Behavioural model elaboration using MTS

Dario Fischbein
Department of Computing - Imperial College

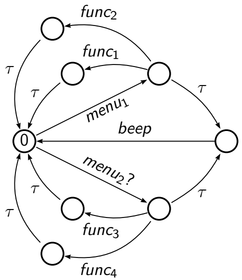
Sebastian Uchitel
Department of Computing - Imperial College,
Universidad de Buenos Aires and CONICET

"Copenhagen"

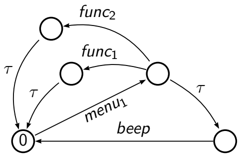
Introduction

- Conformance between MTS and LTS
- Refinement and Semantics Revisited
- Elaboration of Models via Merge
- The Modal Transition System Analyser (MTSA)

Strong Semantics (Larsen et al - 1988)



$\preceq$

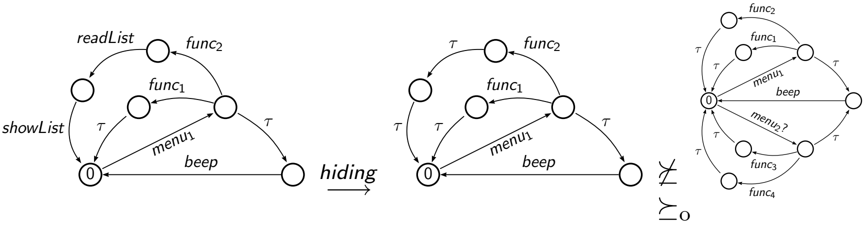


N is a refinement of M if:

- N preserves all of the required behaviour of M
- N preserves all of the proscribed behaviour of M

What happens if we need to elaborate out model with a lower level of abstraction?

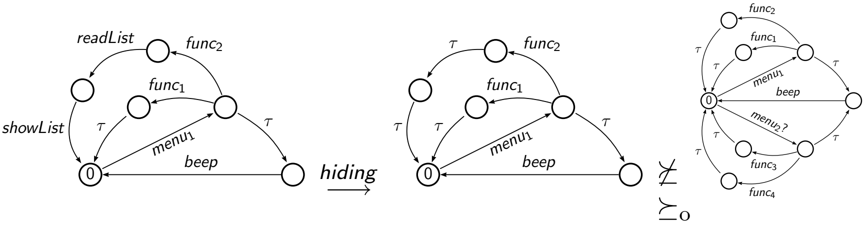
- The alphabet is expanded



- Strong semantics does not take τ transitions as internal or unobservable ones. $\Rightarrow$ an observational semantics is needed.
- Weak Semantics (Larsen et al - 1989) may be the solution...

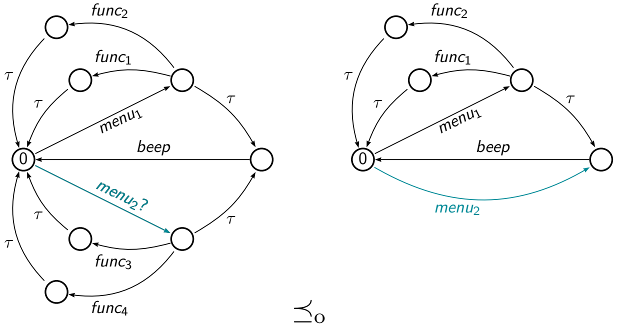
What happens if we need to elaborate out model with a lower level of abstraction?

- The alphabet is expanded

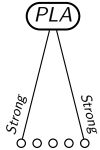


- Strong semantics does not take τ transitions as internal or unobservable ones. $\Rightarrow$ an observational semantics is needed.
- Weak Semantics (Larsen et al - 1989) may be the solution...

Unexpected Behaviour of Weak Refinement



- The users are not able to select functionalities of *menu$_n$* after having chosen it.
- This example breaks the intuition of what behaviour conformance should preserve.



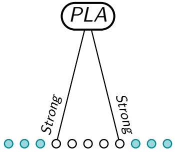
Summary of Semantics

- Strong: preserves the branching structure, but does not distinguish unobservable actions.
- Weak: allows products that contradict the intuition the modeller may have of conformance.

Objective

- To define a new semantics that captures the pros of strong and weak semantics. i.e. an observational semantics that preserves the branching structure.

Summary of Semantics

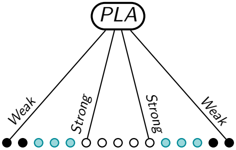


- Strong: preserves the branching structure, but does not distinguish unobservable actions.
- Weak: allows products that contradict the intuition the modeller may have of conformance.

Objective

- To define a new semantics that captures the pros of strong and weak semantics. i.e. an observational semantics that preserves the branching structure.

Summary of Semantics

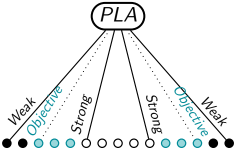


- Strong: preserves the branching structure, but does not distinguish unobservable actions.
- Weak: allows products that contradict the intuition the modeller may have of conformance.

Objective

- To define a new semantics that captures the pros of strong and weak semantics. i.e. an observational semantics that preserves the branching structure.

Summary of Semantics



- Strong: preserves the branching structure, but does not distinguish unobservable actions.
- Weak: allows products that contradict the intuition the modeller may have of conformance.

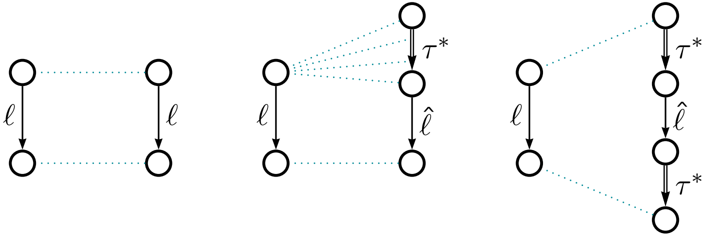
Objective

- To define a new semantics that captures the pros of strong and weak semantics. i.e. an observational semantics that preserves the branching structure.

Branching Semantics

Intuitive Idea

One model is allowed to simulate the other using τ transitions, but checking that every intermediate state the model goes through does not add nor proscribe behaviour compare to the initial state of the other model.



Definition

Branching Implementation Relation

Let R be a binary relation between MTS and LTS, R is a branching implementation relation iff for all pairs (M, I) in R and all events ℓ the following holds:

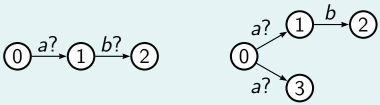
1. $(M \xrightarrow{\ell}_{\mathrm{r}} M') \implies (\exists\ I_0, \ldots, I_n, I' \cdot I_0 = I \ \wedge\ I_i \xrightarrow{\tau} I_{i+1}\ \forall 0 \leq i < n\ \wedge$
   $I_n \xrightarrow{\hat{\ell}} I' \ \wedge\ (M', I') \in R \ \wedge\ (M, I_i) \in R\ \forall 0 \leq i \leq n)$
2. $(I \xrightarrow{\ell} I') \implies (\exists M_0, \ldots, M_n, M' \cdot M_0 = M \ \wedge\ M_i \xrightarrow{\tau}_{\mathrm{p}} M_{i+1}\ \forall 0 \leq i < n\ \wedge$
   $M_n \xrightarrow{\hat{\ell}}_{\mathrm{p}} M' \ \wedge\ (M', I') \in R \ \wedge\ (M_i, I) \in R\ \forall 0 \leq i \leq n)$

- $M \preceq_{\mathrm{b}} N \equiv M \preceq_{\mathrm{O}} N$ if M or N do not have tau transitions.

Refinement relation as definition of semantics

Current Semantics are based on an operational definition of refinement - Refinement relation

Problem - Refinement relation is not complete



Semantics redefined ?

Should we redefine the semantics in terms of implementations?

Leaving refinement relations as approximate operations for checking refinement

Make “the problem” explicit

It cannot be used to check refinement, but it can be used to prove properties

Merge definition

$$0 \xrightarrow{a?} 1 \xrightarrow{b} 2 \qquad 0 \xrightarrow{a} 1 \xrightarrow{b?} 2$$

$$0 \xrightarrow{a} 1 \xrightarrow{b} 2$$

Merge $\equiv$ Least Common Refinement

A modal transition system P is the least common refinement (LCR) of modal transition systems M and N if P is a common refinement of M and N, and for any common refinement Q of M and N, $P \preceq Q$.

Consistency

Consistency

Two MTSs M and N are consistent if there exists an LTS I such that I is a common implementation of M and N.

Strong Consistency Relation

A *strong consistency relation* is a binary relation $C \subseteq \delta \times \delta$, such that the following conditions hold for all $(M, N) \in C$:

1. $(\forall \ell, M')(M \xrightarrow{\ell}_{\mathrm{r}} M' \implies (\exists N')(N \xrightarrow{\ell}_{\mathrm{p}} N' \wedge (M', N') \in C))$
2. $(\forall \ell, N')(N \xrightarrow{\ell}_{\mathrm{r}} N' \implies (\exists M')(M \xrightarrow{\ell}_{\mathrm{p}} M' \wedge (M', N') \in C))$

Consistency

Strong Consistency Relation Characterizes Consistency

Two MTSs M and N are consistent if and only if there exists a strong consistency relation C_{MN} such that (M, N) is contained in C_{MN}.

Consistency - Proof sketch

$\Leftarrow$)

Let CI be a LTS defined by
$CI = (C_{MN}, Act, \Delta_{CI}, (M_0, N_0))$ where Δ_{CI} is the smallest relation that satisfies the following rules, assuming that $\{(M, N), (M', N') \subseteq C_{MN}\}$.

$$\text{RP}\ \frac{M \xrightarrow{\ell}_{\mathrm{r}} M',\ N \xrightarrow{\ell}_{\mathrm{p}} N'}{(M,N) \xrightarrow{\ell} (M',N')} \qquad \text{PR}\ \frac{M \xrightarrow{\ell}_{\mathrm{p}} M',\ N \xrightarrow{\ell}_{\mathrm{r}} N'}{(M,N) \xrightarrow{\ell} (M',N')}$$

It is easy to prove that $M \preceq CI$ using that
$R = \{(M, (M, N)) \mid (M, N) \in C_{MN}\}$ is an implementation relation between M and CI.

Consistency - Proof sketch

$\Rightarrow$)

Since M and N are consistent we can take an LTS CI such that $M \preceq CI$ and $N \preceq CI$. By definition of strong semantics there exist R_M and R_N implementation relations between M and CI, and between N and CI respectively.
Let C_{MN} be a relation defined by $C_{MN} = R_M \circ R_N^{-1}$. It can easily be proven that C_{MN} is a strong consistency relation between M and N.

Conjunction

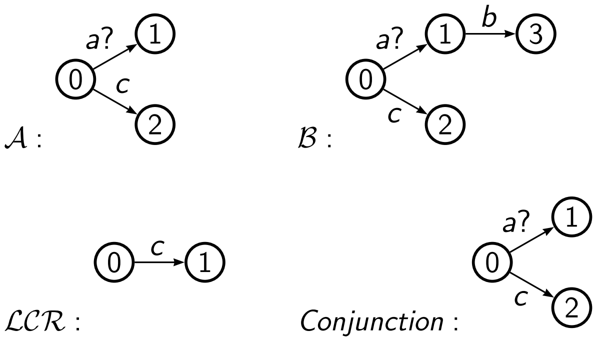
Conjunction [Larsen et al, 1995]

Let M and N be *MTSs*, the conjunction of M and N is defined as $M \wedge N = (S_M \times S_N, L, \Delta^r_{M\wedge N}, \Delta^p_{M\wedge N}, (m_0, n_0))$, where $\Delta^r_{M\wedge N}, \Delta^p_{M\wedge N}$ are the smallest relations which satisfy the following rules:

$$\text{RP}\ \frac{M \xrightarrow{\ell}_{\mathrm{r}} M',\ N \xrightarrow{\ell}_{\mathrm{p}} N'}{(M,N) \xrightarrow{\ell}_{\mathrm{r}} (M',N')} \qquad \text{PR}\ \frac{M \xrightarrow{\ell}_{\mathrm{p}} M',\ N \xrightarrow{\ell}_{\mathrm{r}} N'}{(M,N) \xrightarrow{\ell}_{\mathrm{r}} (M',N')}$$

$$\text{PP}\ \frac{M \xrightarrow{\ell}_{\mathrm{p}} M',\ N \xrightarrow{\ell}_{\mathrm{p}} N'}{(M,N) \xrightarrow{\ell}_{\mathrm{p}} (M',N')}$$

Conjunction



This problem occurs when two models are not independent but they are consistent.

