \overline{a}.nil \stackrel{\overline{a}}{\rightarrow} nil \mid nil \\
 & \stackrel{\overline{a}}{\rightarrow} & a.nil \mid nil \stackrel{a}{\rightarrow} nil \mid nil \\
 & \stackrel{\tau}{\rightarrow} & nil \mid nil \\
(a.nil \mid \overline{a}.nil)\backslash a & \stackrel{\tau}{\rightarrow} & nil \mid nil
\end{array}
$$

Notation

Use $\Longrightarrow$ for $(\stackrel{\tau}{\rightarrow})^n, n \geq 0$
Use $\stackrel{a}{\Longrightarrow}$ for $\Longrightarrow\stackrel{a}{\rightarrow}\Longrightarrow$
Use $\stackrel{m}{\Longrightarrow}$ where $m = a_1.a_2.\ldots.a_k, k \geq 0$ for $\stackrel{a_1}{\Longrightarrow}\stackrel{a_2}{\Longrightarrow}\cdots\stackrel{a_k}{\Longrightarrow}$

Observational equivalence

Consider the labelled transition system $(\mathcal{P}, \mathcal{L}^*, \{\stackrel{m}{\Longrightarrow} \subseteq \mathcal{P} \times \mathcal{P} \mid m \in \mathcal{L}^*\})$

A **(weak) bisimulation** is a binary relation $\mathcal{R} \subseteq \mathcal{P} \times \mathcal{P}$ such that $(P, Q) \in \mathcal{R}$ if for all $m \in \mathcal{L}^*$

1. whenever $P \stackrel{m}{\Longrightarrow} P'$, then there exists $Q' \in \mathcal{P}$ such that $Q \stackrel{m}{\Longrightarrow} Q'$ and $(P', Q') \in \mathcal{R}$
2. whenever $Q \stackrel{m}{\Longrightarrow} Q'$, then there exists $P' \in \mathcal{P}$ such that $P \stackrel{m}{\Longrightarrow} P'$ and $(P', Q') \in \mathcal{R}$.

Observational equivalence $\approx$ is the union of all weak bisimulations and is the largest weak bisimulation

Two processes are **observationally equivalent** if they occur as a pair in a weak bisimulation

Example

Consider

$$a.nil \mid b.nil$$

and

$$a.b.nil + b.a.nil$$

where $a \neq \bar{b}$. Then

$$\{(a.nil \mid b.nil, a.b.nil + b.a.nil), (a.nil \mid nil, a.nil),$$
$$(b.nil \mid nil, b.nil), (nil \mid nil, nil)\}.$$

is a weak bisimulation and

$$a.nil \mid b.nil \approx a.b.nil + b.a.nil$$

Syntax for CCS with locations

$$P ::= nil \mid u :: P \mid \alpha.P \mid P + P \mid P|P \mid P\backslash L \mid P[f]$$

- $u \in Loc^*$
- $\mathcal{P}_{Loc}$ denotes the set of processes generated by this syntax

Operational semantics for CCS with locations

(LT1) $a.P \xrightarrow[u]{a} u :: P \quad a \in \mathcal{L}, \quad u \in Loc^*$

(LT2) $P \xrightarrow[u]{a} P'$ implies $v :: P \xrightarrow[vu]{a} v :: P'$

(LT3) $P \xrightarrow[u]{a} P'$ implies $P + Q \xrightarrow[u]{a} P'$
$Q + P \xrightarrow[u]{a} P'$

(LT4) $P \xrightarrow[u]{a} P'$ implies $P \mid Q \xrightarrow[u]{a} P' \mid Q$
$Q \mid P \xrightarrow[u]{a} Q \mid P'$

(LT5) $P \xrightarrow[u]{a} P'$ implies $P[f] \xrightarrow[u]{f(a)} P'[f]$

(LT6) $P \xrightarrow[u]{a} P'$ implies $P \backslash L \xrightarrow[u]{a} P' \backslash L \quad \alpha, \overline{\alpha} \notin L$

Example

$a.b.nil \xrightarrow[u]{a} u :: b.nil \xrightarrow[uv]{b} u :: v :: nil$

Notation

Use $\overset{a}{\underset{u}{\Longrightarrow}}$ for $\Longrightarrow \xrightarrow[u]{a} \Longrightarrow$

Loose location bisimulation

Consider the modified labelled transition system

$(\mathcal{P}_{Loc}, \mathcal{L}, Loc, \{\overset{a}{\underset{u}{\Longrightarrow}} \subseteq \mathcal{P}_{Loc} \times \mathcal{P}_{Loc} \mid a \in \mathcal{L}, u \in Loc^*\} \cup \{\overset{\tau}{\Longrightarrow} \subseteq \mathcal{P}_{Loc} \times \mathcal{P}_{Loc}\})$

Loose location bisimulation (cont.)

