Theoretical Foundations of the UML

Lecture 12: Regular MSCs

Joost-Pieter Katoen

Lehrstuhl für Informatik 2
Software Modeling and Verification Group

moves.rwth-aachen.de/teaching/ss-16/theoretical-foundations-of-the-uml/

15. Juni 2016

Outline

1. Realisability and safe realisability
2. Regular MSCs
3. Regularity and realisability for MSCs
4. Regularity and realisability for MSGs
   - Communication closedness

Overview

1. Realisability and safe realisability
2. Regular MSCs
3. Regularity and realisability for MSCs
4. Regularity and realisability for MSGs
   - Communication closedness

Realisabiliy and safe realisability

Definition (Realisability)

1. MSC M is realisable whenever $\{M\} = \mathcal{L}(\mathcal{A})$ for some CFM $\mathcal{A}$.
2. A finite set $\{M_1, \ldots, M_n\}$ of MSCs is realisable whenever $\{M_1, \ldots, M_n\} = \mathcal{L}(\mathcal{A})$ for some CFM $\mathcal{A}$.
3. MSG G is realisable whenever $\mathcal{L}(G) = \mathcal{L}(\mathcal{A})$ for some CFM $\mathcal{A}$.

Realisabiliy and safe realisability

Definition (Realisability)

1. MSC M is realisable whenever $\{M\} = \mathcal{L}(\mathcal{A})$ for some CFM $\mathcal{A}$.
2. A finite set $\{M_1, \ldots, M_n\}$ of MSCs is realisable whenever $\{ M_1, \ldots, M_n\} = \mathcal{L}(\mathcal{A})$ for some CFM $\mathcal{A}$.
3. MSG G is realisable whenever $\mathcal{L}(G) = \mathcal{L}(\mathcal{A})$ for some CFM $\mathcal{A}$.

Definition (Safe realisability)

Same as above except that the CFM should be deadlock-free.

Summary of results

Approach so far:

The (safe) realisation of a (finite) set of MSCs by a weak CFM is the one where the automaton $\mathcal{A}_p$ of process p generates the projections of these MSCs on p.

Summary of results

Approach so far:

The (safe) realisation of a (finite) set of MSCs by a weak CFM is the one where the automaton $\mathcal{A}_p$ of process p generates the projections of these MSCs on p.

Results so far:

1. Conditions for (safe) realisability for finite sets of MSCs.
2. Checking safe realisability for finite sets of MSCs is in P.
3. Checking realisability for finite sets of MSCs is co-NP complete.

Some remaining questions

- Can similar results be obtained for larger classes of MSGs?

Some remaining questions

- Can similar results be obtained for larger classes of MSGs?
- What happens if we allow synchronisation messages?
  - recall that weak CFMs do not involve synchronisation messages

Some remaining questions

- Can similar results be obtained for larger classes of MSGs?
- What happens if we allow synchronisation messages?
  - recall that weak CFMs do not involve synchronisation messages
- How do we obtain a CFM realising an MSG algorithmically?
  - in particular, for non-local choice MSGs

Some remaining questions

- Can similar results be obtained for larger classes of MSGs?
- What happens if we allow synchronisation messages?
  - recall that weak CFMs do not involve synchronisation messages
- How do we obtain a CFM realising an MSG algorithmically?
  - in particular, for non-local choice MSGs
- Are there simple conditions on MSGs that guarantee realisability?
  - e.g., easily identifiable subsets of (safe) realisable MSGs

Today's lecture

Today's lecture

Today's setting

(Safe) Realisability of a regular set of MSCs.

Today's lecture

Today's setting

(Safe) Realisability of a regular set of MSCs.

Or, equivalently: (safe) realisability of a regular set of well-formed words (that is, a regular language).

Today's lecture

Today's setting

(Safe) Realisability of a regular set of MSCs.

Or, equivalently: (safe) realisability of a regular set of well-formed words (that is, a regular language).

Results:

Today's lecture

Today's setting

(Safe) Realisability of a regular set of MSCs.

Or, equivalently: (safe) realisability of a regular set of well-formed words (that is, a regular language).

Results:

1. Checking whether a regular language L is well-formed is decidable.

Today's lecture

Today's setting

(Safe) Realisability of a regular set of MSCs.

