Theoretical Foundations of the UML

Lecture 4: Properties of Message Sequence Graphs

Joost-Pieter Katoen

Lehrstuhl für Informatik 2
Software Modeling and Verification Group

http://moves.rwth-aachen.de/i2/uml09100/

29. Oktober 2012

Outline

Overview

1. Message sequence graphs (MSGs)
2. Expressiveness and races
3. Intersection of MSGs
4. Non-local choice
5. A non-decomposable MSC

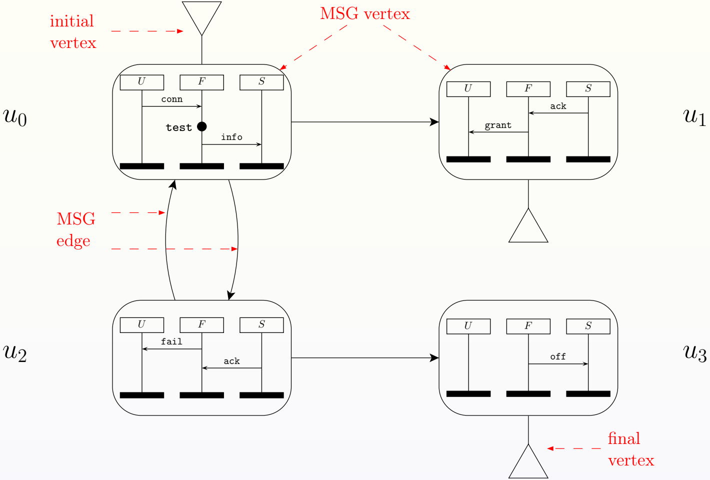
Message sequence graphs

Let $\mathbb{M}$ be the set of MSCs (up to isomorphism, i.e., event identities).

A Message Sequence Graph (MSG) G is a tuple $G = (V, \rightarrow, v_0, F, \lambda)$ with:

- $(V, \rightarrow)$ is a digraph with finite set V of vertices and $\rightarrow \subseteq V \times V$ a set of edges
- $v_0 \in V$ is the starting (or: initial) vertex
- $F \subseteq V$ is a set of final vertices
- $\lambda : V \rightarrow \mathbb{M}$ associates to each vertex $v \in V$, an MSC $\lambda(v)$

Message sequence graphs



$$u_0\ u_2\ u_0\ u_1 = \lambda(u_0) \bullet \lambda(u_2) \bullet \lambda(u_0) \bullet \lambda(u_1)$$

Concatenation of MSCs (1)

Let $M_i = (\mathcal{P}_i, E_i, \mathcal{C}_i, l_i, m_i, <_i) \qquad i \in \{1, 2\}$
be two MSCs with $E_1 \cap E_2 = \varnothing$

The concatenation of M_1 and M_2 is the MSC
$M_1 \bullet M_2 = (\mathcal{P}, E, \mathcal{C}, l, m, <)$ with:

$$\mathcal{P} = \mathcal{P}_1 \cup \mathcal{P}_2 \qquad E = E_1 \cup E_2 \qquad \mathcal{C} = \mathcal{C}_1 \cup \mathcal{C}_2$$

$$(\text{with } E_? = E_{1,?} \cup E_{2,?} \text{ etc.})$$

$$l(e) = \begin{cases} l_1(e) & \text{if} \quad e \in E_1 \\ l_2(e) & \text{if} \quad e \in E_2 \end{cases} \qquad m(e) = \begin{cases} m_1(e) & \text{if} \quad e \in E_1 \\ m_2(e) & \text{if} \quad e \in E_2 \end{cases}$$

$$< \, = \Big(<_1 \cup <_2 \cup \, \{ (e, e') \mid \exists p \in \mathcal{P}.\, e \in E_1 \cap E_p \, , \; e' \in E_2 \cap E_p \} \Big)^*$$

MSC language of an MSG

Let $G = (V, \rightarrow, v_0, F, \lambda)$ be an MSG.

Definition

Path $\pi = u_0 \; \ldots \; u_n$ is accepting if: $u_0 = v_0$ and $u_n \in F$.

