Behavioural Equivalences for Co-operating Transactions

Matthew Hennessy

joint work with Vasileois Koutavas, Carlo Spaccasassi, Edsko de Vries

Concur, September 2015

lero THE IRISH SOFTWARE RESEARCH CENTRE

TRINITY COLLEGE DUBLIN
Coláiste na Tríonóide, Baile Átha Cliath

Outline

Co-operating Transactions what are they?

TransCCS

Behaviour

History bisimulations

Property logics

STM: Software Transactional Memory

- Database technology applied to software
- concurrency control: *atomic memory transactions*
- lock-free programming in multithreaded programmes
- threads run optimistically
- conflicts are automatically rolled back by system

Implementations:

- Haskell, OCaml, Csharp, Intel Haswell architecture

Issues:

- Language Design
- Implementation strategies
- Semantics what should happen when programs are run

STM: Software Transactional Memory

- Database technology applied to software
- concurrency control: *atomic memory transactions*
- lock-free programming in multithreaded programmes
- threads run optimistically
- conflicts are automatically rolled back by system

Implementations:

- Haskell, OCaml, Csharp, Intel Haswell architecture

Issues:

- Language Design
- Implementation strategies
- Semantics what should happen when programs are run

Standard Transactions on which STM is based

- Transactions provide *an abstraction for error recovery* in a concurrent setting.
- Guarantees:
  - Atomicity: Each transaction either runs in its entirety (commits) or not at all
  - Consistency: When faults are detected the transaction is automatically rolled-back
  - Isolation: The effects of a transaction are concealed from the rest of the system until the transaction commits
  - Durability: After a transaction commits, its effects are permanent
- Isolation:
  - Higher levels limit concurrency
  - Lower levels have implementation difficulties and precise semantic understanding

Standard Transactions on which STM is based

- Transactions provide *an abstraction for error recovery* in a concurrent setting.
- Guarantees:
  - Atomicity: Each transaction either runs in its entirety (commits) or not at all
  - Consistency: When faults are detected the transaction is automatically rolled-back
  - Isolation: The effects of a transaction are concealed from the rest of the system until the transaction commits
  - Durability: After a transaction commits, its effects are permanent
- Isolation:
  - Higher levels limit concurrency
  - Lower levels have implementation difficulties and precise semantic understanding

Communicating/Co-operating Transactions

- We *drop isolation completely*:
  - There is no limit on the co-operation/communication between a transaction and its environment.
  - There is no barrier to concurrency
  - Understanding the behaviour of these new transactional systems is problematic
- Should guarantee:
  - Atomicity: Each transaction will either run in its entirety or not at all
  - Consistency: When faults are detected the transaction is automatically rolled-back, *together with all effects of the transaction on its environment*
  - Durability: After *all transactions that have interacted* commit, their effects are permanent (coordinated checkpointing)

Communicating/Co-operating Transactions

- We *drop isolation completely*:
  - There is no limit on the co-operation/communication between a transaction and its environment.
  - There is no barrier to concurrency
  - Understanding the behaviour of these new transactional systems is problematic
- Should guarantee:
  - Atomicity: Each transaction will either run in its entirety or not at all
  - Consistency: When faults are detected the transaction is automatically rolled-back, *together with all effects of the transaction on its environment*
  - Durability: After *all transactions that have interacted* commit, their effects are permanent (coordinated checkpointing)

Programming with Co-operating Transactions

Add to your favourite programming language:

- `atomic`⟦. . . . . .⟧
- commands `commit` and `abort&retry`

Example: three-way rendezvous

$$P_1 \mid\mid P_2 \mid\mid P_3 \mid\mid P_4$$

Problem:

- P_i process/transaction subject to failure
- Some coalition of three from P_1, P_2, P_3, P_4 should decide to collaborate

Result:

- Each P_j in the successful coalition outputs id of its partners on channel out_j

Programming with Co-operating Transactions

Add to your favourite programming language:

- atomic⟦. . . . . .⟧
- commands commit and abort&retry

Example: three-way rendezvous

$$P_1 \mid\mid P_2 \mid\mid P_3 \mid\mid P_4$$

Problem:

- P_i process/transaction subject to failure
- Some coalition of three from P_1, P_2, P_3, P_4 should decide to collaborate

Result:

- Each P_j in the successful coalition outputs id of its partners on channel out_j

Programming with Co-operating Transactions

Add to your favourite programming language:

- atomic⟦......⟧
- commands commit and abort&retry

Example: three-way rendezvous

$$P_1 \mid\mid P_2 \mid\mid P_3 \mid\mid P_4$$

Problem:

- P_i process/transaction subject to failure
- Some coalition of three from P_1, P_2, P_3, P_4 should decide to collaborate

Result:

- Each P_j in the successful coalition outputs id of its partners on channel out_j

Example: three-way rendezvous

$$P_1 \,||\, P_2 \,||\, P_3 \,||\, P_4$$

Algorithm for P_n:

- Broadcast id n randomly to two arbitrary partners
  $b!\langle n\rangle \mid b!\langle n\rangle$
- Receive ids from two random partners $b?(y).b?(z)$
- Propose coalition with these partners $s_y!\langle n, z\rangle .s_z!\langle n, y\rangle$
- Confirm that partners are in agreement:
  - if YES, `commit` and report
  - if NO, `abort&retry`

TRINITY COLLEGE DUBLIN
Coláiste na Tríonóide, Baile Átha Cliath

Example: three-way rendezvous

$$P_1 \,||\, P_2 \,||\, P_3 \,||\, P_4$$

Algorithm for P_n:

- Broadcast id n randomly to two arbitrary partners
  $b!\langle n\rangle \mid b!\langle n\rangle$
- Receive ids from two random partners $b?(y)\,.b?(z)$
- Propose coalition with these partners $s_y!\langle n, z\rangle\,.s_z!\langle n, y\rangle$
- Confirm that partners are in agreement:
  - if YES, `commit` and report
  - if NO, `abort&retry`

Example: three-way rendezvous

$$P_1 \,||\, P_2 \,||\, P_3 \,||\, P_4$$

$$\begin{aligned}
P_n \quad \Leftarrow \quad & b!\langle n\rangle \mid b!\langle n\rangle \mid \\
& \texttt{atomic}[\![b?(y)\,.b?(z)\,. \\
& \qquad s_y!\langle n, z\rangle\,.s_z!\langle n, y\rangle\,. \qquad \text{proposing} \\
& \qquad s_n?(y_1, z_1)\,.s_n?(y_2, z_2)\,. \qquad \text{confirming} \\
& \qquad \texttt{if}\ \{y, z\} = \{y_1, z_1\} = \{y_2, z_2\} \\
& \qquad\quad \texttt{then}\ \texttt{commit} \mid \mathsf{out}_n!\langle y, z\rangle \\
& \qquad\quad \texttt{else}\ \texttt{abrt\&retry}\ ]\!]
\end{aligned}$$

Co-operating Transactions: Issues

- Language Design and Implementation
  - Transaction Synchronisers (Luchangco et al 2005)
  - cJoin with commits Bruni, Melgratti, Montanari ENTCS 2004
  - Transactional Events for ML (Fluet, Grossman et al. ICFP 2008)
  - Communication Memory Transactions (Lesani, Palsberg PPoPP 2011)
  - ... Abstractions for Concurrent Consensus (Spaccasassi, Koutavas, Trends in Functional Programming 2013)
  - ......
- Semantics what should happen when programs are run
  - Topic of todays talk

Approach:

- Take a well-studied small language, with well understood semantic theory: *CCS*
- extend with transactional constructs
- extend existing semantic theory

Co-operating Transactions: Issues

- Language Design and Implementation
  - Transaction Synchronisers (Luchangco et al 2005)
  - cJoin with commits Bruni, Melgratti, Montanari ENTCS 2004
  - Transactional Events for ML (Fluet, Grossman et al. ICFP 2008)
  - Communication Memory Transactions (Lesani, Palsberg PPoPP 2011)
  - . . . Abstractions for Concurrent Consensus (Spaccasassi, Koutavas, Trends in Functional Programming 2013)
  - . . . . . .
- Semantics what should happen when programs are run
  - Topic of todays talk

Approach:

- Take a well-studied small language, with well understood semantic theory: *CCS*
- extend with transactional constructs
- extend existing semantic theory

CCS

Syntax: P, Q	$::=$	$\sum \mu_i.P_i$	guarded choice	$\mu_i \in Act_\tau$
	$\mid$	$P \mid Q$	parallel	
	$\mid$	$\nu a.P$	hiding	
	$\mid$	$\mathtt{rec}X.P$	recursion	

Minimal concurrent programming/specification language:

- Act_τ: abstract actions supporting communication/co-operation
- Concurrency: $P \mid Q$: independent concurrent processes
- Local resources: $\nu a.P$: action a is local to P
- Iteration/Recursion: $\mathtt{rec}X.P$

$a \in Act$ ← needs co-operation of → $\overline{a} \in Act$

CCS

$$
\text{Syntax:} \quad P, Q \; ::= \; \sum \mu_i.P_i \quad \text{guarded choice} \qquad \mu_i \in Act_\tau
$$
$$
\mid \; P \mid Q \quad \text{parallel}
$$
$$
\mid \; \nu a.P \quad \text{hiding}
$$
$$
\mid \; \mathtt{rec} X.P \quad \text{recursion}
$$

Minimal concurrent programming/specification language:

- Act_τ: abstract actions supporting communication/co-operation
- Concurrency: $P \mid Q$: independent concurrent processes
- Local resources: $\nu a.P$: action a is local to P
- Iteration/Recursion: $\mathtt{rec} X.P$

$a \in Act$ ← needs co-operation of → $\bar{a} \in Act$

CCS: Executing processes: $P \to Q$ Reduction semantics:

- Co-operation/Communication:

$$\text{(R-COMM)} \quad \sum \mu_i.P_i \mid \sum \nu_j.Q_j \to P_i \mid Q_j \qquad \text{if } \nu_j = \overline{\mu_i}$$

- Contextual rules:

$$\text{(R-PAR)} \quad \frac{P \to P'}{P \mid Q \to P' \mid Q} \qquad\qquad \text{(R-NEW)} \quad \frac{P \to P'}{\nu a.P \to \nu a.P'}$$

- Housekeeping rules:

$$\text{(R-REC)} \quad \texttt{rec}X.P \to P\{\texttt{rec}X.P/X\}$$

$TCCS^m$

Fossacs 2014

Syntax: $P, Q \;::=\; CCS$ syntax

$\mid$	$\dots$	
$\mid$	$[\![P \triangleright_k Q]\!]$	running transaction named k
$\mid$	$co.P$	commit
$\mid$	$[\![P \blacktriangleright Q]\!]$	uninitiated transaction

Transaction $[\![P \triangleright_k Q]\!]$:

- execute P to completion (commit)
- subject to random aborts
- if aborted, roll back environmental impact of P and initiate Q

Simplification: in $[\![P \triangleright_k Q]\!]$ bodies P and Q do not contain active transactions

$TCCS^m$

Fossacs 2014

Syntax:	P, Q	$::=$	CCS syntax	
		$\mid$	$\dots$	
		$\mid$	$[\![P \triangleright_k Q]\!]$	running transaction named k
		$\mid$	$\mathsf{co}.P$	commit
		$\mid$	$[\![P \blacktriangleright Q]\!]$	uninitiated transaction

Transaction $[\![P \triangleright_k Q]\!]$:

- execute P to completion (commit)
- subject to random aborts
- if aborted, roll back environmental impact of P and initiate Q

Simplification: in $[\![P \triangleright_k Q]\!]$ bodies P and Q do not contain active transactions

Examples

$\llbracket a.b.\mathrm{co} \rhd_k \mathbf{0} \rrbracket$

$\nu p.\llbracket a.\mathrm{co}.p \rhd_{k_1} \mathbf{0} \rrbracket \mid \llbracket \overline{p}.b.\mathrm{co} \rhd_{k_2} \mathbf{0} \rrbracket$

$\mu X.\llbracket a.(b.\mathrm{co} + c.\mathrm{co}) \rhd_k X \rrbracket$

$\mu X.\llbracket a.b.\mathrm{co} + a.c.\mathrm{co}) \rhd_k X \rrbracket$

$\mu X.\llbracket a.b.\mathrm{co} \rhd_k X \rrbracket$

$\mu X.\llbracket a.b.\mathrm{co} + a.c.\mathbf{0}) \rhd_k X \rrbracket$

$\llbracket a.\mathrm{co} \rhd_{k_1} \mathbf{0} \rrbracket \mid \llbracket b.\mathrm{co} \rhd_{k_2} \mathbf{0} \rrbracket$

$\nu p.\overline{p} \mid \llbracket a.p.\mathrm{co}.\overline{p} \rhd_{k_1} \mathbf{0} \rrbracket \mid \llbracket b.p.\mathrm{co}.\overline{p} \rhd_{k_2} \mathbf{0} \rrbracket$

$\llbracket a.b.\mathrm{co} + b.a.\mathrm{co} \rhd_k \mathbf{0} \rrbracket$

$\nu p.\llbracket a.p.\mathrm{co} \rhd_{k_1} \mathbf{0} \rrbracket \mid \llbracket b.\overline{p}.\mathrm{co} \rhd_{k_2} \mathbf{0} \rrbracket$

Executing Transactions: $P \to Q$ reduction semantics

- Co-operation/Communication
- Contextual rules
- Housekeeping rules
- aborts/commits eg. $[\![P \triangleright_k Q]\!] \to Q$ random abort
- roll back management

Executing Transactions: $P \rightarrow Q$ reduction semantics

- Co-operation/Communication
- Contextual rules
- Housekeeping rules
- aborts/commits eg. $[\![P \triangleright_k Q]\!] \rightarrow Q$ random abort
- roll back management

Co-operation/Communication

Co-operation means shared destiny:

$$[\![p_1.a_1.a_2.\mathsf{co} \rhd_{l_1} a]\!] \mid [\![\overline{p_1}.\mathsf{co}.c + \overline{p_2}.\mathsf{co}.c \rhd_{l} c]\!] \mid [\![p_2.b_1.b_2.\mathsf{co} \rhd_{l_2} b]\!]$$