Or, equivalently: (safe) realisability of a regular set of well-formed words (that is, a regular language).

Results:

1. Checking whether a regular language L is well-formed is decidable.
2. For well-formed language L:
   - L is regular iff it is (safely) realisable by a $\forall$-bounded CFM.

Today's lecture

Today's setting

(Safe) Realisability of a regular set of MSCs.

Or, equivalently: (safe) realisability of a regular set of well-formed words (that is, a regular language).

Results:

1. Checking whether a regular language L is well-formed is decidable.
2. For well-formed language L:
   - L is regular iff it is (safely) realisable by a $\forall$-bounded CFM.
3. Checking whether an MSG is regular is undecidable.

Today's lecture

Today's setting

(Safe) Realisability of a regular set of MSCs.

Or, equivalently: (safe) realisability of a regular set of well-formed words (that is, a regular language).

Results:

1. Checking whether a regular language L is well-formed is decidable.
2. For well-formed language L:
   - L is regular iff it is (safely) realisable by a $\forall$-bounded CFM.
3. Checking whether an MSG is regular is undecidable.
4. Every (locally) communication-closed MSG is regular.

Today's lecture

Today's setting

(Safe) Realisability of a regular set of MSCs.

Or, equivalently: (safe) realisability of a regular set of well-formed words (that is, a regular language).

Results:

1. Checking whether a regular language L is well-formed is decidable.
2. For well-formed language L:
   - L is regular iff it is (safely) realisable by a $\forall$-bounded CFM.
3. Checking whether an MSG is regular is undecidable.
4. Every (locally) communication-closed MSG is regular.
5. Checking whether an MSG is comm.-closed is coNP-complete.

Today's lecture

Today's setting

(Safe) Realisability of a regular set of MSCs.

Or, equivalently: (safe) realisability of a regular set of well-formed words (that is, a regular language).

Results:

1. Checking whether a regular language L is well-formed is decidable.
2. For well-formed language L:
   - L is regular iff it is (safely) realisable by a $\forall$-bounded CFM.
3. Checking whether an MSG is regular is undecidable.
4. Every (locally) communication-closed MSG is regular.
5. Checking whether an MSG is comm.-closed is coNP-complete.
6. Checking whether an MSG is locally communication-closed is in P.

Today's lecture

Today's setting

(Safe) Realisability of a regular set of MSCs.

Or, equivalently: (safe) realisability of a regular set of well-formed words (that is, a regular language).

Results:

1. Checking whether a regular language L is well-formed is decidable.
2. For well-formed language L:
   - L is regular iff it is (safely) realisable by a $\forall$-bounded CFM.
3. Checking whether an MSG is regular is undecidable.
4. Every (locally) communication-closed MSG is regular.
5. Checking whether an MSG is comm.-closed is coNP-complete.
6. Checking whether an MSG is locally communication-closed is in P.

Overview

Regular MSCs

Let $\mathbb{M}$ be the set of MSCs over $\mathcal{P}$ and $\mathcal{C}$.

Definition (Regular)

1. $\mathcal{M} = \{ M_1, \ldots, M_n \}$ with $n \in \mathbb{N} \cup \{ \infty \}$ is called regular if $Lin(\mathcal{M}) = \bigcup_{i=1}^{n} Lin(M_i)$ is a regular word language over Act^*.
2. MSG G is regular if $Lin(G)$ is a regular word language over Act^*.
3. CFM $\mathcal{A}$ is regular if $Lin(\mathcal{A})$ is a regular word language over Act^*.

Here, Act is the set of actions in $\mathcal{M}$, G, and $\mathcal{A}$, respectively.

Regular MSCs

Let $\mathbb{M}$ be the set of MSCs over $\mathcal{P}$ and $\mathcal{C}$.

Definition (Regular)

1. $\mathcal{M} = \{ M_1, \ldots, M_n \}$ with $n \in \mathbb{N} \cup \{ \infty \}$ is called regular if $Lin(\mathcal{M}) = \bigcup_{i=1}^{n} Lin(M_i)$ is a regular word language over Act^*.
2. MSG G is regular if $Lin(G)$ is a regular word language over Act^*.
3. CFM $\mathcal{A}$ is regular if $Lin(\mathcal{A})$ is a regular word language over Act^*.

Here, Act is the set of actions in $\mathcal{M}$, G, and $\mathcal{A}$, respectively.