A **loose location bisimulation** is a binary relation $\mathcal{R} \subseteq \mathcal{P}_{Loc} \times \mathcal{P}_{Loc}$ such that $(P, Q) \in \mathcal{R}$ iff

1. whenever $P \Longrightarrow P'$ then there exists $Q' \in \mathcal{P}_{Loc}$ such that $Q \Longrightarrow Q'$ and $(P', Q') \in \mathcal{R}$
2. whenever $Q \Longrightarrow Q'$ then there exists $P' \in \mathcal{P}_{Loc}$ such that $P \Longrightarrow P'$ and $(P', Q') \in \mathcal{R}$
3. whenever $P \overset{a}{\underset{u}{\Longrightarrow}} P'$ then there exists $Q' \in \mathcal{P}_{Loc}$ such that $Q \overset{a}{\underset{u}{\Longrightarrow}} Q'$ and $(P', Q') \in \mathcal{R}$.
4. whenever $Q \overset{a}{\underset{u}{\Longrightarrow}} Q'$ then there exists $P' \in \mathcal{P}_{Loc}$ such that $P \overset{a}{\underset{u}{\Longrightarrow}} P'$ and $(P', Q') \in \mathcal{R}$.

$\approx_l'$ is defined to be the largest loose location bisimulation

Example

$$(a.c.nil \mid \overline{c}.b.nil)\backslash\{c\} \not\approx_l' a.b.nil$$

Consider the following transitions for $l, m \in Loc$ and $l \neq m$

$$(a.c.nil \mid \overline{c}.b.nil)\backslash\{c\} \overset{a}{\underset{l}{\Longrightarrow}} (l :: nil \mid b.nil)\backslash\{c\} \overset{b}{\underset{m}{\Longrightarrow}} (l :: nil \mid m :: nil)$$

These must be matched

$$a.b.nil \overset{a}{\underset{l}{\Longrightarrow}} l :: b.nil$$

however the only transition that can be performed after that is

$$l :: b.nil \overset{b}{\underset{lu}{\Longrightarrow}} l :: u :: nil$$

for some $u \in Loc^*$ and it is not possible for lu to equal m.

Other true concurrency equivalences

- causal bisimulation: $P \xrightarrow{\langle a,B \rangle} P'$
- distributed bisimulation $P \xrightarrow{a} \langle P', P'' \rangle$.
- equivalences based on duration
  - ST-bisimulation: $a.P \xrightarrow{s(a)} f(a).P$ and $f(a).P \xrightarrow{f(a)} P$
- equivalences based on time
- equivalences based on structure of actions

Proposed research

- Comparison
  - in terms of CCS processes
  - in terms of labelled transition systems
- Application
  - how to determine which equivalence to use
  - properties

Comparison in terms of CCS processes

$\approx$	observational equivalence
$\approx_d$	distributed bisimulation
$\approx_l$	(dynamic) location bisimulation
$\approx_l^l$	loose location bisimulation
$\approx_l^s$	static location bisimulation
$\approx_c$	weak causal bisimulation
$\approx_{lc}$	local cause bisimulation
$\approx_{gc}$	global cause bisimulation
$\approx_{lg}$	local/global cause bisimulation
$\approx_{rw}$	read/write bisimulation

Comparison in terms of labelled transition system

- Modified labelled transition systems
- Disadvantages of comparison in terms of CCS processes
- More general approach to modified labelled transition systems
  - union
  - general labelled transition system
  - parameterised labelled transition system

Applicability of equivalences

- Justification and criteria for use
- Survey of usage
- Decidability and efficiency issues
- Properties
  - An equivalence $\asymp$ is said to distinguish
    location iff $(a.c.b \mid d.\bar{c}.e) \setminus c \not\asymp (a.c.e \mid d.\bar{c}.b) \setminus c$
    read-write causality iff $(a.c.b \mid d.\bar{c}.e) \setminus c \not\asymp (a.\bar{c}.b \mid d.c.e) \setminus c$
    concurrency iff $a \mid b \not\asymp a.b + b.a$
  - $\approx_l$, $\approx_l'$ and $\approx_l^s$ all distinguish location, but not read-write causality
  - $\approx_{rw}$ distinguishes read-write causality, but not location
  - $\approx_c$ doesn't distinguish location or read-write causality
  - All equivalences shown previously except $\approx$, distinguish concurrency.