The $+_{cr}$operator

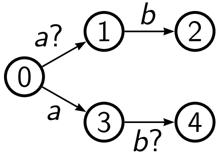
The $+_{cr}$operator* [Uchitel et al '04, Brunet et al]

Let M and N be $MTSs$ and let C_{MN} be the largest strong consistency relation between them. The $+_{cr}$operator between M and N is defined as
$M +_{cr} N = (C_{MN}, L, \Delta^r_{M+_{cr}N}, \Delta^p_{M+_{cr}N}, (m_0, n_0))$, where $\Delta^r_{M+_{cr}N}, \Delta^p_{M+_{cr}N}$ are the smallest relations which satisfy rules RP, PR, PP of Conjunction:

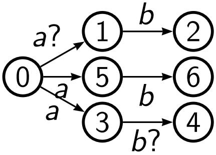
* restricted to models with the same alphabet and no unobservable actions under strong semantics

The $+_{cr}$operator

$\mathcal{H}$:



$\mathcal{H} +_{cr} \mathcal{H}$:



Clearly the merge of a model with itself should result in the same model (i.e. merge is idempotent).

$+_{cr}$ does not deal correctly with nondeterminism when there is a mix of required and maybe transitions. $+_{cr}$ will apply rules RP and PR, taking a conservative decision, which guarantee to produce a CR but might fail to produce the LCR.

A New Merge Algorithm

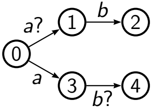
- Iteratively abstracts the result of $M +_{cr} N$ by replacing required transitions with maybe transitions.
- Guarantees that the resulting MTS after each iteration continues to be a refinement.
- Decision based on anlysing all outgoing required transitions from a given state on a given label.

Cover Set

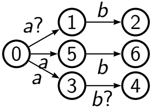
Cover Set

Intuitively a cover set describes a set of outgoing required transitions from a given state and on a given label such that if we only keep these as required the model continues to be a common refinement of M and N.

$\mathcal{H}$:

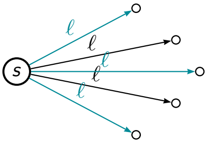


$\mathcal{H} +_{cr} \mathcal{H}$:



$\{5\}$, $\{3\}$ and $\{3,5\}$ (these sets come from considering $\{0 \xrightarrow{a} 5\}$, $\{0 \xrightarrow{a} 3\}$, and $\{0 \xrightarrow{a} 3, 0 \xrightarrow{a} 5\}$).

Abstraction operation

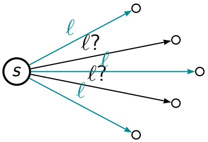


Abstraction operation

replaces any required transitions from s on ℓ that is not in the cover set with a maybe transition.

- It is straightforward to show that the abstraction operation effectively produces an abstraction. However, it is also the case that it produces a common refinement of original models.

Abstraction operation



Abstraction operation

replaces any required transitions from s on ℓ that is not in the cover set with a maybe transition.

- It is straightforward to show that the abstraction operation effectively produces an abstraction. However, it is also the case that it produces a common refinement of original models.

Base Merge algorithm

1. $M \leftarrow A +_{cr} B$, $isLCR \leftarrow true$
2. For each $(x, y) \in S_M$ and each $\ell \in Act$ do
   2.1 Get most abstract minimal cover set of (x, y) on ℓ.
   2.2 If not unique, choose any and
   $isLCR \leftarrow false$.
   2.3 $M \leftarrow \mathcal{A}(M, \zeta_{(x,y),\ell})$
3. Return (M,isLCR)

Merge algorithm

- Abstraction Operation 2 - handles the case where there are not unique most abstract cover set.
- Observational
  - Observational $+_{cr}$
  - Observational Cover Set
- Guarantees LCR construction ? (current work)

The Modal Transition System Analyser (MTSA)

- Prototype tool aimed at supporting the elaboration and verification of behaviour models for reactive systems

Demo

Conclusions

- Analysis of adequacy of the existing semantics for MTS to support modelling and analysis of software.
- Formal definition of a novel conformance relation that fulfils the desired characteristics.
- Should we "redefine" MTS semantics in terms of implementations, leaving the refinement operation as an approximation of refinement?
- An improved merge algorithm.
- A software tool aimed at supporting the elaboration and verication of behaviour models for reactive systems

Questions

?