Lemma:

Every $\forall$-bounded CFM is regular. Why?

Examples

On the black board.

Regularity and well-formedness

Theorem [Henriksen *et. al*, 2005]

The decision problem "is a regular language $L \subseteq Act^*$ well-formed"? —that is, does L represent a set of MSCs?— is decidable.

Proof.

Regularity and well-formedness

Theorem [Henriksen *et. al*, 2005]

The decision problem "is a regular language $L \subseteq Act^*$ well-formed"? —that is, does L represent a set of MSCs?— is decidable.

Proof.

Since L is regular, there exists a minimal DFA $\mathcal{A} = (S, Act, s_0, \delta, F)$ with $\mathcal{L}(\mathcal{A}) = L$.

Regularity and well-formedness

Theorem [Henriksen *et. al*, 2005]

The decision problem "is a regular language $L \subseteq Act^*$ well-formed"? —that is, does L represent a set of MSCs?— is decidable.

Proof.

Since L is regular, there exists a minimal DFA $\mathcal{A} = (S, Act, s_0, \delta, F)$ with $\mathcal{L}(\mathcal{A}) = L$. Consider the productive states in this DFA, i.e., all states from which some state in F can be reached.

Regularity and well-formedness

Theorem [Henriksen *et. al*, 2005]

The decision problem "is a regular language $L \subseteq Act^*$ well-formed"? —that is, does L represent a set of MSCs?— is decidable.

Proof.

Since L is regular, there exists a minimal DFA $\mathcal{A} = (S, Act, s_0, \delta, F)$ with $\mathcal{L}(\mathcal{A}) = L$. Consider the productive states in this DFA, i.e., all states from which some state in F can be reached. We label every productive state s with a channel-capacity function $K_s : Ch \to \mathbb{N}$ such that four constraints (cf. next slide) are fulfilled.

Regularity and well-formedness

Theorem [Henriksen *et. al*, 2005]

The decision problem "is a regular language $L \subseteq Act^*$ well-formed"? —that is, does L represent a set of MSCs?— is decidable.

Proof.

Since L is regular, there exists a minimal DFA $\mathcal{A} = (S, Act, s_0, \delta, F)$ with $\mathcal{L}(\mathcal{A}) = L$. Consider the productive states in this DFA, i.e., all states from which some state in F can be reached. We label every productive state s with a channel-capacity function $K_s : Ch \to \mathbb{N}$ such that four constraints (cf. next slide) are fulfilled. Then: L is well-formed iff each productive state in the DFA $\mathcal{A}$ can be labelled with K_s satisfying these constraints.

Regularity and well-formedness

Theorem [Henriksen *et. al*, 2005]

The decision problem "is a regular language $L \subseteq Act^*$ well-formed"? —that is, does L represent a set of MSCs?— is decidable.

Proof.

Since L is regular, there exists a minimal DFA $\mathcal{A} = (S, Act, s_0, \delta, F)$ with $\mathcal{L}(\mathcal{A}) = L$. Consider the productive states in this DFA, i.e., all states from which some state in F can be reached. We label every productive state s with a channel-capacity function $K_s : Ch \rightarrow \mathbb{N}$ such that four constraints (cf. next slide) are fulfilled. Then: L is well-formed iff each productive state in the DFA $\mathcal{A}$ can be labelled with K_s satisfying these constraints. In fact, if a state-labelling violates any of these constraints, it is due to a word that is not well-formed. □

Constraints on state-labelling

1. $s \in F \cup \{s_0\}$, implies $K_s((p, q)) = 0$ for every channel (p, q).

Constraints on state-labelling

1. $s \in F \cup \{s_0\}$, implies $K_s((p,q)) = 0$ for every channel (p,q).
2. $\delta(s, !(p,q,a)) = s'$ implies

$$K_{s'}(c) = \begin{cases} K_s(c) + 1 & \text{if } c = (p,q) \\ K_s(c) & \text{otherwise.} \end{cases}$$

Constraints on state-labelling

1. $s \in F \cup \{s_0\}$, implies $K_s((p,q)) = 0$ for every channel (p,q).
2. $\delta(s, !(p,q,a)) = s'$ implies

$$K_{s'}(c) = \begin{cases} K_s(c) + 1 & \text{if } c = (p,q) \\ K_s(c) & \text{otherwise.} \end{cases}$$