$\rightarrow$

$$[\![a_1.a_2.\mathsf{co} \rhd_{k} a]\!] \mid [\![\mathsf{co}.c \rhd_{k} c]\!] \mid [\![p_2.b_1.b_2.\mathsf{co} \rhd_{l_2} b]\!]$$

l_1, l both succeed together, or both fail

$\rightarrow$

$$[\![p_1.a_1.a_2.\mathsf{co} \rhd_{l_1} a]\!] \mid [\![\mathsf{co}.c \rhd_{k} c]\!] \mid [\![b_1.b_2.\mathsf{co} \rhd_{k} b]\!]$$

l_2, l both succeed together, or both fail

- shared destiny via fresh renaming of transactions
- shared destiny via distributed transactions

Co-operation/Communication

Co-operation means shared destiny:

$$[\![p_1.a_1.a_2.\mathrm{co} \rhd_{l_1} a]\!] \mid [\![\overline{p_1}.\mathrm{co}.c + \overline{p_2}.\mathrm{co}.c \rhd_{l} c]\!] \mid [\![p_2.b_1.b_2.\mathrm{co} \rhd_{l_2} b]\!]$$

$$\rightarrow$$

$$[\![a_1.a_2.\mathrm{co} \rhd_{k} a]\!] \mid [\![\mathrm{co}.c \rhd_{k} c]\!] \mid [\![p_2.b_1.b_2.\mathrm{co} \rhd_{l_2} b]\!]$$

l_1, l both succeed together, or both fail

$$\rightarrow$$

$$[\![p_1.a_1.a_2.\mathrm{co} \rhd_{l_1} a]\!] \mid [\![\mathrm{co}.c \rhd_{k} c]\!] \mid [\![b_1.b_2.\mathrm{co} \rhd_{k} b]\!]$$

l_2, l both succeed together, or both fail

- shared destiny via fresh renaming of transactions
- shared destiny via distributed transactions

Co-operation/Communication

Co-operation means shared destiny:

$$[\![p_1.a_1.a_2.\mathrm{co} \triangleright_{l_1} a]\!] \mid [\![\overline{p_1}.\mathrm{co}.c + \overline{p_2}.\mathrm{co}.c \triangleright_{l} c]\!] \mid [\![p_2.b_1.b_2.\mathrm{co} \triangleright_{l_2} b]\!]$$

$\rightarrow$

$$[\![a_1.a_2.\mathrm{co} \triangleright_{k} a]\!] \mid [\![\mathrm{co}.c \triangleright_{k} c]\!] \mid [\![p_2.b_1.b_2.\mathrm{co} \triangleright_{l_2} b]\!]$$

l_1, l both succeed together, or both fail

$\rightarrow$

$$[\![p_1.a_1.a_2.\mathrm{co} \triangleright_{l_1} a]\!] \mid [\![\mathrm{co}.c \triangleright_{k} c]\!] \mid [\![b_1.b_2.\mathrm{co} \triangleright_{k} b]\!]$$

l_2, l both succeed together, or both fail

- shared destiny via fresh renaming of transactions
- shared destiny via distributed transactions

Co-operation/Communication

Co-operation means shared destiny:

$$[\![p_1.a_1.a_2.\mathrm{co} \rhd_{l_1} a]\!] \mid [\![\overline{p_1}.\mathrm{co}.c + \overline{p_2}.\mathrm{co}.c \rhd_{l} c]\!] \mid [\![p_2.b_1.b_2.\mathrm{co} \rhd_{l_2} b]\!]$$

$$\rightarrow$$

$$[\![a_1.a_2.\mathrm{co} \rhd_{k} a]\!] \mid [\![\mathrm{co}.c \rhd_{k} c]\!] \mid [\![p_2.b_1.b_2.\mathrm{co} \rhd_{l_2} b]\!]$$

l_1, l both succeed together, or both fail

$$\rightarrow$$

$$[\![p_1.a_1.a_2.\mathrm{co} \rhd_{l_1} a]\!] \mid [\![\mathrm{co}.c \rhd_{k} c]\!] \mid [\![b_1.b_2.\mathrm{co} \rhd_{k} b]\!]$$

l_2, l both succeed together, or both fail

- shared destiny via fresh renaming of transactions
- shared destiny via distributed transactions

Co-operation/Communication: reduction semantics

- Communication:

(R-COMM)

$$\left[\!\left[R_1 \mid \sum \mu_i P_i \rhd_{l_1} - \right]\!\right] \mid \left[\!\left[R_2 \mid \sum \nu_j Q_j \rhd_{l_2} - \right]\!\right]$$

$$\rightarrow$$

$$[\![R_1 \mid P_i \rhd_k -]\!] \mid [\![R_2 \mid Q_j \rhd_k -]\!] \qquad \text{if } \nu_j = \overline{\mu_i} \qquad k \text{ fresh}$$

- Contextual rules: ......

- Housekeeping rules: ......

Co-operation/Communication: reduction semantics

- Communication:

(R-COMM)

$$\left[\!\left[R_1 \mid \sum \mu_i P_i \triangleright_{l_1} - \right]\!\right] \mid \left[\!\left[R_2 \mid \sum \nu_j Q_j \triangleright_{l_2} - \right]\!\right]$$

$$\rightarrow$$

$$[\![R_1 \mid P_i \triangleright_k -]\!] \mid [\![R_2 \mid Q_j \triangleright_k -]\!] \qquad \text{if } \nu_j = \overline{\mu_i}$$

k fresh

- Contextual rules: . . . . . .
- Housekeeping rules: . . . . . .

Example

$$[\![a.b.co \rhd_{k_1} \mathbb{0}]\!] \mid [\![\overline{b}.co \rhd_{k_2} \mathbb{0}]\!] \mid [\![\overline{a}.co.A \rhd_{k_3} B]\!]$$

$$\rightarrow [\![b.co \rhd_{k} \mathbb{0}]\!] \mid [\![\overline{b}.co \rhd_{k_2} \mathbb{0}]\!] \mid [\![co.A \rhd_{k} B]\!]$$

$$\rightarrow [\![co \rhd_{l} \mathbb{0}]\!] \mid [\![co \rhd_{l} \mathbb{0}]\!] \mid [\![co.A \rhd_{l} B]\!]$$

$\rightarrow \mathbb{0} \mid \mathbb{0} \mid A$ via distributed commit l

$\rightarrow \mathbb{0} \mid \mathbb{0} \mid B$ via distributed abort l

Example

$$[\![a.b.co \rhd_{k_1} \mathbb{0}]\!] \mid [\![\overline{b}.co \rhd_{k_2} \mathbb{0}]\!] \mid [\![\overline{a}.co.A \rhd_{k_3} B]\!]$$

$$\to [\![b.co \rhd_{k} \mathbb{0}]\!] \mid [\![\overline{b}.co \rhd_{k_2} \mathbb{0}]\!] \mid [\![co.A \rhd_{k} B]\!]$$

$$\to [\![co \rhd_{l} \mathbb{0}]\!] \mid [\![co \rhd_{l} \mathbb{0}]\!] \mid [\![co.A \rhd_{l} B]\!]$$

$\to \mathbb{0} \mid \mathbb{0} \mid A$ via distributed commit l

$\to \mathbb{0} \mid \mathbb{0} \mid B$ via distributed abort l

Example

$$[\![a.b.co \triangleright_{k_1} \mathbf{0}]\!] \mid [\![\overline{b}.co \triangleright_{k_2} \mathbf{0}]\!] \mid [\![\overline{a}.co.A \triangleright_{k_3} B]\!]$$

$$\to [\![b.co \triangleright_{k} \mathbf{0}]\!] \mid [\![\overline{b}.co \triangleright_{k_2} \mathbf{0}]\!] \mid [\![co.A \triangleright_{k} B]\!]$$

$$\to [\![co \triangleright_{l} \mathbf{0}]\!] \mid [\![co \triangleright_{l} \mathbf{0}]\!] \mid [\![co.A \triangleright_{l} B]\!]$$

$\to \mathbf{0} \mid \mathbf{0} \mid A$ via distributed commit l

$\to \mathbf{0} \mid \mathbf{0} \mid B$ via distributed abort l

TRINITY COLLEGE DUBLIN
COLÁISTE NA TRÍONÓIDE, BAILE ÁTHA CLIATH

Example

$$[\![a.b.co \triangleright_{k_1} \mathbf{0}]\!] \mid [\![\overline{b}.co \triangleright_{k_2} \mathbf{0}]\!] \mid [\![\overline{a}.co.A \triangleright_{k_3} B]\!]$$

$$\rightarrow [\![b.co \triangleright_{k} \mathbf{0}]\!] \mid [\![\overline{b}.co \triangleright_{k_2} \mathbf{0}]\!] \mid [\![co.A \triangleright_{k} B]\!]$$

$$\rightarrow [\![co \triangleright_{l} \mathbf{0}]\!] \mid [\![co \triangleright_{l} \mathbf{0}]\!] \mid [\![co.A \triangleright_{l} B]\!]$$

$\rightarrow \mathbf{0} \mid \mathbf{0} \mid A$ via distributed commit l

$\rightarrow \mathbf{0} \mid \mathbf{0} \mid B$ via distributed abort l

Example

$$[\![a.b.co \rhd_{k_1} \mathbb{0}]\!] \mid [\![\overline{b}.co \rhd_{k_2} \mathbb{0}]\!] \mid [\![\overline{a}.co.A \rhd_{k_3} B]\!]$$

$$\rightarrow [\![b.co \rhd_{k} \mathbb{0}]\!] \mid [\![\overline{b}.co \rhd_{k_2} \mathbb{0}]\!] \mid [\![co.A \rhd_{k} B]\!]$$

$$\rightarrow [\![co \rhd_{l} \mathbb{0}]\!] \mid [\![co \rhd_{l} \mathbb{0}]\!] \mid [\![co.A \rhd_{l} B]\!]$$

$\rightarrow \mathbb{0} \mid \mathbb{0} \mid A$ via distributed commit l

$\rightarrow \mathbb{0} \mid \mathbb{0} \mid B$ via distributed abort l

Environment roll-back: reduction semantics

(R-ROLLBACK)

$$\sum \mu_i P_i \mid \llbracket R_2 \mid \sum \nu_j Q_j \triangleright_l - \rrbracket$$

$$\rightarrow$$

$$\llbracket P_i \mid \mathsf{co} \triangleright_k \sum \mu_i P_i \rrbracket \mid \llbracket R_2 \mid Q_j \triangleright_k - \rrbracket \qquad \text{if } \nu_j = \overline{\mu_i} \qquad k \text{ fresh}$$

rollback as compensation

Environment roll-back: reduction semantics

(R-ROLLBACK)

$$\sum \mu_i P_i \mid \left[\!\left[R_2 \mid \sum \nu_j Q_j \triangleright_l - \right]\!\right]$$

$$\rightarrow$$

$$[\![P_i \mid \mathsf{co} \triangleright_k \sum \mu_i P_i]\!] \mid [\![R_2 \mid Q_j \triangleright_k -]\!] \qquad \text{if } \nu_j = \overline{\mu_i} \qquad k \text{ fresh}$$

rollback as compensation

Example

$$\mathtt{T1} = \mu X.[\![\overline{p_1}.\mathtt{co}.a_1 \rhd_{k_1} X]\!] \qquad \mathtt{T2} = \mu X.[\![\overline{p_2}.\mathtt{co}.a_2 \rhd_{k_2} X]\!]$$

$$\begin{array}{lll}
 & (p_1.b_1 + p_2.b_2) \mid \mathtt{T1} \mid \mathtt{T2} & \\
\to & [\![b_1 \mid \mathtt{co} \rhd_k p_1.b_1 + p_2.b_2)]\!] \mid [\![\mathtt{co}.a_1 \rhd_k \mathtt{T1}]\!] \mid \mathtt{T2} & \text{using } p_1 \\
\to & (p_1.b_1 + p_2.b_2) \mid \mathtt{T1} \mid \mathtt{T2} \quad \text{abort } k & \\
\to & [\![b_2 \mid \mathtt{co} \rhd_k p_1.b_1 + p_2.b_2)]\!] \mid \mathtt{T1} \mid [\![\mathtt{co}.a_2 \rhd_k \mathtt{T2}]\!] & \text{using } p_2 \\
\to & b_2 \mid a_2 \quad \text{commit } k &
\end{array}$$

Environment roll-back:

- Original environment $(p_1.b_1 + p_2.b_2)$ re-instated
- reduction semantics supports consistency

Example

$$\mathtt{T1} = \mu X.[\![\overline{p_1}.\mathtt{co}.a_1 \rhd_{k_1} X]\!] \qquad \mathtt{T2} = \mu X.[\![\overline{p_2}.\mathtt{co}.a_2 \rhd_{k_2} X]\!]$$

$$
\begin{array}{lll}
 & (p_1.b_1 + p_2.b_2) \mid \mathtt{T1} \mid \mathtt{T2} & \\
\rightarrow & [\![b_1 \mid \mathtt{co} \rhd_k p_1.b_1 + p_2.b_2)]\!] \mid [\![\mathtt{co}.a_1 \rhd_k \mathtt{T1}]\!] \mid \mathtt{T2} & \text{using } p_1 \\
\rightarrow & (p_1.b_1 + p_2.b_2) \mid \mathtt{T1} \mid \mathtt{T2} \quad \text{abort } k & \\
\rightarrow & [\![b_2 \mid \mathtt{co} \rhd_k p_1.b_1 + p_2.b_2)]\!] \mid \mathtt{T1} \mid [\![\mathtt{co}.a_2 \rhd_k \mathtt{T2}]\!] & \text{using } p_2 \\
\rightarrow & b_2 \mid a_2 \quad \text{commit } k &
\end{array}
$$

Environment roll-back:

- Original environment $(p_1.b_1 + p_2.b_2)$ re-instated
- reduction semantics supports consistency

informal claim

TRINITY COLLEGE DUBLIN
Coláiste na Tríonóide, Baile Átha Cliath

Behavioural equivalences

What transactions should be behavourally indistinguishable?