Refinements

Strong Refinement Relation(Larsen et al - 1988)

Let R be a binary relation over the universe of MTS, R is a strong refinement relation iff for all pairs (M, N) in R and all events ℓ the following holds:

1. $(M \xrightarrow{\ell}_{\mathrm{r}} M') \Longrightarrow (\exists N' \cdot N \xrightarrow{\ell}_{\mathrm{r}} N' \ \wedge \ (M', N') \in R)$
2. $(N \xrightarrow{\ell}_{\mathrm{p}} N') \Longrightarrow (\exists M' \cdot M \xrightarrow{\ell}_{\mathrm{p}} M' \ \wedge \ (M', N') \in R)$

Weak Semantics

Weak Refinement Relation (Larsen et al - 1989)

Let R be a binary relation over the universe of MTS, R is a weak refinement relation iff for all pairs (M, N) in R and all events ℓ the following holds:

1. $(M \xrightarrow{\ell}_{\mathrm{r}} M') \implies (\exists N' \cdot N \stackrel{\hat{\ell}}{\Longrightarrow}_{\mathrm{r}} N' \ \wedge \ (M', N') \in R)$
2. $(N \xrightarrow{\ell}_{\mathrm{p}} N') \implies (\exists M' \cdot M \stackrel{\hat{\ell}}{\Longrightarrow}_{\mathrm{p}} M' \ \wedge \ (M', N') \in R)$

Notation: $P \stackrel{\ell}{\Longrightarrow} P' \equiv P(\xrightarrow{\tau})^* \xrightarrow{\ell} (\xrightarrow{\tau})^* P'$.

Branching Semantics

Branching Implementation Relation

Let R be a binary relation between MTS and LTS, R is a branching implementation relation iff for all pairs (M, I) in R and all events ℓ the following holds:

1. $(M \xrightarrow{\ell}_{\mathrm{r}} M') \implies (\exists\ I_0, \ldots, I_n, I' \cdot I_0 = I\ \wedge$
$I_i \xrightarrow{\tau} I_{i+1}\ \forall\, 0 \leq i < n\ \wedge$
$I_n \xrightarrow{\hat{\ell}} I'\ \wedge\ (M', I') \in R\ \wedge$
$(M, I_i) \in R\ \forall\, 0 \leq i \leq n)$
2. $(I \xrightarrow{\ell} I') \implies (\exists\, M_0, \ldots, M_n, M' \cdot M_0 = M\ \wedge$
$M_i \xrightarrow{\tau}_{\mathrm{p}} M_{i+1}\ \forall\, 0 \leq i < n\ \wedge$
$M_n \xrightarrow{\hat{\ell}}_{\mathrm{p}} M'\ \wedge\ (M', I') \in R\ \wedge$

Independence

Independence [Larsen et al, 1995]

An *indepence relation* R is a binary relation on δ such that if $(S, T) \in R$ then:

1. $(\forall \ell, S')(S \xrightarrow{\ell}_{\mathrm{r}} S' \implies (\exists! T')(T \xrightarrow{\ell}_{\mathrm{p}} T' \wedge (S', T') \in R))$
2. $(\forall \ell, T')(T \xrightarrow{\ell}_{\mathrm{r}} T' \implies (\exists! S')(S \xrightarrow{\ell}_{\mathrm{p}} S' \wedge (S', T') \in R))$
3. $(\forall \ell, S', T')(S \xrightarrow{\ell}_{\mathrm{p}} S' \wedge T \xrightarrow{\ell}_{\mathrm{p}} T') \implies (S', T') \in R$

Cover Set

Cover Set

Let A, B, C be *MTSs*, R_{AC}, R_{BC} be refinement relations between A and C, and B and C respectively. Given $C_i \in S_C$ and $\ell \in Act$ we define a cover set over C_i on ℓ as a set $\zeta_{C_i,\ell}$ of states of C for which the following holds:

1. $\zeta_{C_i,\ell} \subseteq \Delta^r_C(C_i, \ell)$
2. $\Delta^r_A(R^{-1}_{AC}(C_i), \ell) \subseteq R^{-1}_{AC}(\zeta_{C_i,\ell})$
3. $\Delta^r_B(R^{-1}_{BC}(C_i), \ell) \subseteq R^{-1}_{BC}(\zeta_{C_i,\ell})$

Notation: $\Delta_r(S, \ell) = \{\ t\ | s \xrightarrow{\ell}_{\mathrm{r}} t\ \wedge\ s \in S\}$

Thank you!!!