3. $\delta(s, ?(p,q,a)) = s'$ implies $K_s((q,p)) > 0$ and

$$K_{s'}(c) = \begin{cases} K_s(c) - 1 & \text{if } c = (q,p) \\ K_s(c) & \text{otherwise.} \end{cases}$$

Constraints on state-labelling

1. $s \in F \cup \{s_0\}$, implies $K_s((p,q)) = 0$ for every channel (p,q).
2. $\delta(s, !(p,q,a)) = s'$ implies
$$K_{s'}(c) = \begin{cases} K_s(c) + 1 & \text{if } c = (p,q) \\ K_s(c) & \text{otherwise.} \end{cases}$$
3. $\delta(s, ?(p,q,a)) = s'$ implies $K_s((q,p)) > 0$ and
$$K_{s'}(c) = \begin{cases} K_s(c) - 1 & \text{if } c = (q,p) \\ K_s(c) & \text{otherwise.} \end{cases}$$
4. $\delta(s, \alpha) = s_1$ and $\delta(s_1, \beta) = s_2$ with $\alpha \in Act_p$ and $\beta \in Act_q$, $p \neq q$, implies

   not $(\alpha = !(p,q,a)$ and $\beta = ?(q,p,a))$, or $K_s((p,q)) > 0$
   implies $\delta(s, \beta) = s'_1$ and $\delta(s'_1, \alpha) = s_2$ for some $s'_1 \in S$.

Constraints on state-labelling

1. $s \in F \cup \{s_0\}$, implies $K_s((p,q)) = 0$ for every channel (p,q).
2. $\delta(s, !(p,q,a)) = s'$ implies

$$K_{s'}(c) = \begin{cases} K_s(c) + 1 & \text{if } c = (p,q) \\ K_s(c) & \text{otherwise.} \end{cases}$$

3. $\delta(s, ?(p,q,a)) = s'$ implies $K_s((q,p)) > 0$ and

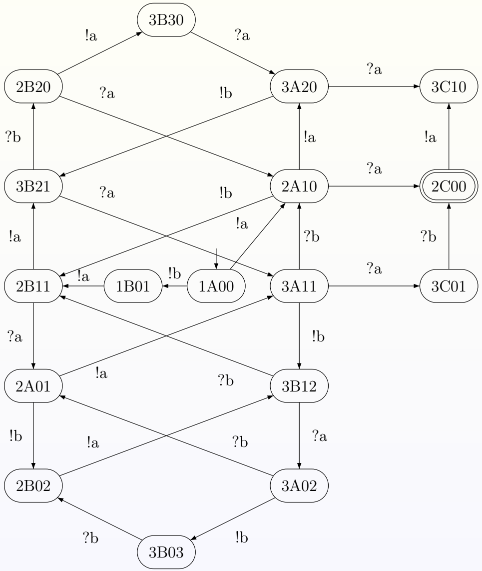
$$K_{s'}(c) = \begin{cases} K_s(c) - 1 & \text{if } c = (q,p) \\ K_s(c) & \text{otherwise.} \end{cases}$$

4. $\delta(s,\alpha) = s_1$ and $\delta(s_1,\beta) = s_2$ with $\alpha \in Act_p$ and $\beta \in Act_q$, $p \neq q$, implies

   not $(\alpha = !(p,q,a)$ and $\beta = ?(q,p,a))$, or $K_s((p,q)) > 0$
   implies $\delta(s,\beta) = s_1'$ and $\delta(s_1',\alpha) = s_2$ for some $s_1' \in S$.

These constraints can be checked in linear time in the size of relation δ.

Yannakakis' example



Boundedness and regularity

Definition (B-bounded words)

Let $B \in \mathbb{N}$ and $B > 0$. A word $w \in Act^*$ is called B-bounded if for any prefix u of w and any channel $(p, q) \in Ch$:

$$0 \leqslant \sum_{a \in \mathcal{C}} |u|_{!(p,q,a)} - \sum_{a \in \mathcal{C}} |u|_{?(q,p,a)} \leqslant B$$

Boundedness and regularity

Definition (B-bounded words)

Let $B \in \mathbb{N}$ and $B > 0$. A word $w \in Act^*$ is called B-bounded if for any prefix u of w and any channel $(p, q) \in Ch$:

$$0 \leqslant \sum_{a \in \mathcal{C}} |u|_{!(p,q,a)} - \sum_{a \in \mathcal{C}} |u|_{?(q,p,a)} \leqslant B$$

Corollary:

For any regular, well-formed language L, there exists $B \in \mathbb{N}$ and $B > 0$ such that every $w \in L$ is B-bounded.

Boundedness and regularity

Definition (B-bounded words)

Let $B \in \mathbb{N}$ and $B > 0$. A word $w \in Act^*$ is called B-bounded if for any prefix u of w and any channel $(p, q) \in Ch$:

$$0 \leqslant \sum_{a \in \mathcal{C}} |u|_{!(p,q,a)} - \sum_{a \in \mathcal{C}} |u|_{?(q,p,a)} \leqslant B$$

Corollary:

For any regular, well-formed language L, there exists $B \in \mathbb{N}$ and $B > 0$ such that every $w \in L$ is B-bounded.

Proof.

The bound B is the largest value attained by the channel-capacity functions assigned to productive states in the proof of the previous theorem. $\square$

Overview

Regularity and realisability

Theorem: [Henriksen *et al.*, 2005], [Baudru & Morin, 2007]

For any set L of well-formed words, the following four statements are equivalent:

1. L is regular.
2. L is realisable by a $\forall$-bounded CFM.
3. L is realisable by a deterministic $\forall$-bounded CFM.
4. L is safely realisable by a $\forall$-bounded CFM.

Regularity and realisability

Theorem: [Henriksen *et al.*, 2005], [Baudru & Morin, 2007]

For any set L of well-formed words, the following four statements are equivalent:

1. L is regular.
2. L is realisable by a $\forall$-bounded CFM.
3. L is realisable by a deterministic $\forall$-bounded CFM.
4. L is safely realisable by a $\forall$-bounded CFM.

Lemma:

The maximal size of the CFM realising L is such that for each process p, the number $|Q_p|$ of states of local automaton $\mathcal{A}_p$ is:

1. double exponential in the bound B and k^2, where $k = |\mathcal{P}|$, and
2. exponential in $m \log m$ where m is the size of the minimal DFA for L.

Overview

1. Realisability and safe realisability
2. Regular MSCs
3. Regularity and realisability for MSCs
4. Regularity and realisability for MSGs
   - Communication closedness

Regularity for MSGs is undecidable

Theorem [Henriksen *et. al*, 2005]

The decision problem “is MSG G regular“? is undecidable.

Proof

Outside the scope of this lecture.

Towards structural conditions for regular MSGs

- MSG G is regular if $Lin(G)$ is a regular language

Towards structural conditions for regular MSGs

- MSG G is regular if $Lin(G)$ is a regular language
- Regularity yields deterministic, or safe, but bounded CFMs

Towards structural conditions for regular MSGs

- MSG G is regular if $Lin(G)$ is a regular language
- Regularity yields deterministic, or safe, but bounded CFMs
- But, “is MSG G regular“? is unfortunately undecidable

Towards structural conditions for regular MSGs

- MSG G is regular if $Lin(G)$ is a regular language
- Regularity yields deterministic, or safe, but bounded CFMs
- But, "is MSG G regular"? is unfortunately undecidable
- Is it possible to impose structural conditions on MSGs that guarantee regularity?

Towards structural conditions for regular MSGs

- MSG G is regular if $Lin(G)$ is a regular language
- Regularity yields deterministic, or safe, but bounded CFMs
- But, “is MSG G regular“? is unfortunately undecidable
- Is it possible to impose structural conditions on MSGs that guarantee regularity?
- Yes we can. For instance, by constraining:
  1. the communication structure of the MSCs in loops of G, or
  2. the structure of expressions describing the MSCs in G

Communication graph

Definition (Communication graph)

The communication graph of the MSC $M = (\mathcal{P}, E, \mathcal{C}, l, m, <)$ is the directed graph $(V, \rightarrow)$ with:

- $V = \mathcal{P} \setminus \{ p \in \mathcal{P} \mid E_p = \varnothing \}$, the set of active processes
- $(p, q) \in \rightarrow$ if and only if $\mathcal{L}(e) = !(p, q, a)$ for some $e \in E$ and $a \in \mathcal{C}$