$$\mu X.[\![P \mid \mathrm{co} \rhd_k X]\!] \quad \stackrel{?}{\approx}_{behav} \quad P$$

$$\mu X.[\![a.b.\mathrm{co} \rhd_k X]\!] \quad \stackrel{?}{\approx}_{behav} \quad \mu X.[\![a.b.\mathrm{co} + a.c.\mathbf{0}) \rhd_k X]\!]$$

$$[\![a.\mathrm{co} \rhd_{k_1} \mathbf{0}]\!] \mid [\![b.\mathrm{co} \rhd_{k_2} \mathbf{0}]\!] \quad \stackrel{?}{\approx}_{behav} \quad \nu p.\overline{p} \mid [\![a.p.\mathrm{co}.\overline{p} \rhd_{k_1} \mathbf{0}]\!] \mid [\![b.p.\mathrm{co}.\overline{p} \rhd_{k_2} \mathbf{0}]\!]$$

Example:

The well known equivalence: *trace equivalence* $\approx_{tr}$

Behavioural equivalences

What transactions should be behavourally indistinguishable?

$$\mu X.\llbracket P \mid \mathsf{co} \rhd_k X \rrbracket \quad \overset{?}{\approx}_{behav} \quad P$$

$$\mu X.\llbracket a.b.\mathsf{co} \rhd_k X \rrbracket \quad \overset{?}{\approx}_{behav} \quad \mu X.\llbracket a.b.\mathsf{co} + a.c.\mathbf{0}) \rhd_k X \rrbracket$$

$$\llbracket a.\mathsf{co} \rhd_{k_1} \mathbf{0} \rrbracket \mid \llbracket b.\mathsf{co} \rhd_{k_2} \mathbf{0} \rrbracket \quad \overset{?}{\approx}_{behav} \quad \nu p.\overline{p} \mid \llbracket a.p.\mathsf{co}.\overline{p} \rhd_{k_1} \mathbf{0} \rrbracket \mid \llbracket b.p.\mathsf{co}.\overline{p} \rhd_{k_2} \mathbf{0} \rrbracket$$

Example:

The well known equivalence: *trace equivalence* $\approx_{\mathrm{tr}}$

CCS: Action semantics

CCS doing actions:

$P \stackrel{a}{\Rightarrow} Q$ whenever $P \mid \overline{a}.\omega \to Q \mid \omega$ $\qquad \omega$ fresh

CCS doing sequences:

$P \stackrel{s}{\Rightarrow} Q$, $s \in Act^\star$, whenever $P \mid \overline{s}.\omega \to Q \mid \omega$

CCS Trace equivalence:

$\mathtt{TR}(P) = \{\, s \in Act^\star \mid P \stackrel{s}{\Rightarrow} \,\}$

$$\boxed{P \approx_{\text{tr}} Q \text{ whenever } \mathtt{TR}(P) = \mathtt{TR}(Q)}$$

$TCCS^m$: committed Action semantics

Transactions doing committed actions:

$P \stackrel{a}{\Longmapsto} Q$ whenever $P \mid \overline{a}.\omega \rightarrow Q \mid \omega$ $\qquad \omega$ fresh

Transaction doing committed sequences:

$P \stackrel{s}{\Longmapsto} Q$, $s \in Act^\star$, whenever $P \mid \overline{s}.\omega \rightarrow Q \mid \omega$

cTrace equivalence for transactions:

$\mathtt{cTR}(P) = \{ s \in Act^\star \mid P \stackrel{s}{\Longmapsto} \}$

$$\boxed{P \approx_{\text{ctr}} Q \text{ whenever } \mathtt{cTR}(P) = \mathtt{cTR}(Q)}$$

Examples: trace equivalence

$$P = [\![a.b.\mathrm{co} \triangleright_k \mathbb{0}]\!] \quad Q = \nu p.[\![a.\mathrm{co}.p \triangleright_{k_1} \mathbb{0}]\!] \mid [\![\overline{p}.b.\mathrm{co} \triangleright_{k_2} \mathbb{0}]\!]$$

$P \not\approx_{\mathrm{ctr}} Q$:

- $\mathrm{cTR}(P) = \{\varepsilon, ab\}$ — not prefix-closed
- $\mathrm{cTR}(Q) = \{\varepsilon, a, ab\}$

$$R = \mu X.[\![a.(b.\mathrm{co} + c.\mathbb{0}) \triangleright_k X]\!] \quad S = \mu X.[\![a.b.\mathrm{co} + a.c.\mathbb{0}) \triangleright_k X]\!]$$

$R \approx_{\mathrm{ctr}} S$:

- $\mathrm{cTR}(R) = \{\varepsilon, ab\}$ — not prefix-closed
- $\mathrm{cTR}(S) = \{\varepsilon, ab\}$ — not prefix-closed

cTR supports atomicity

Examples: trace equivalence

$$P = [\![a.b.\mathrm{co} \rhd_k \mathbb{0}]\!] \quad Q = \nu p.[\![a.\mathrm{co}.p \rhd_{k_1} \mathbb{0}]\!] \mid [\![\overline{p}.b.\mathrm{co} \rhd_{k_2} \mathbb{0}]\!]$$

$P \not\approx_{\mathrm{ctr}} Q$:

- $\mathtt{cTR}(P) = \{\varepsilon, ab\}$ not prefix-closed
- $\mathtt{cTR}(Q) = \{\varepsilon, a, ab\}$

$$R = \mu X.[\![a.(b.\mathrm{co} + c.\mathbb{0}) \rhd_k X]\!] \quad S = \mu X.[\![a.b.\mathrm{co} + a.c.\mathbb{0}) \rhd_k X]\!]$$

$R \approx_{\mathrm{ctr}} S$:

- $\mathtt{cTR}(R) = \{\varepsilon, ab\}$ not prefix-closed
- $\mathtt{cTR}(S) = \{\varepsilon, ab\}$ not prefix-closed

cTR supports atomicity

Examples: trace equivalence

$$P = \llbracket a.b.\mathrm{co} \rhd_k \mathbb{0} \rrbracket \quad Q = \nu p.\llbracket a.\mathrm{co}.p \rhd_{k_1} \mathbb{0} \rrbracket \mid \llbracket \overline{p}.b.\mathrm{co} \rhd_{k_2} \mathbb{0} \rrbracket$$

$P \not\approx_{\mathrm{ctr}} Q$:

- $\mathtt{cTR}(P) = \{\varepsilon, ab\}$ not prefix-closed
- $\mathtt{cTR}(Q) = \{\varepsilon, a, ab\}$

$$R = \mu X.\llbracket a.(b.\mathrm{co} + c.\mathbb{0}) \rhd_k X \rrbracket \quad S = \mu X.\llbracket a.b.\mathrm{co} + a.c.\mathbb{0}) \rhd_k X \rrbracket$$

$R \approx_{\mathrm{ctr}} S$:

- $\mathtt{cTR}(R) = \{\varepsilon, ab\}$ not prefix-closed
- $\mathtt{cTR}(S) = \{\varepsilon, ab\}$ not prefix-closed

cTR supports atomicity

TRINITY COLLEGE DUBLIN
Coláiste na Tríonóide, Baile Átha Cliath

Examples: trace equivalence

$$P = [\![a.b.\mathrm{co} \rhd_k \mathbf{0}]\!] \quad Q = \nu p.[\![a.\mathrm{co}.p \rhd_{k_1} \mathbf{0}]\!] \mid [\![\overline{p}.b.\mathrm{co} \rhd_{k_2} \mathbf{0}]\!]$$

$P \not\approx_{\mathrm{ctr}} Q$:

- $\mathtt{cTR}(P) = \{\varepsilon, ab\}$ — not prefix-closed
- $\mathtt{cTR}(Q) = \{\varepsilon, a, ab\}$

$$R = \mu X.[\![a.(b.\mathrm{co} + c.\mathbf{0}) \rhd_k X]\!] \quad S = \mu X.[\![a.b.\mathrm{co} + a.c.\mathbf{0}) \rhd_k X]\!]$$

$R \approx_{\mathrm{ctr}} S$:

- $\mathtt{cTR}(R) = \{\varepsilon, ab\}$ — not prefix-closed
- $\mathtt{cTR}(S) = \{\varepsilon, ab\}$ — not prefix-closed

cTR supports atomicity

Justifying Trace equivalence: Safety properties

Safety: "Nothing bad will happen" [Lamport'77]

- A safety property can be formulated as a *safety test* T^{ω} which signals on fresh channel ω when it detects the bad behaviour

Definition (Passing tests)

P fails safety test T^{ω} whenever $P \mid T^{\omega} \rightarrow^* P' \mid \omega$

Example tests:

- $\mu X.(a.X + \mathrm{err}.\omega)$ can not perform err while performing any sequence of *a*s
- $T^{\omega} = \mathrm{err}.\omega \mid \overline{a}.\overline{b}$ can not perform err when *a* followed by *b* is offered.

Examples:

- $\mu X.[\![a.b.\mathrm{co} \mid \overline{\mathrm{err}} \rhd_k X]\!]$ fails safety test T^{ω}
- $\mu X.[\![a.b.\mathrm{co} + \overline{\mathrm{err}} \rhd_k X]\!]$ passes safety test T^{ω}

Justifying Trace equivalence: Safety properties

Safety: "Nothing bad will happen" [Lamport'77]

- A safety property can be formulated as a *safety test* T^{ω} which signals on fresh channel ω when it detects the bad behaviour

Definition (Passing tests)

P fails safety test T^{ω} whenever $P \mid T^{\omega} \rightarrow^{*} P' \mid \omega$

Example tests:

- $\mu X.(a.X + \mathrm{err}.\omega)$ can not perform err while performing any sequence of *a*s
- $T^{\omega} = \mathrm{err}.\omega \mid \overline{a}.\overline{b}$ can not perform err when *a* followed by *b* is offered.

Examples:

- $\mu X.[\![a.b.\mathrm{co} \mid \overline{\mathrm{err}} \triangleright_k X]\!]$ fails safety test T^{ω}
- $\mu X.[\![a.b.\mathrm{co} + \overline{\mathrm{err}} \triangleright_k X]\!]$ passes safety test T^{ω}

Justifying Trace equivalence: Safety properties

Safety: "Nothing bad will happen" [Lamport'77]

- A safety property can be formulated as a *safety test* T^{ω} which signals on fresh channel ω when it detects the bad behaviour

Definition (Passing tests)

P fails safety test T^{ω} whenever $P \mid T^{\omega} \rightarrow^{*} P' \mid \omega$

Example tests:

- $\mu X.(a.X + \text{err}.\omega)$ can not perform err while performing any sequence of *a*s
- $T^{\omega} = \text{err}.\omega \mid \overline{a}.\overline{b}$ can not perform err when *a* followed by *b* is offered.

Examples:

- $\mu X.[\![a.b.\text{co} \mid \overline{\text{err}} \triangleright_k X]\!]$ fails safety test T^{ω}
- $\mu X.[\![a.b.\text{co} + \overline{\text{err}} \triangleright_k X]\!]$ passes safety test T^{ω}

Justifying Traces

In *CCS*: well-known

$P \approx_{tr} Q$ if and only for every T^{ω},

$$P \text{ passes safety test } T^{\omega} \iff Q \text{ passes safety test } T^{\omega}$$

In $TCCS^m$: conjecture

$P \approx_{tr} Q$ if and only for every T^{ω},

$$P \text{ passes safety test } T^{\omega} \iff Q \text{ passes safety test } T^{\omega}$$

See: Concur 2010 for proof in different language of transactions.

Justifying Traces

In *CCS*: well-known

$P \approx_{tr} Q$ if and only for every T^{ω},

$$P \text{ passes safety test } T^{\omega} \iff Q \text{ passes safety test } T^{\omega}$$

In *TCCS^m*: conjecture

$P \approx_{tr} Q$ if and only for every T^{ω},

$$P \text{ passes safety test } T^{\omega} \iff Q \text{ passes safety test } T^{\omega}$$

See: Concur 2010 for proof in different language of transactions.

The problem with traces

very well-known

Trace equivalence insensitive to presence of deadlocks

In *CCS*: $a.b.\mathbb{0} \approx_{\mathrm{tr}} a.b.\mathbb{0} + a.\mathbb{0}$

In $TCCS^m$: What constitutes a deadlock?
In $TCCS^m$: What does *insensitive to deadlock* mean?

Lots of other possible behavioural equivalences: sensitive to deadlocks

- Rob J. van Glabbeek: The Linear Time-Branching Time Spectrum. CONCUR 1990: and later

CONCUR 1990: The first ever CONCUR conference

The problem with traces

very well-known

Trace equivalence insensitive to presence of deadlocks

In *CCS*: $a.b.\mathbb{0} \approx_{\mathrm{tr}} a.b.\mathbb{0} + a.\mathbb{0}$

In $TCCS^m$: What constitutes a deadlock?
In $TCCS^m$: What does *insensitive to deadlock* mean?

Lots of other possible behavioural equivalences: sensitive to deadlocks

- Rob J. van Glabbeek: The Linear Time-Branching Time Spectrum. CONCUR 1990: and later

CONCUR 1990: The first ever CONCUR conference

The problem with traces

very well-known

Trace equivalence insensitive to presence of deadlocks

In *CCS*: $a.b.\mathbb{0} \approx_{\mathrm{tr}} a.b.\mathbb{0} + a.\mathbb{0}$

In $TCCS^m$: What constitutes a deadlock?
In $TCCS^m$: What does *insensitive to deadlock* mean?

Lots of other possible behavioural equivalences: sensitive to deadlocks

- Rob J. van Glabbeek: The Linear Time-Branching Time Spectrum. CONCUR 1990: and later

CONCUR 1990: The first ever CONCUR conference

CCS Bisimulations $P \approx_{\text{bisim}} Q$

The largest relation over processes such that, if $P \approx_{\text{bisim}} Q$ then, for every $\mu \in Act_\tau$

- $P \stackrel{\mu}{\Rightarrow} P'$ implies $Q \stackrel{\mu}{\Rightarrow} Q'$ such that $P' \approx_{\text{bisim}} Q'$
- $Q \stackrel{\mu}{\Rightarrow} Q'$ implies $P \stackrel{\mu}{\Rightarrow} P'$ such that $P' \approx_{\text{bisim}} Q'$ symmetrically

Trace version:

The largest relation over processes such that, if $P \approx_{\text{bisim}} Q$ then, for every $s \in Act^*$,

- $P \stackrel{s}{\Rightarrow} P'$ implies $Q \stackrel{s}{\Rightarrow} Q'$ such that $P' \approx_{\text{bisim}} Q'$
- $Q \stackrel{s}{\Rightarrow} Q'$ implies $P \stackrel{s}{\Rightarrow} P'$ such that $P' \approx_{\text{bisim}} Q'$ symmetrically

CCS Bisimulations $P \approx_{\text{bisim}} Q$

The largest relation over processes such that, if $P \approx_{\text{bisim}} Q$ then, for every $\mu \in Act_\tau$

- $P \stackrel{\mu}{\Rightarrow} P'$ implies $Q \stackrel{\mu}{\Rightarrow} Q'$ such that $P' \approx_{\text{bisim}} Q'$
- $Q \stackrel{\mu}{\Rightarrow} Q'$ implies $P \stackrel{\mu}{\Rightarrow} P'$ such that $P' \approx_{\text{bisim}} Q'$ symmetrically

Trace version:

The largest relation over processes such that, if $P \approx_{\text{bisim}} Q$ then, for every $s \in Act^*$,

- $P \stackrel{s}{\Rightarrow} P'$ implies $Q \stackrel{s}{\Rightarrow} Q'$ such that $P' \approx_{\text{bisim}} Q'$
- $Q \stackrel{s}{\Rightarrow} Q'$ implies $P \stackrel{s}{\Rightarrow} P'$ such that $P' \approx_{\text{bisim}} Q'$ symmetrically

$TCCS^m$: Bisimulations a suggestion

The largest relation over transactions such that, if $P \approx_{\text{cbisim}} Q$ then, for $s \in Act^*$,

- $P \overset{s}{\Longmapsto} P'$ implies $Q \overset{s}{\Longmapsto} Q'$ such that $P' \approx_{\text{cbisim}} Q'$
- $Q \overset{s}{\Longmapsto} Q'$ implies $P \overset{s}{\Longmapsto} P'$ such that $P' \approx_{\text{cbisim}} Q'$

Suspicions:

- In *CCS*: $a.(b.0 + c.0) \not\approx_{\text{bisim}} a.b.0 + a.c.0$
- In $TCCS^m$:
  $[\![a.(b.\text{co} + c.\text{co}) \rhd_k 0]\!] \approx_{\text{cbisim}} [\![a.b.\text{co} + a.c.\text{co}) \rhd_k 0]\!]$

Question:

Should $[\![a.(b.\text{co} + c.\text{co}) \rhd_k 0]\!] \overset{?}{\approx}_{behav} [\![a.b.\text{co} + a.c.\text{co} \rhd_k 0]\!]$

TRINITY COLLEGE DUBLIN
Coláiste na Tríonóide, Baile Átha Cliath

$TCCS^m$: Bisimulations a suggestion

The largest relation over transactions such that, if $P \approx_{\text{cbisim}} Q$ then, for $s \in Act^*$,

- $P \stackrel{s}{\Longmapsto} P'$ implies $Q \stackrel{s}{\Longmapsto} Q'$ such that $P' \approx_{\text{cbisim}} Q'$
- $Q \stackrel{s}{\Longmapsto} Q'$ implies $P \stackrel{s}{\Longmapsto} P'$ such that $P' \approx_{\text{cbisim}} Q'$

Suspicions:

- In CCS: $a.(b.\mathbb{0} + c.\mathbb{0}) \not\approx_{\text{bisim}} a.b.\mathbb{0} + a.c.\mathbb{0}$
- In $TCCS^m$:
  $[\![a.(b.\text{co} + c.\text{co}) \rhd_k \mathbb{0}]\!] \approx_{\text{cbisim}} [\![a.b.\text{co} + a.c.\text{co}) \rhd_k \mathbb{0}]\!]$

Question:

Should $[\![a.(b.\text{co} + c.\text{co}) \rhd_k \mathbb{0}]\!] \stackrel{?}{\approx}_{behav} [\![a.b.\text{co} + a.c.\text{co} \rhd_k \mathbb{0}]\!]$

$TCCS^m$: Bisimulations a suggestion

The largest relation over transactions such that, if $P \approx_{\text{cbisim}} Q$ then, for $s \in Act^*$,

- $P \stackrel{s}{\Longmapsto} P'$ implies $Q \stackrel{s}{\Longmapsto} Q'$ such that $P' \approx_{\text{cbisim}} Q'$
- $Q \stackrel{s}{\Longmapsto} Q'$ implies $P \stackrel{s}{\Longmapsto} P'$ such that $P' \approx_{\text{cbisim}} Q'$

Suspicions:

- In CCS: $a.(b.\mathbb{0} + c.\mathbb{0}) \not\approx_{\text{bisim}} a.b.\mathbb{0} + a.c.\mathbb{0}$
- In $TCCS^m$:
  $[\![a.(b.\text{co} + c.\text{co}) \triangleright_k \mathbb{0}]\!] \approx_{\text{cbisim}} [\![a.b.\text{co} + a.c.\text{co}) \triangleright_k \mathbb{0}]\!]$

Question:

Should $[\![a.(b.\text{co} + c.\text{co}) \triangleright_k \mathbb{0}]\!] \stackrel{?}{\approx}_{behav} [\![a.b.\text{co} + a.c.\text{co} \triangleright_k \mathbb{0}]\!]$

Justifying Bisimulations

Robin Milner, Davide Sangiorgi: Barbed Bisimulation. ICALP 1992

We propose in this paper barbed bisimulation as a tool to describe bisimulation-based equivalence uniformly for any calculi possessing

(a) *a reduction relation*

(b) *a convergency predicate which simply detects the possibility of performing some observable action.*

This opens interesting perspectives for the adoption of a reduction semantics in process algebras. As a test-case we prove that strong bisimulation of CCS coincides with the congruence induced by barbed bisimulation.

Justifying Bisimulations: Reduction closure

Requirement: A reduction relation $P \to Q$ between processes.

Definition:

A relation $P \approx_{\text{behav}} Q$ is reduction-closed if, whenever $P \approx_{\text{behav}} Q$,

(i) $P \to^{*} P'$ implies $Q \to^{*} Q'$ such that $P' \approx_{\text{behav}} Q'$

(ii) $Q \to^{*} Q'$ implies $P \to^{*} P'$ such that $P' \approx_{\text{behav}} Q'$

Intuition:

P and Q must maintain the equivalent choice possibilities

Justifying Bisimulations: Contextual equivalence : (variation on M & S)

Requirements:

(i) A collection of observation relations on processes: e.g. $P \Downarrow a$

P can do the action a

(ii) a *parallel* operator on processes: e.g. $P \mid Q$

Definition: (Honda Yoshida)

$P \approx_{cxt} Q$ is the largest relation which is

- preserved by *parallel composition*
- reduction closed
- preserves observations.

Remark:

$P \approx_{cxt} Q$ is definable for many languages

Justifying Bisimulations: Contextual equivalence : (variation on M & S)

Requirements:

(i) A collection of observation relations on processes: e.g. $P \Downarrow a$

P can do the action a

(ii) a *parallel* operator on processes: e.g. $P \mid Q$

Definition: (Honda Yoshida)

$P \approx_{\mathrm{cxt}} Q$ is the largest relation which is

- preserved by *parallel composition*
- reduction closed
- preserves observations.

Remark:

$P \approx_{\mathrm{cxt}} Q$ is definable for many languages

TRINITY COLLEGE DUBLIN
Coláiste na Tríonóide, Baile Átha Cliath

CCS: Justifying Bisimulations

Theorem: In *CCS* $\boxed{P \approx_{\text{cxt}} Q \iff P \approx_{\text{bisim}} Q}$

Significance:

- Bisimulations provide a sound and complete proof method for contextual equivalence in *CCS*
- Variations on bisimulations are also sound and complete for many languages

Inconvenience:

In $TCCS^m$: $P \approx_{\text{cbisim}} Q$ does NOT imply $P \approx_{\text{cxt}} Q$ cbisimulations are unsound

Counter-example:

- $[\![a.(b.\text{co} + c.\text{co}) \triangleright_k \mathbb{0}]\!] \approx_{\text{cbisim}} [\![a.b.\text{co} + a.c.\text{co}) \triangleright_k \mathbb{0}]\!]$
- $[\![a.(b.\text{co} + c.\text{co}) \triangleright_k \mathbb{0}]\!] \not\approx_{\text{cxt}} [\![a.b.\text{co} + a.c.\text{co} \triangleright_k \mathbb{0}]\!]$

CCS: Justifying Bisimulations

Theorem: In *CCS* $\boxed{P \approx_{\text{cxt}} Q \iff P \approx_{\text{bisim}} Q}$

Significance:

- Bisimulations provide a sound and complete proof method for contextual equivalence in *CCS*
- Variations on bisimulations are also sound and complete for many languages

Inconvenience:

In $TCCS^m$: $P \approx_{\text{cbisim}} Q$ does NOT imply $P \approx_{\text{cxt}} Q$ cbisimulations are unsound

Counter-example:

- $[\![a.(b.\text{co} + c.\text{co}) \triangleright_k \mathbb{0}]\!] \approx_{\text{cbisim}} [\![a.b.\text{co} + a.c.\text{co}) \triangleright_k \mathbb{0}]\!]$
- $[\![a.(b.\text{co} + c.\text{co}) \triangleright_k \mathbb{0}]\!] \not\approx_{\text{cxt}} [\![a.b.\text{co} + a.c.\text{co} \triangleright_k \mathbb{0}]\!]$

CCS: Justifying Bisimulations

Theorem: In *CCS* $\boxed{P \approx_{\text{cxt}} Q \iff P \approx_{\text{bisim}} Q}$

Significance:

- Bisimulations provide a sound and complete proof method for contextual equivalence in *CCS*
- Variations on bisimulations are also sound and complete for many languages

Inconvenience:

In $TCCS^m$: $P \approx_{\text{cbisim}} Q$ does NOT imply $P \approx_{\text{cxt}} Q$ cbisimulations are unsound

Counter-example:

- $[\![a.(b.\text{co} + c.\text{co}) \triangleright_k 0]\!] \approx_{\text{cbisim}} [\![a.b.\text{co} + a.c.\text{co}) \triangleright_k 0]\!]$
- $[\![a.(b.\text{co} + c.\text{co}) \triangleright_k 0]\!] \not\approx_{\text{cxt}} [\![a.b.\text{co} + a.c.\text{co} \triangleright_k 0]\!]$

CCS: Justifying Bisimulations

Theorem: In *CCS* $\boxed{P \approx_{\text{cxt}} Q \iff P \approx_{\text{bisim}} Q}$

Significance:

- Bisimulations provide a sound and complete proof method for contextual equivalence in *CCS*
- Variations on bisimulations are also sound and complete for many languages

Inconvenience:

In $TCCS^m$: $P \approx_{\text{cbisim}} Q$ does NOT imply $P \approx_{\text{cxt}} Q$ cbisimulations are unsound

Counter-example:

- $[\![a.(b.\text{co} + c.\text{co}) \rhd_k \mathbb{0}]\!] \approx_{\text{cbisim}} [\![a.b.\text{co} + a.c.\text{co}) \rhd_k \mathbb{0}]\!]$
- $[\![a.(b.\text{co} + c.\text{co}) \rhd_k \mathbb{0}]\!] \not\approx_{\text{cxt}} [\![a.b.\text{co} + a.c.\text{co} \rhd_k \mathbb{0}]\!]$

The inconvenience

$$P = [\![a.(b.\mathrm{co} + c.\mathrm{co}) \rhd_k \mathbb{0}]\!] \qquad Q = [\![a.b.\mathrm{co} + a.c.\mathrm{co} \rhd_k \mathbb{0}]\!]$$

- $P \not\approx_{\mathrm{cxt}} Q$
- because $P \mid [\![\bar{a}.\mathrm{co} \rhd_k \mathbb{0}]\!] \not\approx_{\mathrm{cxt}} Q \mid [\![\bar{a}.\mathrm{co} \rhd_k \mathbb{0}]\!]$
- because
  - $P \mid [\![\bar{a}.\mathrm{co} \rhd_k \mathbb{0}]\!] \;\rightarrow\; [\![b.\mathrm{co} + c.\mathrm{co} \rhd_{k_1} \mathbb{0}]\!] \mid [\![\mathrm{co} \rhd_{k_1} \mathbb{0}]\!]$
  - $Q \mid [\![\bar{a}.\mathrm{co} \rhd_k \mathbb{0}]\!] \;\rightarrow^*\; ?$

Moral:

Internal tentative decision states matter

remember *CCS*: $a.(b.\mathbb{0} + c.\mathbb{0}) \not\approx_{\mathrm{cxt}} a.b.\mathbb{0} + a.c.\mathbb{0}$

The inconvenience

$$P = [\![a.(b.\mathrm{co} + c.\mathrm{co}) \rhd_k \mathbb{0}]\!] \qquad Q = [\![a.b.\mathrm{co} + a.c.\mathrm{co} \rhd_k \mathbb{0}]\!]$$

- $P \not\eqsim_{\mathrm{cxt}} Q$
- because $P \mid [\![\overline{a}.\mathrm{co} \rhd_k \mathbb{0}]\!] \not\eqsim_{\mathrm{cxt}} Q \mid [\![\overline{a}.\mathrm{co} \rhd_k \mathbb{0}]\!]$
- because
  - $P \mid [\![\overline{a}.\mathrm{co} \rhd_k \mathbb{0}]\!] \rightarrow [\![b.\mathrm{co} + c.\mathrm{co} \rhd_{k_1} \mathbb{0}]\!] \mid [\![\mathrm{co} \rhd_{k_1} \mathbb{0}]\!]$
  - $Q \mid [\![\overline{a}.\mathrm{co} \rhd_k \mathbb{0}]\!] \rightarrow^* ?$

Moral:

Internal tentative decision states matter

remember *CCS*: $a.(b.\mathbb{0} + c.\mathbb{0}) \not\eqsim_{\mathrm{cxt}} a.b.\mathbb{0} + a.c.\mathbb{0}$

The inconvenience

$$P = [\![a.(b.\mathsf{co} + c.\mathsf{co}) \rhd_k \mathbb{0}]\!] \qquad Q = [\![a.b.\mathsf{co} + a.c.\mathsf{co} \rhd_k \mathbb{0}]\!]$$

- $P \not\approx_{\mathsf{cxt}} Q$
- because $P \mid [\![\overline{a}.\mathsf{co} \rhd_k \mathbb{0}]\!] \not\approx_{\mathsf{cxt}} Q \mid [\![\overline{a}.\mathsf{co} \rhd_k \mathbb{0}]\!]$
- because
  - $P \mid [\![\overline{a}.\mathsf{co} \rhd_k \mathbb{0}]\!] \rightarrow [\![b.\mathsf{co} + c.\mathsf{co} \rhd_{k_1} \mathbb{0}]\!] \mid [\![\mathsf{co} \rhd_{k_1} \mathbb{0}]\!]$
  - $Q \mid [\![\overline{a}.\mathsf{co} \rhd_k \mathbb{0}]\!] \rightarrow^* ?$

Moral:

Internal tentative decision states matter

remember *CCS*: $a.(b.\mathbb{0} + c.\mathbb{0}) \not\approx_{\mathsf{cxt}} a.b.\mathbb{0} + a.c.\mathbb{0}$

The inconvenience

$P = [\![a.(b.\mathrm{co} + c.\mathrm{co}) \triangleright_k \mathbb{0}]\!]$ $\qquad Q = [\![a.b.\mathrm{co} + a.c.\mathrm{co} \triangleright_k \mathbb{0}]\!]$

- $P \not\approx_{\mathrm{cxt}} Q$
- because $P \mid [\![\overline{a}.\mathrm{co} \triangleright_k \mathbb{0}]\!] \not\approx_{\mathrm{cxt}} Q \mid [\![\overline{a}.\mathrm{co} \triangleright_k \mathbb{0}]\!]$
- because
  - $P \mid [\![\overline{a}.\mathrm{co} \triangleright_k \mathbb{0}]\!] \rightarrow [\![b.\mathrm{co} + c.\mathrm{co} \triangleright_{k_1} \mathbb{0}]\!] \mid [\![\mathrm{co} \triangleright_{k_1} \mathbb{0}]\!]$
  - $Q \mid [\![\overline{a}.\mathrm{co} \triangleright_k \mathbb{0}]\!] \rightarrow^* ?$

Moral:

Internal tentative decision states matter

remember *CCS*: $a.(b.\mathbb{0} + c.\mathbb{0}) \not\approx_{\mathrm{cxt}} a.b.\mathbb{0} + a.c.\mathbb{0}$

$TCCS^m$ Challenge

Find a notion of bisimulation which characterises contextual equivalence $\approx_{cxt}$

Obstacles:

- some tentative states are relevant:
  $[\![a.(b.\mathrm{co} + c.\mathrm{co}) \triangleright_k \mathbb{0}]\!] \not\approx_{cxt} [\![a.b.\mathrm{co} + a.c.\mathrm{co} \triangleright_k \mathbb{0}]\!]$
- some tentative states are not relevant:
  $[\![a.(b.\mathrm{co} + c.\mathbb{0}) \triangleright_k \mathbb{0}]\!] \approx_{cxt} [\![a.b.\mathrm{co} + a.c.\mathbb{0}) \triangleright_k \mathbb{0}]\!]$

History is important:

- record tentative actions
- later decide which actions were really relevant

$TCCS^m$ Challenge

Find a notion of bisimulation which characterises contextual equivalence $\approx_{\text{cxt}}$

Obstacles:

- some tentative states are relevant:
  $[\![a.(b.\text{co} + c.\text{co}) \triangleright_k \mathbb{0}]\!] \not\approx_{\text{cxt}} [\![a.b.\text{co} + a.c.\text{co} \triangleright_k \mathbb{0}]\!]$
- some tentative states are not relevant:
  $[\![a.(b.\text{co} + c.\mathbb{0}) \triangleright_k \mathbb{0}]\!] \approx_{\text{cxt}} [\![a.b.\text{co} + a.c.\mathbb{0}) \triangleright_k \mathbb{0}]\!]$

History is important:

- record tentative actions
- later decide which actions were really relevant

$TCCS^m$ Challenge

Find a notion of bisimulation which characterises contextual equivalence $\approx_{\text{cxt}}$

Obstacles:

- some tentative states are relevant:
  $[\![a.(b.\text{co} + c.\text{co}) \rhd_k \mathbb{0}]\!] \not\approx_{\text{cxt}} [\![a.b.\text{co} + a.c.\text{co} \rhd_k \mathbb{0}]\!]$
- some tentative states are not relevant:
  $[\![a.(b.\text{co} + c.\mathbb{0}) \rhd_k \mathbb{0}]\!] \approx_{\text{cxt}} [\![a.b.\text{co} + a.c.\mathbb{0}) \rhd_k \mathbb{0}]\!]$

History is important:

- record tentative actions
- later decide which actions were really relevant

History actions

- Tentative external action: $\mathcal{R} \triangleright P \xrightarrow{k(a)} \mathcal{R}', k(a) \triangleright P'$ $\quad k$ fresh
- Internal action: $\mathcal{R} \triangleright P \xrightarrow{\tau} \mathcal{R}' \triangleright P'$
  - housekeeping
  - communication
  - transaction commit/abort

$\mathcal{R}$:

- records tentative external actions taken
- records retrospectively if tentative actions become
  - permanent
  - or aborted

History actions

- Tentative external action: $\mathcal{R} \triangleright P \xrightarrow{k(a)} \mathcal{R}', k(a) \triangleright P'$ k fresh
- Internal action: $\mathcal{R} \triangleright P \xrightarrow{\tau} \mathcal{R}' \triangleright P'$
  - housekeeping
  - communication
  - transaction commit/abort

$\mathcal{R}$:

- records tentative external actions taken
- records retrospectively if tentative actions become
  - permanent
  - or aborted

History actions

- Tentative external action: $\mathcal{R} \rhd P \xrightarrow{k(a)} \mathcal{R}', k(a) \rhd P'$ k fresh
- Internal action: $\mathcal{R} \rhd P \xrightarrow{\tau} \mathcal{R}' \rhd P'$
  - housekeeping
  - communication
  - transaction commit/abort

$\mathcal{R}$:

- records tentative external actions taken
- records retrospectively if tentative actions become
  - permanent
  - or aborted

Example

$$\varepsilon \rhd [\![a.p.co \rhd_{l_1} \mathbf{0}]\!] \mid [\![b.q.co \rhd_{l_2} \mathbf{0}]\!] \mid [\![c.\overline{q}.\overline{p}.co \rhd_{l_3} \mathbf{0}]\!]$$

$$\xrightarrow{k_1(a)} \quad \text{fresh } k_1$$

$$k_1(a) \rhd [\![p.co \rhd_{k_1} \mathbf{0}]\!] \mid [\![b.q.co \rhd_{l_2} \mathbf{0}]\!] \mid [\![c.\overline{q}.\overline{p}.co \rhd_{l_3} \mathbf{0}]\!]$$

$$\xrightarrow{k_2(b)} \quad \text{fresh } k_2$$

$$k_1(a)\, k_2(b) \rhd [\![p.co \rhd_{k_1} \mathbf{0}]\!] \mid [\![q.co \rhd_{k_2} \mathbf{0}]\!] \mid [\![c.\overline{q}.\overline{p}.co \rhd_{l_3} \mathbf{0}]\!]$$

$$\xrightarrow{k_3(c)} \quad \text{fresh } k_3$$

$$k_1(a)\, k_2(b)\, k_3(c) \rhd [\![p.co \rhd_{k_1} \mathbf{0}]\!] \mid [\![q.co \rhd_{k_2} \mathbf{0}]\!] \mid [\![\overline{q}.\overline{p}.co \rhd_{k_3} \mathbf{0}]\!]$$

$$\xrightarrow{\tau} \quad \text{communication}$$

$$k_1(a)\, k_4(b)\, k_4(c) \rhd [\![p.co \rhd_{k_1} \mathbf{0}]\!] \mid [\![co \rhd_{k_4} \mathbf{0}]\!] \mid [\![\overline{p} co \rhd_{k_4} \mathbf{0}]\!]$$

$$\xrightarrow{\tau} \quad \text{communication}$$

$$k_5(a)\, k_5(b)\, k_5(c) \rhd [\![co \rhd_{k_5} \mathbf{0}]\!] \mid [\![co \rhd_{k_5} \mathbf{0}]\!] \mid [\![co \rhd_{k_5} \mathbf{0}]\!]$$

$$\xrightarrow{\tau} \quad \text{distributed commit}$$

$$k_5(co)\, k_5(co)\, k_5(co) \rhd \mathbf{0} \mid \mathbf{0} \mid \mathbf{0}$$

Example

$$\varepsilon \rhd [\![a.p.\mathrm{co} \rhd_{l_1} \mathbf{0}]\!] \mid [\![b.q.\mathrm{co} \rhd_{l_2} \mathbf{0}]\!] \mid [\![c.\overline{q}.\overline{p}.\mathrm{co} \rhd_{l_3} \mathbf{0}]\!]$$

$\xrightarrow{k_1(a)}$ fresh k_1

$$k_1(a) \rhd [\![p.\mathrm{co} \rhd_{k_1} \mathbf{0}]\!] \mid [\![b.q.\mathrm{co} \rhd_{l_2} \mathbf{0}]\!] \mid [\![c.\overline{q}.\overline{p}.\mathrm{co} \rhd_{l_3} \mathbf{0}]\!]$$

$\xrightarrow{k_2(b)}$ fresh k_2

$$k_1(a)\, k_2(b) \rhd [\![p.\mathrm{co} \rhd_{k_1} \mathbf{0}]\!] \mid [\![q.\mathrm{co} \rhd_{k_2} \mathbf{0}]\!] \mid [\![c.\overline{q}.\overline{p}.\mathrm{co} \rhd_{l_3} \mathbf{0}]\!]$$

$\xrightarrow{k_3(c)}$ fresh k_3

$$k_1(a)\, k_2(b)\, k_3(c) \rhd [\![p.\mathrm{co} \rhd_{k_1} \mathbf{0}]\!] \mid [\![q.\mathrm{co} \rhd_{k_2} \mathbf{0}]\!] \mid [\![\overline{q}.\overline{p}.\mathrm{co} \rhd_{k_3} \mathbf{0}]\!]$$

$\xrightarrow{\tau}$ communication

$$k_1(a)\, k_4(b)\, k_4(c) \rhd [\![p.\mathrm{co} \rhd_{k_1} \mathbf{0}]\!] \mid [\![\mathrm{co} \rhd_{k_4} \mathbf{0}]\!] \mid [\![\overline{p}\mathrm{co} \rhd_{k_4} \mathbf{0}]\!]$$

$\xrightarrow{\tau}$ communication

$$k_5(a)\, k_5(b)\, k_5(c) \rhd [\![\mathrm{co} \rhd_{k_5} \mathbf{0}]\!] \mid [\![\mathrm{co} \rhd_{k_5} \mathbf{0}]\!] \mid [\![\mathrm{co} \rhd_{k_5} \mathbf{0}]\!]$$

$\xrightarrow{\tau}$ distributed commit

$$k_5(\mathrm{co})\, k_5(\mathrm{co})\, k_5(\mathrm{co}) \rhd \mathbf{0} \mid \mathbf{0} \mid \mathbf{0}$$

Example

$$\varepsilon \rhd [\![a.p.co \rhd_{l_1} \mathbf{0}]\!] \mid [\![b.q.co \rhd_{l_2} \mathbf{0}]\!] \mid [\![c.\overline{q}.\overline{p}.co \rhd_{l_3} \mathbf{0}]\!]$$

$\xrightarrow{k_1(a)}$ fresh k_1

$$k_1(a) \rhd [\![p.co \rhd_{k_1} \mathbf{0}]\!] \mid [\![b.q.co \rhd_{l_2} \mathbf{0}]\!] \mid [\![c.\overline{q}.\overline{p}.co \rhd_{l_3} \mathbf{0}]\!]$$

$\xrightarrow{k_2(b)}$ fresh k_2

$$k_1(a)\, k_2(b) \rhd [\![p.co \rhd_{k_1} \mathbf{0}]\!] \mid [\![q.co \rhd_{k_2} \mathbf{0}]\!] \mid [\![c.\overline{q}.\overline{p}.co \rhd_{l_3} \mathbf{0}]\!]$$

$\xrightarrow{k_3(c)}$ fresh k_3

$$k_1(a)\, k_2(b)\, k_3(c) \rhd [\![p.co \rhd_{k_1} \mathbf{0}]\!] \mid [\![q.co \rhd_{k_2} \mathbf{0}]\!] \mid [\![\overline{q}.\overline{p}.co \rhd_{k_3} \mathbf{0}]\!]$$

$\xrightarrow{\tau}$ communication

$$k_1(a)\, k_4(b)\, k_4(c) \rhd [\![p.co \rhd_{k_1} \mathbf{0}]\!] \mid [\![co \rhd_{k_4} \mathbf{0}]\!] \mid [\![\overline{p}co \rhd_{k_4} \mathbf{0}]\!]$$

$\xrightarrow{\tau}$ communication

$$k_5(a)\, k_5(b)\, k_5(c) \rhd [\![co \rhd_{k_5} \mathbf{0}]\!] \mid [\![co \rhd_{k_5} \mathbf{0}]\!] \mid [\![co \rhd_{k_5} \mathbf{0}]\!]$$

$\xrightarrow{\tau}$ distributed commit

$$k_5(co)\, k_5(co)\, k_5(co) \rhd \mathbf{0} \mid \mathbf{0} \mid \mathbf{0}$$

Example

$$\varepsilon \rhd [\![a.p.co \rhd_{l_1} \mathbf{0}]\!] \mid [\![b.q.co \rhd_{l_2} \mathbf{0}]\!] \mid [\![c.\overline{q}.\overline{p}.co \rhd_{l_3} \mathbf{0}]\!]$$

$\xrightarrow{k_1(a)}$ fresh k_1

$$k_1(a) \rhd [\![p.co \rhd_{k_1} \mathbf{0}]\!] \mid [\![b.q.co \rhd_{l_2} \mathbf{0}]\!] \mid [\![c.\overline{q}.\overline{p}.co \rhd_{l_3} \mathbf{0}]\!]$$

$\xrightarrow{k_2(b)}$ fresh k_2

$$k_1(a)\, k_2(b) \rhd [\![p.co \rhd_{k_1} \mathbf{0}]\!] \mid [\![q.co \rhd_{k_2} \mathbf{0}]\!] \mid [\![c.\overline{q}.\overline{p}.co \rhd_{l_3} \mathbf{0}]\!]$$

$\xrightarrow{k_3(c)}$ fresh k_3

$$k_1(a)\, k_2(b)\, k_3(c) \rhd [\![p.co \rhd_{k_1} \mathbf{0}]\!] \mid [\![q.co \rhd_{k_2} \mathbf{0}]\!] \mid [\![\overline{q}.\overline{p}.co \rhd_{k_3} \mathbf{0}]\!]$$

$\xrightarrow{\tau}$ communication

$$k_1(a)\, k_4(b)\, k_4(c) \rhd [\![p.co \rhd_{k_1} \mathbf{0}]\!] \mid [\![co \rhd_{k_4} \mathbf{0}]\!] \mid [\![\overline{p}co \rhd_{k_4} \mathbf{0}]\!]$$

$\xrightarrow{\tau}$ communication

$$k_5(a)\, k_5(b)\, k_5(c) \rhd [\![co \rhd_{k_5} \mathbf{0}]\!] \mid [\![co \rhd_{k_5} \mathbf{0}]\!] \mid [\![co \rhd_{k_5} \mathbf{0}]\!]$$

$\xrightarrow{\tau}$ distributed commit

$$k_5(co)\, k_5(co)\, k_5(co) \rhd \mathbf{0} \mid \mathbf{0} \mid \mathbf{0}$$

Example

$$\varepsilon \rhd [\![a.p.co \rhd_{l_1} \mathbf{0}]\!] \mid [\![b.q.co \rhd_{l_2} \mathbf{0}]\!] \mid [\![c.\overline{q}.\overline{p}.co \rhd_{l_3} \mathbf{0}]\!]$$

$$\xrightarrow{k_1(a)} \quad \text{fresh } k_1$$

$$k_1(a) \rhd [\![p.co \rhd_{k_1} \mathbf{0}]\!] \mid [\![b.q.co \rhd_{l_2} \mathbf{0}]\!] \mid [\![c.\overline{q}.\overline{p}.co \rhd_{l_3} \mathbf{0}]\!]$$

$$\xrightarrow{k_2(b)} \quad \text{fresh } k_2$$

$$k_1(a)\, k_2(b) \rhd [\![p.co \rhd_{k_1} \mathbf{0}]\!] \mid [\![q.co \rhd_{k_2} \mathbf{0}]\!] \mid [\![c.\overline{q}.\overline{p}.co \rhd_{l_3} \mathbf{0}]\!]$$

$$\xrightarrow{k_3(c)} \quad \text{fresh } k_3$$

$$k_1(a)\, k_2(b)\, k_3(c) \rhd [\![p.co \rhd_{k_1} \mathbf{0}]\!] \mid [\![q.co \rhd_{k_2} \mathbf{0}]\!] \mid [\![\overline{q}.\overline{p}.co \rhd_{k_3} \mathbf{0}]\!]$$

$$\xrightarrow{\tau} \quad \text{communication}$$

$$k_1(a)\, k_4(b)\, k_4(c) \rhd [\![p.co \rhd_{k_1} \mathbf{0}]\!] \mid [\![co \rhd_{k_4} \mathbf{0}]\!] \mid [\![\overline{p}co \rhd_{k_4} \mathbf{0}]\!]$$

$$\xrightarrow{\tau} \quad \text{communication}$$

$$k_5(a)\, k_5(b)\, k_5(c) \rhd [\![co \rhd_{k_5} \mathbf{0}]\!] \mid [\![co \rhd_{k_5} \mathbf{0}]\!] \mid [\![co \rhd_{k_5} \mathbf{0}]\!]$$

$$\xrightarrow{\tau} \quad \text{distributed commit}$$

$$k_5(co)\, k_5(co)\, k_5(co) \rhd \mathbf{0} \mid \mathbf{0} \mid \mathbf{0}$$

Example

$$\varepsilon \rhd [\![a.p.co \rhd_{l_1} \mathbb{0}]\!] \mid [\![b.q.co \rhd_{l_2} \mathbb{0}]\!] \mid [\![c.\overline{q}.\overline{p}.co \rhd_{l_3} \mathbb{0}]\!]$$

$\xrightarrow{k_1(a)}$ fresh k_1

$$k_1(a) \rhd [\![p.co \rhd_{k_1} \mathbb{0}]\!] \mid [\![b.q.co \rhd_{l_2} \mathbb{0}]\!] \mid [\![c.\overline{q}.\overline{p}.co \rhd_{l_3} \mathbb{0}]\!]$$

$\xrightarrow{k_2(b)}$ fresh k_2

$$k_1(a)\,k_2(b) \rhd [\![p.co \rhd_{k_1} \mathbb{0}]\!] \mid [\![q.co \rhd_{k_2} \mathbb{0}]\!] \mid [\![c.\overline{q}.\overline{p}.co \rhd_{l_3} \mathbb{0}]\!]$$

$\xrightarrow{k_3(c)}$ fresh k_3

$$k_1(a)\,k_2(b)\,k_3(c) \rhd [\![p.co \rhd_{k_1} \mathbb{0}]\!] \mid [\![q.co \rhd_{k_2} \mathbb{0}]\!] \mid [\![\overline{q}.\overline{p}.co \rhd_{k_3} \mathbb{0}]\!]$$

$\xrightarrow{\tau}$ communication

$$k_1(a)\,k_4(b)\,k_4(c) \rhd [\![p.co \rhd_{k_1} \mathbb{0}]\!] \mid [\![co \rhd_{k_4} \mathbb{0}]\!] \mid [\![\overline{p}co \rhd_{k_4} \mathbb{0}]\!]$$

$\xrightarrow{\tau}$ communication

$$k_5(a)\,k_5(b)\,k_5(c) \rhd [\![co \rhd_{k_5} \mathbb{0}]\!] \mid [\![co \rhd_{k_5} \mathbb{0}]\!] \mid [\![co \rhd_{k_5} \mathbb{0}]\!]$$

$\xrightarrow{\tau}$ distributed commit

$$k_5(co)\,k_5(co)\,k_5(co) \rhd \mathbb{0} \mid \mathbb{0} \mid \mathbb{0}$$

Example

$$\varepsilon \rhd [\![a.p.co \rhd_{l_1} \mathbb{0}]\!] \mid [\![b.q.co \rhd_{l_2} \mathbb{0}]\!] \mid [\![c.\overline{q}.\overline{p}.co \rhd_{l_3} \mathbb{0}]\!]$$

$$\xrightarrow{k_1(a)} \quad \text{fresh } k_1$$

$$k_1(a) \rhd [\![p.co \rhd_{k_1} \mathbb{0}]\!] \mid [\![b.q.co \rhd_{l_2} \mathbb{0}]\!] \mid [\![c.\overline{q}.\overline{p}.co \rhd_{l_3} \mathbb{0}]\!]$$

$$\xrightarrow{k_2(b)} \quad \text{fresh } k_2$$

$$k_1(a)\, k_2(b) \rhd [\![p.co \rhd_{k_1} \mathbb{0}]\!] \mid [\![q.co \rhd_{k_2} \mathbb{0}]\!] \mid [\![c.\overline{q}.\overline{p}.co \rhd_{l_3} \mathbb{0}]\!]$$

$$\xrightarrow{k_3(c)} \quad \text{fresh } k_3$$

$$k_1(a)\, k_2(b)\, k_3(c) \rhd [\![p.co \rhd_{k_1} \mathbb{0}]\!] \mid [\![q.co \rhd_{k_2} \mathbb{0}]\!] \mid [\![\overline{q}.\overline{p}.co \rhd_{k_3} \mathbb{0}]\!]$$

$$\xrightarrow{\tau} \quad \text{communication}$$

$$k_1(a)\, k_4(b)\, k_4(c) \rhd [\![p.co \rhd_{k_1} \mathbb{0}]\!] \mid [\![co \rhd_{k_4} \mathbb{0}]\!] \mid [\![\overline{p}co \rhd_{k_4} \mathbb{0}]\!]$$

$$\xrightarrow{\tau} \quad \text{communication}$$

$$k_5(a)\, k_5(b)\, k_5(c) \rhd [\![co \rhd_{k_5} \mathbb{0}]\!] \mid [\![co \rhd_{k_5} \mathbb{0}]\!] \mid [\![co \rhd_{k_5} \mathbb{0}]\!]$$

$$\xrightarrow{\tau} \quad \text{distributed commit}$$

$$k_5(co)\, k_5(co)\, k_5(co) \rhd \mathbb{0} \mid \mathbb{0} \mid \mathbb{0}$$

TRINITY COLLEGE DUBLIN
Coláiste na Tríonóide, Baile Átha Cliath

What is recorded in $\mathcal{R}$?

$$\mathcal{R} : I \longrightarrow_{\text{finite}} \{\, k(a), k(\texttt{co}), k(\texttt{ab}) \mid k \text{ a transaction}, a \text{ an action} \,\}$$

- I: an index set

Intuition: $R \triangleright P$

$\mathcal{R}(i) = k(a)$: k is the current name (in P) of transaction used in ith external interaction

Note: Historical names are forgotten

What is recorded in $\mathcal{R}$?

$$\mathcal{R} : I \longrightarrow_{\text{finite}} \{\, k(a), k(\mathtt{co}), k(\mathtt{ab}) \mid k \text{ a transaction}, a \text{ an action} \,\}$$

- I: an index set

Intuition: $R \triangleright P$

$\mathcal{R}(i) = k(a)$: k is the current name (in P) of transaction used in ith external interaction

Note: Historical names are forgotten

History actions: inference rules

some

- External actions
- Commiting/aborting rules broadcasts
- Communication
- Contextual rules
- Housekeeping rules

History actions: inference rules

Tentative external actions:

k fresh

$$\frac{P \xrightarrow{a} P' \quad \text{in } CCS}{\mathcal{R} \triangleright [\![P \triangleright_l \; Q]\!] \xrightarrow{k(a)} \mathcal{R}\{k/l\},\, k(a) \triangleright [\![P' \triangleright_k \; Q]\!]}$$

$$\mathcal{R} \triangleright \Sigma \mu_i . P_i \xrightarrow{k(a)} \mathcal{R}, k(a) \triangleright [\![P_j \mid \mathrm{co} \triangleright_k \; \Sigma \mu_i . P_i]\!] \quad \mu_j = a$$

Intuition:

k is a fresh transaction in the environment requesting a communication on a

History actions: inference rules

Communication

$$\frac{\mathcal{R} \rhd P \xrightarrow{k(a)} \mathcal{R}\sigma, k(a) \rhd P' \qquad \mathcal{K} \rhd Q \xrightarrow{k(\bar{a})} \mathcal{K}\pi, k(\bar{a}) \rhd Q'}{\mathcal{R}, \mathcal{K} \rhd P \mid Q \xrightarrow{\tau} \mathcal{R}\sigma\pi, \mathcal{K}\pi\sigma \rhd P' \mid Q'}$$

Intuition:

- standard *CCS* communication rule
- histories need updating

History actions: Committing/Aborting

(R-CO)

$$\frac{P \xrightarrow{\text{co}} P' \quad \text{in } CCS}{\mathcal{R} \triangleright [\![P \triangleright_k Q]\!] \xrightarrow{\tau}_{\text{co}\,k} \mathcal{R} \setminus_{\text{co}} k \triangleright P}$$

Intuition:

- $\mathcal{R} \setminus_{\text{co}} k$ records that all tentative actions $k(a)$ are now permanent — transforms every $k(a)$ in $\mathcal{R}$ to $k(\text{co})$

Example:

$$k_3(a)\, k_2(b)\, k_3(c) \triangleright [\![\text{co}.P \triangleright_{k_3} 0]\!] \mid [\![b.\text{co}.R \triangleright_{k_2} 0]\!] \mid [\![\text{co}.Q \triangleright_{k_3} 0]\!]$$

$$\xrightarrow{\tau}_{\text{co}\,k}$$

$$k_3(\text{co})\, k_2(b)\, k_3(\text{co}) \triangleright P \mid [\![b.\text{co}.R \triangleright_{k_2} 0]\!] \mid Q$$

History actions: Committing/Aborting

(R-CO)

$$\frac{P \xrightarrow{\mathsf{co}} P' \quad \text{in } CCS}{\mathcal{R} \rhd [\![P \rhd_k Q]\!] \xrightarrow{\tau}_{\mathsf{co}\,k} \mathcal{R} \setminus_{\mathsf{co}} k \rhd P}$$

Intuition:

- $\mathcal{R} \setminus_{\mathsf{co}} k$ records that all tentative actions $k(a)$ are now permanent transforms every $k(a)$ in $\mathcal{R}$ to $k(\mathsf{co})$

Example:

$$k_3(a)\, k_2(b)\, k_3(c) \rhd [\![\mathsf{co}.P \rhd_{k_3} \mathbb{0}]\!] \mid [\![b.\mathsf{co}.R \rhd_{k_2} \mathbb{0}]\!] \mid [\![\mathsf{co}.Q \rhd_{k_3} \mathbb{0}]\!]$$

$$\xrightarrow{\tau}_{\mathsf{co}\,k}$$

$$k_3(\mathsf{co})\, k_2(b)\, k_3(\mathsf{co}) \rhd P \mid [\![b.\mathsf{co}.R \rhd_{k_2} \mathbb{0}]\!] \mid Q$$

History actions: Committing/Aborting

(R-CO)

$$\frac{\ldots}{\ldots}$$

(R-BCAST)

$$\frac{\mathcal{R} \triangleright P \xrightarrow{\tau}_{\mathsf{cok}} \mathcal{R}' \triangleright P' \qquad \mathcal{K} \triangleright Q \xrightarrow{\tau}_{\mathsf{cok}} \mathcal{K}' \triangleright Q'}{\mathcal{R}, \mathcal{K} \triangleright P \mid Q \xrightarrow{\tau}_{\mathsf{cok}} \mathcal{R}', \mathcal{K}' \triangleright P \mid Q}$$

(R-IGNORE)

$$\frac{\mathcal{R} \triangleright P \xrightarrow{\tau}_{\mathsf{cok}} \mathcal{R}' \triangleright P'}{\mathcal{R}, \mathcal{K} \triangleright P \mid Q \xrightarrow{\tau}_{\mathsf{cok}} \mathcal{R}', \mathcal{K} \triangleright P \mid Q} \quad k \text{ fresh to } \mathcal{K} \triangleright Q$$

Intuition:

- All components of the distributed transaction k must commit $\xrightarrow{\mathsf{co}}$ simultaneously

History bisimulations $\mathcal{R} \triangleright P \approx_{\text{bisim}} \mathcal{K} \triangleright Q$

The largest relation over configurations such that, if $\mathcal{R} \triangleright P \approx_{\text{bisim}} \mathcal{K} \triangleright Q$ then, for every μ

- $\mathcal{R} \triangleright P \stackrel{\mu}{\Rightarrow} \mathcal{R}' \triangleright P'$ implies $\mathcal{K} \triangleright Q \stackrel{\mu}{\Rightarrow} \mathcal{K}' \triangleright Q'$ such that $\mathcal{R}' \triangleright Q' \approx_{\text{bisim}} \mathcal{K}' \triangleright Q'$
- symmetrically $\mathcal{K} \triangleright Q \stackrel{\mu}{\Rightarrow} \mathcal{K}' \triangleright Q'$ implies ......
- Records $\mathcal{R}$, $\mathcal{K}$ are consistent: they agree on committed actions.

Intuition:
Permanent actions must match

Consistent: for every index $i \in I$, $\mathcal{R}(i) = k(\text{co})$ iff $\mathcal{K}(i) = k'(\text{co})$

History bisimulations $\mathcal{R} \triangleright P \approx_{\text{bisim}} \mathcal{K} \triangleright Q$

The largest relation over configurations such that, if $\mathcal{R} \triangleright P \approx_{\text{bisim}} \mathcal{K} \triangleright Q$ then, for every μ

- $\mathcal{R} \triangleright P \stackrel{\mu}{\Rightarrow} \mathcal{R}' \triangleright P'$ implies $\mathcal{K} \triangleright Q \stackrel{\mu}{\Rightarrow} \mathcal{K}' \triangleright Q'$ such that $\mathcal{R}' \triangleright Q' \approx_{\text{bisim}} \mathcal{K}' \triangleright Q'$
- symmetrically $\mathcal{K} \triangleright Q \stackrel{\mu}{\Rightarrow} \mathcal{K}' \triangleright Q'$ implies ......
- Records $\mathcal{R}$, $\mathcal{K}$ are consistent: they agree on committed actions.

Intuition:

Permanent actions must match

Consistent: for every index $i \in I$, $\mathcal{R}(i) = k(\text{co})$ iff $\mathcal{K}(i) = k'(\text{co})$

A problem

$$[\![a.b.\mathrm{co} \rhd_k \mathbb{0}]\!] \approx_{\mathrm{cxt}} [\![a.b.\mathrm{co} + a.c.\mathbb{0} \rhd_k \mathbb{0}]\!]$$ difficult to prove

But $P = [\![a.b.\mathrm{co} \rhd_k \mathbb{0}]\!] \not\approx_{\mathrm{bisim}} [\![a.b.\mathrm{co} + a.c.\mathbb{0} \rhd_k \mathbb{0}]\!] = Q$

- $\epsilon \rhd Q \xrightarrow{k_1(a)} k_1(a) \rhd [\![c.\mathbb{0} \rhd_{k_1} \mathbb{0}]\!] \xrightarrow{k_2(c)} k_2(a)k_2(c) \rhd [\![\mathbb{0} \rhd_{k_2} \mathbb{0}]\!]$
- $\epsilon \rhd P \xrightarrow{k_1(a)} k_1(a) \rhd [\![b.\mathrm{co} \rhd_{k_1} \mathbb{0}]\!] \overset{k_2(c)}{\Longrightarrow}$??

A solution:

Allow free degenerate tentative actions: $\mathcal{R} \rhd S \xrightarrow{k(x)} \mathcal{R}, k(\mathrm{ab}) \rhd S$

Because:

- $\epsilon \rhd P \xrightarrow{k_1(a)} \xrightarrow{k_2(c)} k_1(a)k_2(\mathrm{ab}) \rhd [\![b.\mathrm{co} \rhd_{k_1} \mathbb{0}]\!]$
  $\xrightarrow{\tau}_{\mathrm{ab}} k_1(a)k_2(\mathrm{ab}) \rhd \mathbb{0}$
- and $\boxed{k_2(a)k_2(b) \rhd [\![\mathbb{0} \rhd_{k_2} \mathbb{0}]\!] \approx_{\mathrm{bisim}} k_1(a)k_2(\mathrm{ab}) \rhd \mathbb{0}}$

A problem

$$[\![a.b.\mathrm{co} \rhd_k \mathbf{0}]\!] \approx_{\mathrm{cxt}} [\![a.b.\mathrm{co} + a.c.\mathbf{0} \rhd_k \mathbf{0}]\!] \qquad \text{difficult to prove}$$

But $P = [\![a.b.\mathrm{co} \rhd_k \mathbf{0}]\!] \not\approx_{\mathrm{bisim}} [\![a.b.\mathrm{co} + a.c.\mathbf{0} \rhd_k \mathbf{0}]\!] = Q$

- $\epsilon \rhd Q \xrightarrow{k_1(a)} k_1(a) \rhd [\![c.\mathbf{0} \rhd_{k_1} \mathbf{0}]\!] \xrightarrow{k_2(c)} k_2(a)k_2(c) \rhd [\![\mathbf{0} \rhd_{k_2} \mathbf{0}]\!]$
- $\epsilon \rhd P \xrightarrow{k_1(a)} k_1(a) \rhd [\![b.\mathrm{co} \rhd_{k_1} \mathbf{0}]\!] \xRightarrow{k_2(c)}$??

A solution:

Allow free degenerate tentative actions: $\mathcal{R} \rhd S \xrightarrow{k(x)} \mathcal{R}, k(\mathrm{ab}) \rhd S$

Because:

- $\epsilon \rhd P \xrightarrow{k_1(a)} \xrightarrow{k_2(c)} k_1(a)k_2(\mathrm{ab}) \rhd [\![b.\mathrm{co} \rhd_{k_1} \mathbf{0}]\!]$
  $\xrightarrow{\tau}_{\mathrm{ab}} k_1(a)k_2(\mathrm{ab}) \rhd \mathbf{0}$
- and $k_2(a)k_2(b) \rhd [\![\mathbf{0} \rhd_{k_2} \mathbf{0}]\!] \approx_{\mathrm{bisim}} k_1(a)k_2(\mathrm{ab}) \rhd \mathbf{0}$

A problem

$$[\![a.b.\mathrm{co} \rhd_k \mathbf{0}]\!] \approx_{\mathrm{cxt}} [\![a.b.\mathrm{co} + a.c.\mathbf{0} \rhd_k \mathbf{0}]\!]$$ difficult to prove

But $P = [\![a.b.\mathrm{co} \rhd_k \mathbf{0}]\!] \not\approx_{\mathrm{bisim}} [\![a.b.\mathrm{co} + a.c.\mathbf{0} \rhd_k \mathbf{0}]\!] = Q$

- $\epsilon \rhd Q \xrightarrow{k_1(a)} k_1(a) \rhd [\![c.\mathbf{0} \rhd_{k_1} \mathbf{0}]\!] \xrightarrow{k_2(c)} k_2(a)k_2(c) \rhd [\![\mathbf{0} \rhd_{k_2} \mathbf{0}]\!]$
- $\epsilon \rhd P \xrightarrow{k_1(a)} k_1(a) \rhd [\![b.\mathrm{co} \rhd_{k_1} \mathbf{0}]\!] \xRightarrow{k_2(c)}$??

A solution:

Allow free degenerate tentative actions: $\mathcal{R} \rhd S \xrightarrow{k(x)} \mathcal{R}, k(\mathtt{ab}) \rhd S$

Because:

- $\epsilon \rhd P \xrightarrow{k_1(a)}\xrightarrow{k_2(c)} k_1(a)k_2(\mathrm{ab}) \rhd [\![b.\mathrm{co} \rhd_{k_1} \mathbf{0}]\!]$
  $\xrightarrow{\tau}_{\mathrm{ab}} k_1(a)k_2(\mathrm{ab}) \rhd \mathbf{0}$
- and $k_2(a)k_2(b) \rhd [\![\mathbf{0} \rhd_{k_2} \mathbf{0}]\!] \approx_{\mathrm{bisim}} k_1(a)k_2(\mathrm{ab}) \rhd \mathbf{0}$

A problem

$$[\![a.b.\mathtt{co} \rhd_k \mathbb{0}]\!] \approx_{\mathrm{cxt}} [\![a.b.\mathtt{co} + a.c.\mathbb{0} \rhd_k \mathbb{0}]\!]$$ difficult to prove

But $P = [\![a.b.\mathtt{co} \rhd_k \mathbb{0}]\!] \not\approx_{\mathrm{bisim}} [\![a.b.\mathtt{co} + a.c.\mathbb{0} \rhd_k \mathbb{0}]\!] = Q$

- $\epsilon \rhd Q \xrightarrow{k_1(a)} k_1(a) \rhd [\![c.\mathbb{0} \rhd_{k_1} \mathbb{0}]\!] \xrightarrow{k_2(c)} k_2(a)k_2(c) \rhd [\![\mathbb{0} \rhd_{k_2} \mathbb{0}]\!]$
- $\epsilon \rhd P \xrightarrow{k_1(a)} k_1(a) \rhd [\![b.\mathtt{co} \rhd_{k_1} \mathbb{0}]\!] \xRightarrow{k_2(c)}$??

A solution:

Allow free degenerate tentative actions: $\mathcal{R} \rhd S \xrightarrow{k(x)} \mathcal{R}, k(\mathrm{ab}) \rhd S$

Because:

- $\epsilon \rhd P \xrightarrow{k_1(a)}\xrightarrow{k_2(c)} k_1(a)k_2(\mathrm{ab}) \rhd [\![b.\mathtt{co} \rhd_{k_1} \mathbb{0}]\!]$
  $\xrightarrow{\tau}_{\mathrm{ab}} k_1(a)k_2(\mathrm{ab}) \rhd \mathbb{0}$
- and $\boxed{k_2(a)k_2(b) \rhd [\![\mathbb{0} \rhd_{k_2} \mathbb{0}]\!] \approx_{\mathrm{bisim}} k_1(a)k_2(\mathrm{ab}) \rhd \mathbb{0}}$

Justifying bisimulations

In $TCCS^m$

$$P \approx_{\text{bisim}} Q \qquad \text{iff} \qquad P \approx_{\text{cxt}} Q$$

History bisimulations give a sound and complete proof method for contextual equivalence of transactions

Fossacs 2014

Inequivalent systems

In *CCS*:

- $P = a.c.(d.\mathbb{0} + e.\mathbb{0}) + a.c.e.\mathbb{0} \;\not\approx_{\text{cxt}}\; a.(c.d.\mathbb{0} + c.e.\mathbb{0}) = Q$
- because $P \not\approx_{\text{bisim}} Q$
- because P and Q satisfy different behavioural properties

$P \models \langle a \rangle\, [c](\langle d \rangle\, \text{tr} \wedge \langle e \rangle\, \text{tr})$ while $Q \not\models \langle a \rangle\, [c](\langle d \rangle\, \text{tr} \wedge \langle e \rangle\, \text{tr})$

In *TCCS^m*:

$$P = [\![a.\text{co} \rhd_{k_1} \mathbb{0}]\!] \mid [\![b.\text{co} \rhd_{k_2} \mathbb{0}]\!]$$
$$Q = \nu p.\overline{p} \mid [\![a.p.\text{co}.\overline{p} \rhd_{k_1} \mathbb{0}]\!] \mid [\![b.p.\text{co}.\overline{p} \rhd_{k_2} \mathbb{0}]\!]$$

- $P \not\approx_{\text{cxt}} Q$
- because $P \not\approx_{\text{bisim}} Q$
- because ???

Inequivalent systems

In *CCS*:

- $P = a.c.(d.\mathbb{0} + e.\mathbb{0}) + a.c.e.\mathbb{0} \;\not\approx_{\text{cxt}}\; a.(c.d.\mathbb{0} + c.e.\mathbb{0}) = Q$
- because $P \not\approx_{\text{bisim}} Q$
- because P and Q satisfy different behavioural properties

$$P \models \langle a \rangle\, [c](\langle d \rangle\, \text{tr} \wedge \langle e \rangle\, \text{tr}) \text{ while } Q \not\models \langle a \rangle\, [c](\langle d \rangle\, \text{tr} \wedge \langle e \rangle\, \text{tr})$$

In *TCCS*^{m}:

$$P = [\![a.\text{co} \triangleright_{k_1} \mathbb{0}]\!] \mid [\![b.\text{co} \triangleright_{k_2} \mathbb{0}]\!]$$
$$Q = \nu p.\overline{p} \mid [\![a.p.\text{co}.\overline{p} \triangleright_{k_1} \mathbb{0}]\!] \mid [\![b.p.\text{co}.\overline{p} \triangleright_{k_2} \mathbb{0}]\!]$$

- $P \;\not\approx_{\text{cxt}}\; Q$
- because $P \not\approx_{\text{bisim}} Q$
- because ???

Inequivalent systems

In *CCS*:

- $P = a.c.(d.\mathbb{0} + e.\mathbb{0}) + a.c.e.\mathbb{0} \;\not\approx_{\text{cxt}}\; a.(c.d.\mathbb{0} + c.e.\mathbb{0}) = Q$
- because $P \not\approx_{\text{bisim}} Q$
- because P and Q satisfy different behavioural properties

$$P \models \langle a \rangle [c](\langle d \rangle \text{tr} \wedge \langle e \rangle \text{tr}) \text{ while } Q \not\models \langle a \rangle [c](\langle d \rangle \text{tr} \wedge \langle e \rangle \text{tr})$$

In *TCCS*^{m}:

$$P = [\![a.\text{co} \rhd_{k_1} \mathbb{0}]\!] \mid [\![b.\text{co} \rhd_{k_2} \mathbb{0}]\!]$$
$$Q = \nu p.\overline{p} \mid [\![a.p.\text{co}.\overline{p} \rhd_{k_1} \mathbb{0}]\!] \mid [\![b.p.\text{co}.\overline{p} \rhd_{k_2} \mathbb{0}]\!]$$

- $P \;\not\approx_{\text{cxt}}\; Q$
- because $P \not\approx_{\text{bisim}} Q$
- because ???

In *CCS*: property logic HML

Properties: $\phi \quad ::= \quad \langle\mu\rangle\,\phi \quad | \quad \neg\phi \quad | \quad \wedge_{\{i\in I\}}\,\phi_i$

Satisfaction:

- $P \models \langle\mu\rangle\,\phi$ if $P \stackrel{\mu}{\Rightarrow} Q$, where $Q \models \phi$
- $P \models \wedge_{\{i\in I\}}\phi_i$ if ......

Well-known result:

$P \not\approx_{\text{bisim}} Q$ iff $P \models \phi,\ Q \not\models \phi$ for some property $\phi \in$ HML

Intuition:

ϕ is a reason for the different behaviour between P and Q

In $TCCS^m$: Why are P, Q different ?

$$P = [\![a.b.\mathsf{co} \triangleright_k \mathbf{0}]\!] \qquad Q = \nu p.[\![a.\mathsf{co}.p \triangleright_{k_1} \mathbf{0}]\!] \mid [\![\overline{p}.b.\mathsf{co} \triangleright_{k_2} \mathbf{0}]\!]$$

Intuition:

- P can perform tentative actions a, b in same transaction, which can subsequently become permanent
- Q can only tentatively perform a, b in independent transactions

Intuition unsupported by current action semantics:

$$\varepsilon \triangleright P \xrightarrow{k_1(a)} k_1(a) \triangleright [\![b.\mathsf{co} \triangleright_{k_1} \mathbf{0}]\!]$$
$$\xrightarrow{k_2(b)} k_2(a)k_2(b) \triangleright [\![b.\mathsf{co} \triangleright_{k_2} \mathbf{0}]\!]$$

In $TCCS^m$: Why are P, Q different ?

$$P \;=\; [\![a.b.\mathrm{co} \rhd_k \; \mathbb{0}]\!] \qquad Q = \nu p.[\![a.\mathrm{co}.p \rhd_{k_1} \; \mathbb{0}]\!] \mid [\![\overline{p}.b.\mathrm{co} \rhd_{k_2} \; \mathbb{0}]\!]$$

Intuition:

- P can perform tentative actions a, b in same transaction, which can subsequently become permanent
- Q can only tentatively perform a, b in independent transactions

Intuition unsupported by current action semantics:

$$\varepsilon \rhd P \xrightarrow{k_1(a)} k_1(a) \rhd [\![b.\mathrm{co} \rhd_{k_1} \; \mathbb{0}]\!]$$
$$\xrightarrow{k_2(b)} k_2(a)k_2(b) \rhd [\![b.\mathrm{co} \rhd_{k_2} \; \mathbb{0}]\!]$$

In $TCCS^m$: Why are P, Q different ?

$$P = [\![a.b.\mathrm{co} \triangleright_k \mathbf{0}]\!] \qquad Q = \nu p.[\![a.\mathrm{co}.p \triangleright_{k_1} \mathbf{0}]\!] \mid [\![\overline{p}.b.\mathrm{co} \triangleright_{k_2} \mathbf{0}]\!]$$

Intuition:

- P can perform tentative actions a, b in same transaction, which can subsequently become permanent
- Q can only tentatively perform a, b in independent transactions

Intuition unsupported by current action semantics:

$$\varepsilon \triangleright P \xrightarrow{k_1(a)} k_1(a) \triangleright [\![b.\mathrm{co} \triangleright_{k_1} \mathbf{0}]\!]$$
$$\xrightarrow{k_2(b)} k_2(a)k_2(b) \triangleright [\![b.\mathrm{co} \triangleright_{k_2} \mathbf{0}]\!]$$

History is important

Recall $\mathcal{R} \rhd P$

- $\mathcal{R} : I \longrightarrow \{\, k(a), k(\texttt{co}), k(\texttt{ab}) \mid k \text{ a transaction name} \,\}$
- $\mathcal{R}(i) = k(a)$: k is the current name in P of ith interaction

New Configurations: remember historic actions

$H; \mathcal{R} \rhd P$ where

- H equivalence relation over names
  - $H \models k_1 \sim k_2$ means k_1, k_2 are the same transactions
- $\mathcal{R}(i)$ is the historic name used in ith interaction

Example:

$$\varepsilon \rhd P \xrightarrow{k_1(a)} \{k_1\} : k_1(a) \rhd [\![b.\texttt{co} \rhd_{k_1} \mathbb{0}]\!]$$

$$\xrightarrow{k_2(b)} \{k_1, k_2\}; k_1(a) k_2(b) \rhd [\![\texttt{co} \rhd_{k_2} \mathbb{0}]\!]$$

History is important

Recall $\mathcal{R} \rhd P$

- $\mathcal{R} : I \longrightarrow \{\, k(a), k(\mathtt{co}), k(\mathtt{ab}) \mid k$ a transaction name $\}$
- $\mathcal{R}(i) = k(a)$: k is the current name in P of ith interaction

New Configurations: remember historic actions

$H; \mathcal{R} \rhd P$ where

- H equivalence relation over names
  - $H \models k_1 \sim k_2$ means k_1, k_2 are the same transactions
- $\mathcal{R}(i)$ is the historic name used in ith interaction

Example:

$$\varepsilon \rhd P \xrightarrow{k_1(a)} \{k_1\} : k_1(a) \rhd [\![b.\mathtt{co} \rhd_{k_1} \mathbf{0}]\!]$$

$$\xrightarrow{k_2(b)} \{k_1, k_2\}; k_1(a)k_2(b) \rhd [\![\mathtt{co} \rhd_{k_2} \mathbf{0}]\!]$$

TRINITY COLLEGE DUBLIN
Coláiste na Tríonóide, Baile Átha Cliath

History is important

Recall $\mathcal{R} \triangleright P$

- $\mathcal{R} : I \longrightarrow \{\, k(a), k(\texttt{co}), k(\texttt{ab}) \mid k \text{ a transaction name } \,\}$
- $\mathcal{R}(i) = k(a)$: k is the current name in P of ith interaction

New Configurations: remember historic actions

$H; \mathcal{R} \triangleright P$ where

- H equivalence relation over names
  - $H \models k_1 \sim k_2$ means k_1, k_2 are the same transactions
- $\mathcal{R}(i)$ is the historic name used in ith interaction

Example:

$$\varepsilon \triangleright P \xrightarrow{k_1(a)} \{k_1\} : k_1(a) \triangleright [\![b.\texttt{co} \triangleright_{k_1} \mathbf{0}]\!]$$

$$\xrightarrow{k_2(b)} \{k_1, k_2\}; k_1(a)k_2(b) \triangleright [\![\texttt{co} \triangleright_{k_2} \mathbf{0}]\!]$$

TRINITY COLLEGE DUBLIN
Coláiste na Tríonóide, Baile Átha Cliath

In $TCCS^m$: property logic trHML

Properties: $\phi \quad ::= \quad \langle k(a)\rangle\,\phi \mid \langle\tau\rangle\,\phi \mid \mathtt{Isco}(k) \mid \neg\phi \mid \bigwedge_{\{i\in I\}}\phi_i$

Satisfaction:

- $H;\mathcal{R} \rhd P \models \langle k(a)\rangle\,\phi$ if $H;\mathcal{R} \rhd P \xRightarrow{k'(a)} H';\mathcal{R}' \rhd Q$, where
  - $H';\mathcal{R}' \rhd Q \models \phi$
  - $E \models k \sim k'$
- $H;\mathcal{R} \rhd P \models \mathtt{Isco}(k)$ if $\exists i,\ \mathcal{R}(i) = k'(\mathrm{co}),\ H \models k \sim k'$

Example:

$P \quad = \quad [\![a.b.\mathrm{co} \rhd_{k_1} \ \mathbb{0}]\!] \qquad Q = \nu p.[\![a.p.\mathrm{co} \rhd_{k_1} \ \mathbb{0}]\!] \mid [\![b.\overline{p}.\mathrm{co} \rhd_{k_2} \ \mathbb{0}]\!]$

$$\epsilon \rhd P \quad \models \quad \langle k(a)\rangle\,\langle k(b)\rangle\,\mathtt{Isco}(k)$$

$$\epsilon \rhd Q \quad \not\models \quad \ldots$$

TRINITY COLLEGE DUBLIN
COLÁISTE NA TRÍONÓIDE, BAILE ÁTHA CLIATH

In $TCCS^m$: property logic trHML

Properties: $\phi \quad ::= \quad \langle k(a)\rangle\, \phi \mid \langle \tau \rangle\, \phi \mid \mathtt{Isco}(k) \mid \neg\phi \mid \bigwedge_{\{i\in I\}} \phi_i$

Satisfaction:

- $H;\mathcal{R} \rhd P \models \langle k(a)\rangle\, \phi$ if $H;\mathcal{R} \rhd P \xRightarrow{k'(a)} H';\mathcal{R}' \rhd Q$, where
  - $H';\mathcal{R}' \rhd Q \models \phi$
  - $E \models k \sim k'$
- $H;\mathcal{R} \rhd P \models \mathtt{Isco}(k)$ if $\exists i,\ \mathcal{R}(i) = k'(\mathrm{co}),\ H \models k \sim k'$

Example:

$P \quad = \quad [\![a.b.\mathrm{co} \rhd_{k_1} \mathbb{0}]\!] \qquad Q = \nu p.[\![a.p.\mathrm{co} \rhd_{k_1} \mathbb{0}]\!] \mid [\![b.\overline{p}.\mathrm{co} \rhd_{k_2} \mathbb{0}]\!]$

$$\epsilon \rhd P \quad \models \quad \langle k(a)\rangle\, \langle k(b)\rangle\, \mathtt{Isco}(k)$$

$$\epsilon \rhd Q \quad \not\models \quad \ldots$$

In $TCCS^m$: property logic trHML

Conjecture:

$P \not\approx_{\text{bisim}} Q$ iff $P \models \phi$, $Q \not\models \phi$ for some property $\phi \in$ trHML

Example:

$$\begin{aligned} P &= [\![a.\mathtt{co} \triangleright_{k_1} \mathbb{0}]\!] \mid [\![b.\mathtt{co} \triangleright_{k_2} \mathbb{0}]\!] \\ Q &= \nu p.\overline{p} \mid [\![a.p.\mathtt{co}.\overline{p} \triangleright_{k_1} \mathbb{0}]\!] \mid [\![b.p.\mathtt{co}.\overline{p} \triangleright_{k_2} \mathbb{0}]\!] \end{aligned}$$

$$\begin{aligned} P &\models ????? \\ Q &\not\models ???? \end{aligned}$$

$$\begin{aligned} P &\models \langle k(a)\rangle \langle k(b)\rangle \mathtt{Isco}(k) \\ Q &\not\models \langle k(a)\rangle \langle k(b)\rangle \mathtt{Isco}(k) \end{aligned}$$

In $TCCS^m$: property logic trHML

Conjecture:

$P \not\approx_{\text{bisim}} Q$ iff $P \models \phi$, $Q \not\models \phi$ for some property $\phi \in$ trHML

Example:

$$P = [\![a.\mathtt{co} \triangleright_{k_1} \mathbf{0}]\!] \mid [\![b.\mathtt{co} \triangleright_{k_2} \mathbf{0}]\!]$$
$$Q = \nu p.\overline{p} \mid [\![a.p.\mathtt{co}.\overline{p} \triangleright_{k_1} \mathbf{0}]\!] \mid [\![b.p.\mathtt{co}.\overline{p} \triangleright_{k_2} \mathbf{0}]\!]$$

$$P \models ?????$$
$$Q \not\models ????$$

$$P \models \langle k(a)\rangle \langle k(b)\rangle \, \mathtt{Isco}(k)$$
$$Q \not\models \langle k(a)\rangle \langle k(b)\rangle \, \mathtt{Isco}(k)$$

In $TCCS^m$: property logic trHML

Conjecture:

$P \not\approx_{\text{bisim}} Q$ iff $P \models \phi$, $Q \not\models \phi$ for some property $\phi \in$ trHML

Example:

$$P = [\![a.\mathtt{co} \triangleright_{k_1} \mathbf{0}]\!] \mid [\![b.\mathtt{co} \triangleright_{k_2} \mathbf{0}]\!]$$
$$Q = \nu p.\overline{p} \mid [\![a.p.\mathtt{co}.\overline{p} \triangleright_{k_1} \mathbf{0}]\!] \mid [\![b.p.\mathtt{co}.\overline{p} \triangleright_{k_2} \mathbf{0}]\!]$$

$$P \models ?????$$
$$Q \not\models ????$$

$$P \models \langle k(a) \rangle \langle k(b) \rangle \mathtt{Isco}(k)$$
$$Q \not\models \langle k(a) \rangle \langle k(b) \rangle \mathtt{Isco}(k)$$

In $TCCS^m$: property logic trHML

Conjecture:

$P \not\approx_{\text{bisim}} Q$ iff $P \models \phi$, $Q \not\models \phi$ for some property $\phi \in$ trHML

Example:

$$
\begin{aligned}
P &= [\![a.\mathrm{co} \rhd_{k_1} \mathbf{0}]\!] \mid [\![b.\mathrm{co} \rhd_{k_2} \mathbf{0}]\!] \\
Q &= \nu p.\overline{p} \mid [\![a.p.\mathrm{co}.\overline{p} \rhd_{k_1} \mathbf{0}]\!] \mid [\![b.p.\mathrm{co}.\overline{p} \rhd_{k_2} \mathbf{0}]\!]
\end{aligned}
$$

$$
\begin{aligned}
P &\models ?????\\
Q &\not\models ????
\end{aligned}
$$

$$
\begin{aligned}
P &\models \langle k(a)\rangle \langle k(b)\rangle \mathtt{Isco}(k)\\
Q &\not\models \langle k(a)\rangle \langle k(b)\rangle \mathtt{Isco}(k)
\end{aligned}
$$

Some work done. More to do.

- Language design and implementation
- Behavioural semantics
  - Decision procedures for equivalence
    upcoming PhD thesis: Carlo Spaccasassi
  - More expressive transaction constructs.
    eg. nested transactions
- Variations
  - Reversible programming languages
  - Web services: long running transactions with compensations
- . . . .

TRINITY COLLEGE DUBLIN
Coláiste na Tríonóide, Baile Átha Cliath

The end

THANKS

Joint work with Vasileois Koutavas, Carlo Spaccasassi, Edsko de Vries