Definition

The MSC of a path $\pi = u_0 \; \ldots \; u_n$ is:

$$M(\pi) \; = \; \underbrace{\lambda(u_0)}_{\text{MSC of } u_0} \bullet \underbrace{\lambda(u_1)}_{\text{MSC of } u_1} \bullet \ldots \bullet \underbrace{\lambda(u_n)}_{\text{MSC of } u_n} \; = \; \prod_{i=0}^{n} \lambda(u_i)$$

Definition

The (MSC) language of MSG G is defined by:

$$L(G) = \{ M(\pi) \mid \pi \text{ is an accepting path of } G \}.$$

Overview

1. Message sequence graphs (MSGs)
2. Expressiveness and races
3. Intersection of MSGs
4. Non-local choice
5. A non-decomposable MSC

Expressiveness and races

Expressiveness

The state space of an MSG is context-sensitive.

Proof: a proof sketch has been provided in the previous lecture.

Races

The decision problem *"does an MSG have a race"* is undecidable.

Proof: reduction from Post's Correspondence Problem (PCP).

Overview

RWTH AACHEN UNIVERSITY

Intersection of MSGs

Theorem: undecidability of empty intersection

The decision problem:

for MSGs G_1 and G_2, do we have $L(G_1) \cap L(G_2) = \varnothing$?

is undecidable.

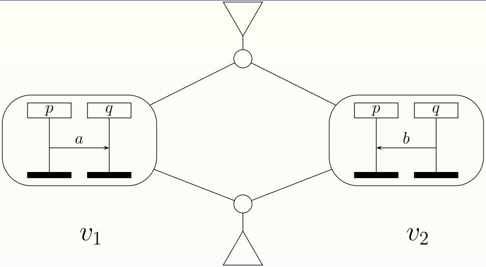
Proof: Reduction from Post's Correspondence Problem (PCP)

... black board ...

Overview

Non-local choice

G:



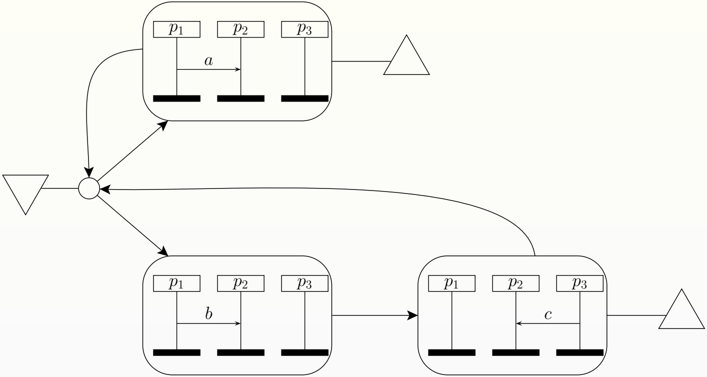
Inconsistency if process p behaves according to vertex v_1 and process q behaves according to vertex v_2

$\Longrightarrow$ possible distributed realization may yield a deadlock

Problem:

Subsequent behavior is determined by distinct processes. When several processes independently decide to initiate behavior, they might start executing different successor MSCs (= vertices). This is called a non-local choice.

A (hidden) non local-choice MSG



Problem:

Inconsistency if p_1 decides to send a and p_3 decides to send c.
Which branch to take in the initial vertex?

Definition (Minimal event)

Let $(E, \preceq)$ be a poset. Event $e \in E$ is a minimal event wrt. $\preceq$ if $\neg(\exists e' \neq e.\ e' \preceq e)$.

Intuition: there is no event that has to happen before e happens.
Or: the occurrence of e does not depend on any other event.

Definition (Partial order of a path)

For path $\pi = v_1 \ldots v_n$ in MSG G, let $<_{M(\pi)}$ be the partial order of the MSC $M(\pi) = \lambda(v_1) \bullet \ldots \bullet \lambda(v_n)$.

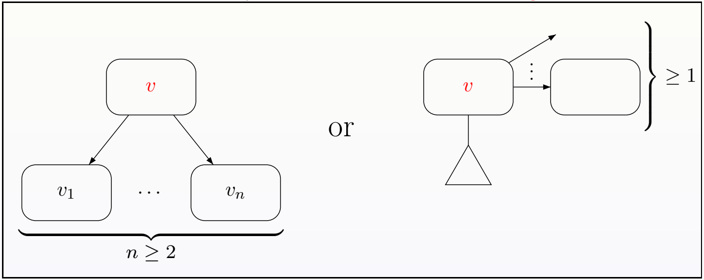
For path π let $\min(\pi)$ be the set of minimal events along π wrt. $<_{M(\pi)}$.

RWTH AACHEN UNIVERSITY

Branching vertices

A branching vertex either has at least two successors, or is an initial vertex with at least one successor.