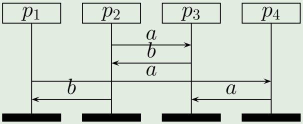
Communication graph

Definition (Communication graph)

The communication graph of the MSC $M = (\mathcal{P}, E, \mathcal{C}, l, m, <)$ is the directed graph $(V, \rightarrow)$ with:

- $V = \mathcal{P} \setminus \{\, p \in \mathcal{P} \mid E_p = \varnothing \,\}$, the set of active processes
- $(p, q) \in \rightarrow$ if and only if $\mathcal{L}(e) = !(p, q, a)$ for some $e \in E$ and $a \in \mathcal{C}$

Example



an example MSC

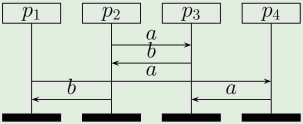
Communication graph

Definition (Communication graph)

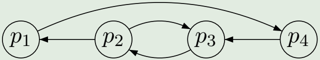
The communication graph of the MSC $M = (\mathcal{P}, E, \mathcal{C}, l, m, <)$ is the directed graph $(V, \rightarrow)$ with:

- $V = \mathcal{P} \setminus \{\, p \in \mathcal{P} \mid E_p = \varnothing \,\}$, the set of active processes
- $(p, q) \in \rightarrow$ if and only if $\mathcal{L}(e) = !(p, q, a)$ for some $e \in E$ and $a \in \mathcal{C}$

Example



an example MSC



its communication graph

Strongly connected components

Let $G = (V, \rightarrow)$ be a directed graph.

Strongly connected component

- $T \subseteq V$ is strongly connected if for every $v, w \in T$, vertices v and w are mutually reachable (via $\rightarrow$) from each other.

Strongly connected components

Let $G = (V, \rightarrow)$ be a directed graph.

Strongly connected component

- $T \subseteq V$ is strongly connected if for every $v, w \in T$, vertices v and w are mutually reachable (via $\rightarrow$) from each other.
- T is a strongly connected component (SCC) of G it T is strongly connected and T is not properly contained in another SCC.

Strongly connected components

Let $G = (V, \rightarrow)$ be a directed graph.

Strongly connected component

- $T \subseteq V$ is strongly connected if for every $v, w \in T$, vertices v and w are mutually reachable (via $\rightarrow$) from each other.
- T is a strongly connected component (SCC) of G it T is strongly connected and T is not properly contained in another SCC.

Determining the SCCs of a digraph can be done in linear time in the size of V and $\rightarrow$.

Communication closedness

Communication closedness

A loop is simple if it visits a vertex at most once, except for the start- and end-vertex which are visited twice.

Communication closedness

A loop is simple if it visits a vertex at most once, except for the start- and end-vertex which are visited twice.

Definition (Communication closedness)

MSG G is communication-closed if for every simple loop $\pi = v_1 v_2 \ldots v_n$ (with $v_1 = v_n$) in G, the communication graph of the MSC $M(\pi) = \lambda(v_1) \bullet \lambda(v_2) \bullet \ldots \bullet \lambda(v_n)$ is strongly connected.

Communication closedness

A loop is simple if it visits a vertex at most once, except for the start- and end-vertex which are visited twice.

Definition (Communication closedness)

MSG G is communication-closed if for every simple loop $\pi = v_1 v_2 \ldots v_n$ (with $v_1 = v_n$) in G, the communication graph of the MSC $M(\pi) = \lambda(v_1) \bullet \lambda(v_2) \bullet \ldots \bullet \lambda(v_n)$ is strongly connected.

Example

On the black board.

Communication-closed vs. regularity

Theorem:

Every communication-closed MSG G is regular.

Example

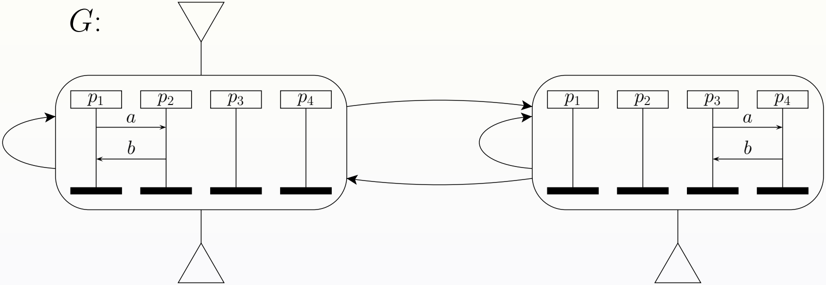
Example on the black board.

Note:

The converse does not hold (cf. next slide).

Communication-closed vs. regularity

Communication-closedness is not a necessary condition for regularity:



MSG G is not communication-closed, but $Lin(G)$ is regular.

Checking communication-closedness

Theorem: [Genest *et. al*, 2006]

The decision problem "is MSG G communication closed?" is co-NP complete.

Checking communication-closedness

Theorem: [Genest *et. al*, 2006]

The decision problem "is MSG G communication closed?" is co-NP complete.

Proof

1. Membership in co-NP can be proven in a standard way: guess a sub-graph of G, check in polynomial time whether this sub-graph has a loop passing through all its vertices, and check whether its communication graph is not strongly connected.
2. Co-NP hardness can be shown by a reduction from the 3-SAT problem.

Communication-closed vs. regularity

Definition (Asynchronous iteration)

For $\mathcal{M}_1, \mathcal{M}_2 \subseteq \mathbb{M}$ sets of MSCs, let:

$$\mathcal{M}_1 \bullet \mathcal{M}_2 \;=\; \{\, M_1 \bullet M_2 \mid M_1 \in \mathcal{M}_1, M_2 \in \mathcal{M}_2 \,\}$$

For $\mathcal{M} \subseteq \mathbb{M}$ let

$$\mathcal{M}^i \;=\; \begin{cases} \{M_\epsilon\} & \text{if } i{=}0\text{, where } M_\epsilon \text{ denotes the empty MSC} \\ \mathcal{M} \bullet \mathcal{M}^{i-1} & \text{if } i > 0 \end{cases}$$

The asynchronous iteration of $\mathcal{M}$ is now defined by:

$$\mathcal{M}^* = \bigcup_{i \geqslant 0} \mathcal{M}^i.$$

Finitely generated

Definition (Finitely generated)

Set of MSCs $\mathcal{M}$ is finitely generated if there is a finite set of MSCs $\widehat{\mathcal{M}}$ such that $\mathcal{M} \subseteq \widehat{\mathcal{M}}^*$.

Finitely generated

Definition (Finitely generated)

Set of MSCs $\mathcal{M}$ is finitely generated if there is a finite set of MSCs $\widehat{\mathcal{M}}$ such that $\mathcal{M} \subseteq \widehat{\mathcal{M}}^*$.

Remarks:

1. Each set of MSCs defined by an MSG G is finitely generated.
2. Not every regular well-formed language is finitely generated.
3. Not every finitely generated set of MSCs is regular.
4. It is decidable to check whether a set of MSCs is finitely generated.

Characterisation of communication-closedness

Theorem: [Henriksen *et. al*, 2005]

Let $\mathcal{M}$ be a (possibly infinite) set of MSCs. Then:

$$\mathcal{M} \text{ is finitely generated and regular}$$

$$\text{iff}$$

$$\mathcal{M} = \mathcal{L}(G) \text{ for some communication-closed MSG } G.$$

Local communication-closedness

Definition (Local communication-closedness)

MSG G is locally communication-closed if for each vertex (v, v') in G, the MSCs $\lambda(v)$, $\lambda(v')$, and $\lambda(v) \bullet \lambda(v')$ all have weakly connected communication graphs.

Local communication-closedness

Definition (Local communication-closedness)

MSG G is locally communication-closed if for each vertex (v, v') in G, the MSCs $\lambda(v)$, $\lambda(v')$, and $\lambda(v) \bullet \lambda(v')$ all have weakly connected communication graphs.

Notes:

1. A directed graph is weakly connected if its induced undirected graph (obtained by ignoring the directions of edges) is strongly connected.
2. Checking whether MSG G is locally communication-closed can be done in linear time.

Locally communication-closed MSGs are realisable

Theorem: [Genest *et al.*, 2006]

Every locally communication-closed MSG G is realisable by a CFM $\mathcal{A}$ of size $m^{\mathcal{O}(|\mathcal{P}|)}$ where m is the number of vertices in G.