Pictorially, vertex v is branching if:



Local choice property

Definition (Local choice)

Let MSG $G = (V, \rightarrow, v_0, F, \lambda)$. MSG G is called local choice if for every branching vertex $v \in V$ it holds:

$$\exists \text{process } p. \ (\forall \pi \in \text{Paths}(v). \ |\min(\pi)| = 1 \ \wedge \ \min(\pi) \subseteq E_p)$$

Intuition:

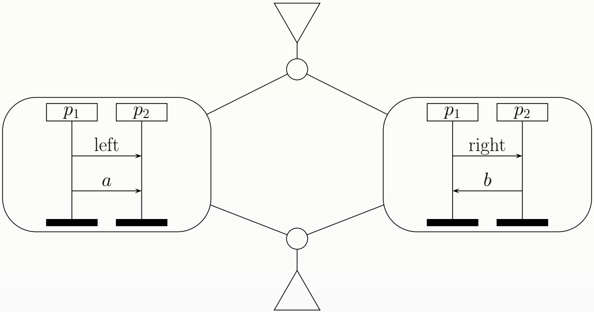
Along every path from a branching vertex in the MSG, there is a single process that initiates behavior. This process decides how to proceed. In a (distributed) implementation, it can inform the other processes how to proceed.

Local choice or not?

Checking whether MSG G is local choice can be done with a worst-case time complexity which is polynomial in the size of G. (Exercise.)

Local choice

G:

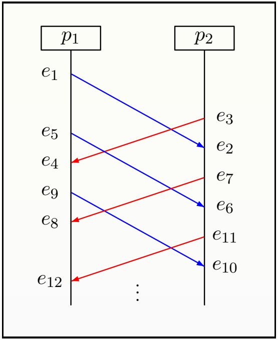


How to resolve a non-local choice?

Amend your MSG and add control messages (cf. above example)

Overview

An MSC that cannot be decomposed [Yannakakis 1999]



This MSC cannot be decomposed as

$$M_1 \bullet M_2 \bullet \ldots \bullet M_n \quad \text{for } n > 1$$

This can be seen as follows:

- e_1 and $e_2 = m(e_1)$ must both belong to M_1
- $e_3 < e_2$ and $e_1 < e_4$ thus
  $e_3, e_4 \notin M_j$, for $j < 1$ or $j > 1$
  $\Longrightarrow e_3, e_4$ must belong to M_1
- by similar reasoning: $e_5, e_6 \in M_1$ etc.

Problem:

Compulsory matching between send and receive in same MSG vertex (i.e., send e and receive $m(e)$ must belong to the same MSC).

Compositional MSCs [Gunter, Muscholl, Peled 2001]

Solution: drop restriction that e and $m(e)$ belong to the same MSC (= allow for incomplete message transfer)

Definition (Compositional MSC)

$M = (\mathcal{P}, E, \mathcal{C}, l, m, <)$ is a compositional MSC (CMSC, for short) where $\mathcal{P}, E, \mathcal{C}$ and l are defined as before, and

- $m : E_! \to E_?$ is a partial, injective function such that (as before):

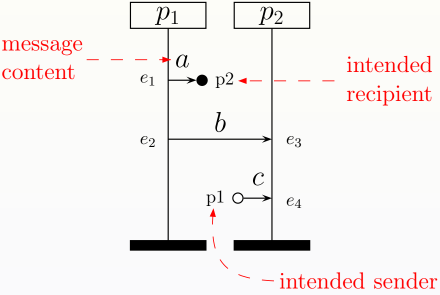
$$m(e) = e' \wedge l(e) = !(p, q, a) \implies l(e') = ?(q, p, a)$$

- $$< \; = \Big(\bigcup_{p \in \mathcal{P}} <_p \quad \cup \quad \underbrace{\{(e, m(e)) \mid e \in \underbrace{dom(m)}_{\text{domain of } m}}_{\text{"}m(e)\text{ is defined"}} \}\Big)^*$$

Note:

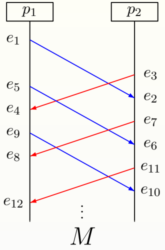
An MSC is a CMSC where m is total and bijective.

CMSC example



$$m(e_2) = e_3$$
$$e_1 \notin dom(m)$$
$$e_4 \notin rng(m)$$

Yannakakis' example as compositional MSG



This MSC cannot be modeled for $n > 1$ by:

$$M = M_1 \bullet M_2 \bullet \ldots \bullet M_n \quad \text{with} \quad M_i \in \mathbb{M}$$

But it can be modeled as compositional MSG: