Theoretical Foundations of the UML

Lecture 12: Realisability for Regular Sets of MSCs

Joost-Pieter Katoen

Lehrstuhl für Informatik 2
Software Modeling and Verification Group

`moves.rwth-aachen.de/teaching/ss-20/fuml/`

May 26, 2020

Outline

1. Realisability and safe realisability
2. Regular MSCs , MSGs, CFMs
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 $\mathcal{A}$.

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.

sufficient + necessary conditions

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?
- 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 local-choice MSGs

sufficient

- Are there "simple" conditions on MSGs that guarantee realisability?
  - e.g., easily identifiable subsets of (safe) realisable MSGs

easy to check

Today's lecture

Today's lecture

(Safe) Realisability of a regular set of MSCs.

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

regular language

Today's lecture

(Safe) Realisability of a regular set of MSCs.

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

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 communication-closed MSG is regular.
5. Checking whether an MSG is comm.-closed is coNP-complete.

= same complexity as checking whether a finite set of MSCs is realisable by a weak CFM.

Today's lecture

(Safe) Realisability of a regular set of MSCs.

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

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 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

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

Regular MSCs

Definition (Regular MSCs, MSGs, and CFMs)

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.

$Lin(M) = a^* b^*$

ε is regular
$\emptyset$ is regular
$\forall a \in \Sigma.$ $\{a\}$ is regular

if A and B are regular, then $A+B$ (or $A \cup B$) $A.B$ and A^*

Regular MSCs

Definition (Regular MSCs, MSGs, and CFMs)

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:

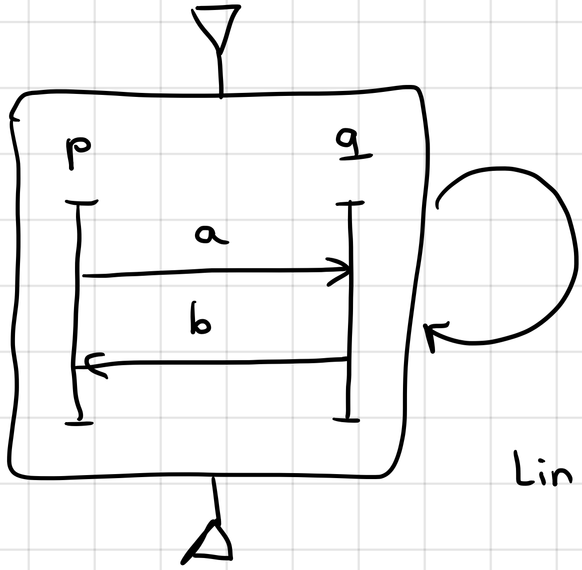
Every $\forall$-bounded CFM is regular. Why?

has finitely many configurations.

its configuration graph is a finite-state automaton

Examples

A.



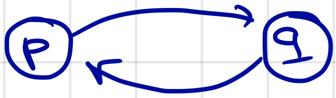
Claim: this MSG is regular.

$$\mathrm{Lin}(A) = \big(!(p,q,a)\ ?(q,p,a)\ !(q,p,b)\ ?(p,q,b)\big)^+$$

$$= (!a\ ?a\ !b\ ?b)^+$$

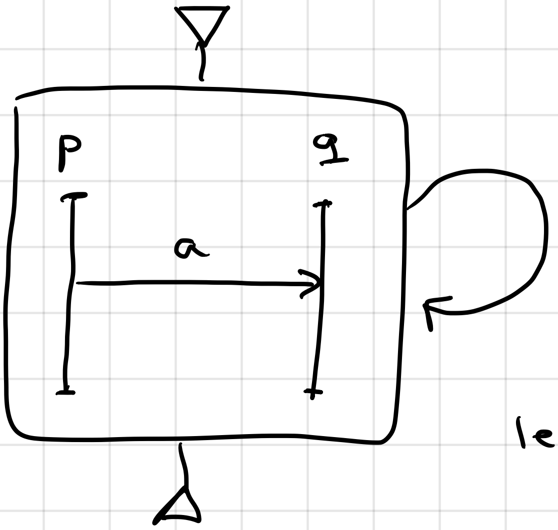
is a regular language

communication closed



B.

G'

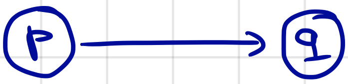


Claim: this MSG is <u>not</u> regular.

let $!(p,q,a)$ be "[" (open bracket)

$?(q,p,a)$ be "]" (closing bracket)

not comm. closed



Then $L(G')$ = Dyck language = <u>not</u> regular

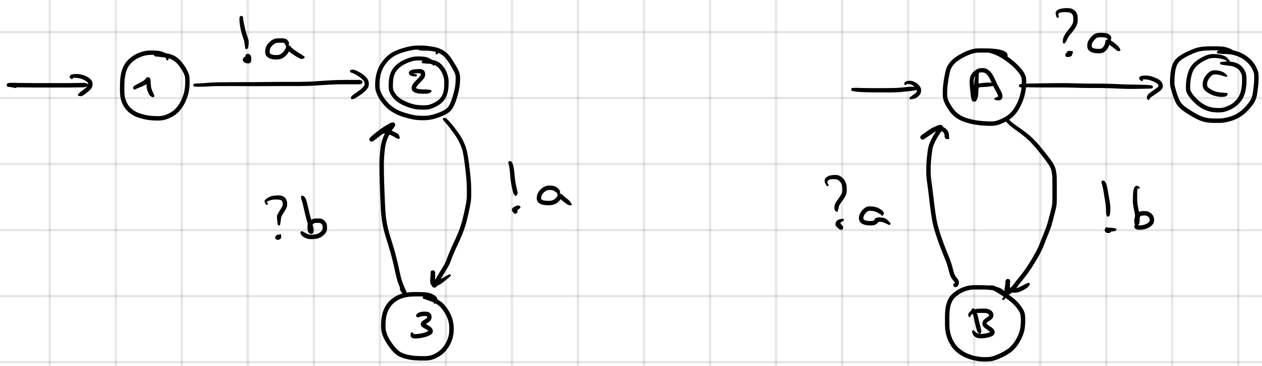
= language of words that contained a balanced set of brackets "[" and "]"

e.g. [[ ]] or [[ ] [[ ]]] etc.

<u>not</u> [ ] [[ ]

(Yannakakis example)

C. CFM



A_p A_q

<u>Claim</u>: this CFM is regular, as it is $\forall 3$ bounded

Regularity and well-formedness

Theorem [Henriksen *et. al*, 2005]

The decision problem "is a regular language $L \subseteq Act^*$ well-formed"? —that is, does regular 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. 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).
2. $\delta(s, !(p,q,a)) = s'$ implies

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

$$s \xrightarrow{!(p,q,a)} s'$$

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

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

$$s \xrightarrow{?(p,q,a)} s'$$

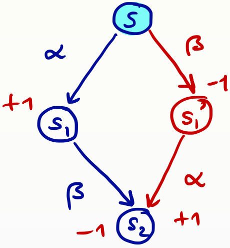
Constraints on state-labelling

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

$k(p,q)=3$

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 $s_1' \in S$.

diamond

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

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

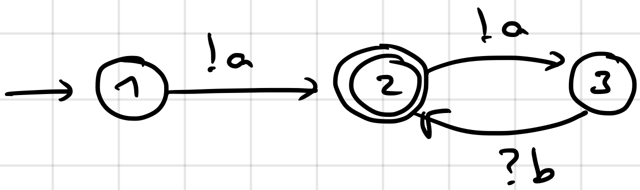
4. $\delta(s,\alpha) = s_1$ and $\delta(s_1,\beta) = s_2$ with $\alpha \in Act_p$ and $\beta \in Act_q$, $p \neq q$, implies

   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 $s_1' \in S$.

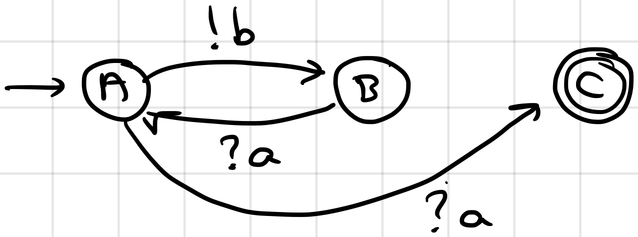
These constraints can be checked in linear time in the size of relation δ.

Yannakakis CFM

A_p



A_q



configurations $1A\underline{0}\underline{0}$ — $\begin{cases} A_p \text{ is in state } 1 \\ A_q \text{ is in state } A \\ \underline{k(p,q)} = \underline{k(q,p)} = 0 \end{cases}$

Configuration graph is defined by the following rules

$1A00 \xrightarrow{!a} 2A10$

$ \xrightarrow{!b} 1B01$

$2Anm \xrightarrow{!a} 3A\ n{+}1\ m$

$ \xrightarrow{!b} 2B\ n\ m{+}1$

$ \xrightarrow{?a} 2C\ n{-}1\ m$ if $n > 0$

$2Bnm \xrightarrow{!a} 3B\ n{+}1\ m$

$ \xrightarrow{?a} 2A\ n{-}1\ m$ if $n > 0$

$3Anm \xrightarrow{?b} 2A\ n\ m{-}1$ if $m > 0$

$ \xrightarrow{!b} 3B\ n\ m{+}1$

$ \xrightarrow{?a} 3C\ n{-}1\ m$ if $n > 0$

$3Bnm \xrightarrow{?b} 2B\ n\ m{-}1$ if $m > 0$

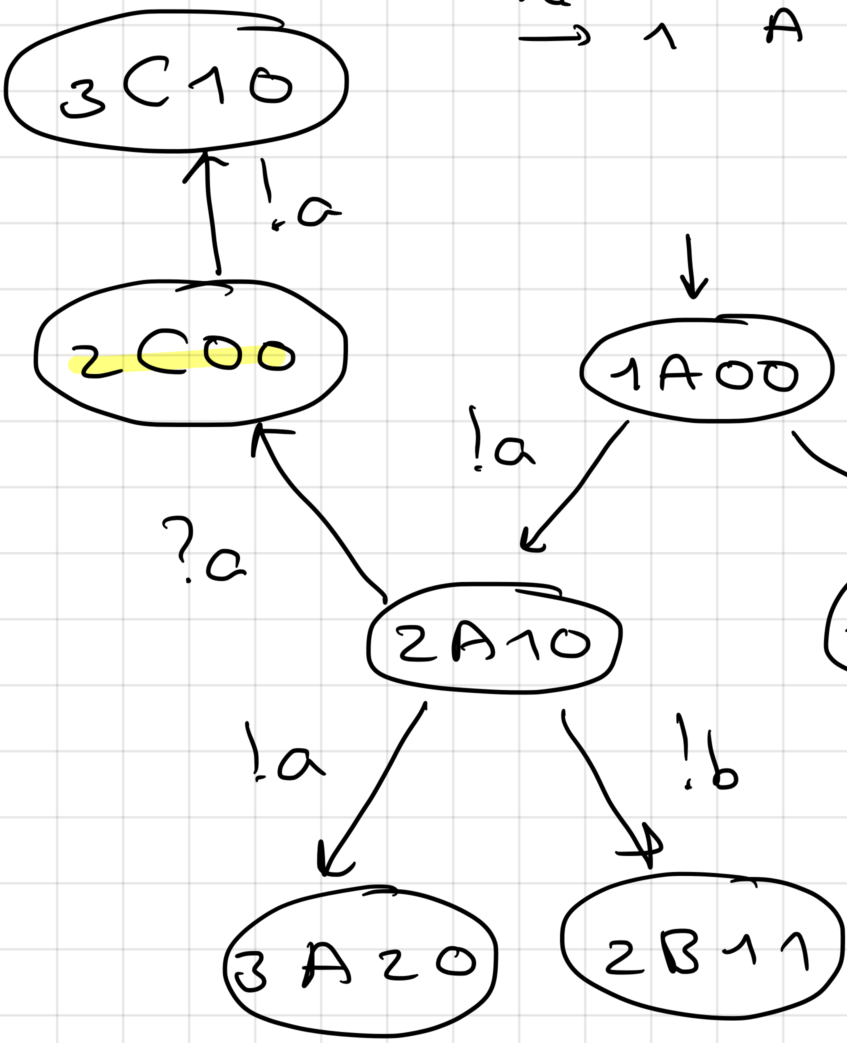
$ \xrightarrow{?a} 3A\ n{-}1\ m$ if $n > 0$

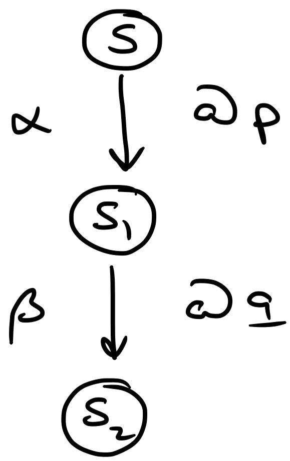
$3\ C\ n\ m \xrightarrow{?b} 2\ C\ n\ m-1$ if $m>0$

$2\ C\ n\ m \xrightarrow{!a} 3\ C\ n+1\ m$

$1\ B\ n\ m \xrightarrow{!a} 2\ B\ n+1\ m$

$\xrightarrow{?a} 1\ A\ n-1\ m$ if $n>0$

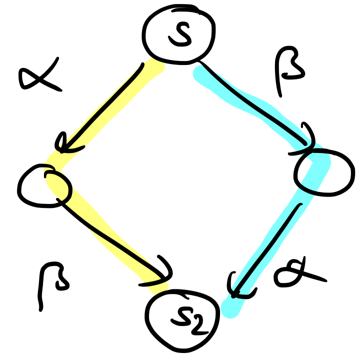


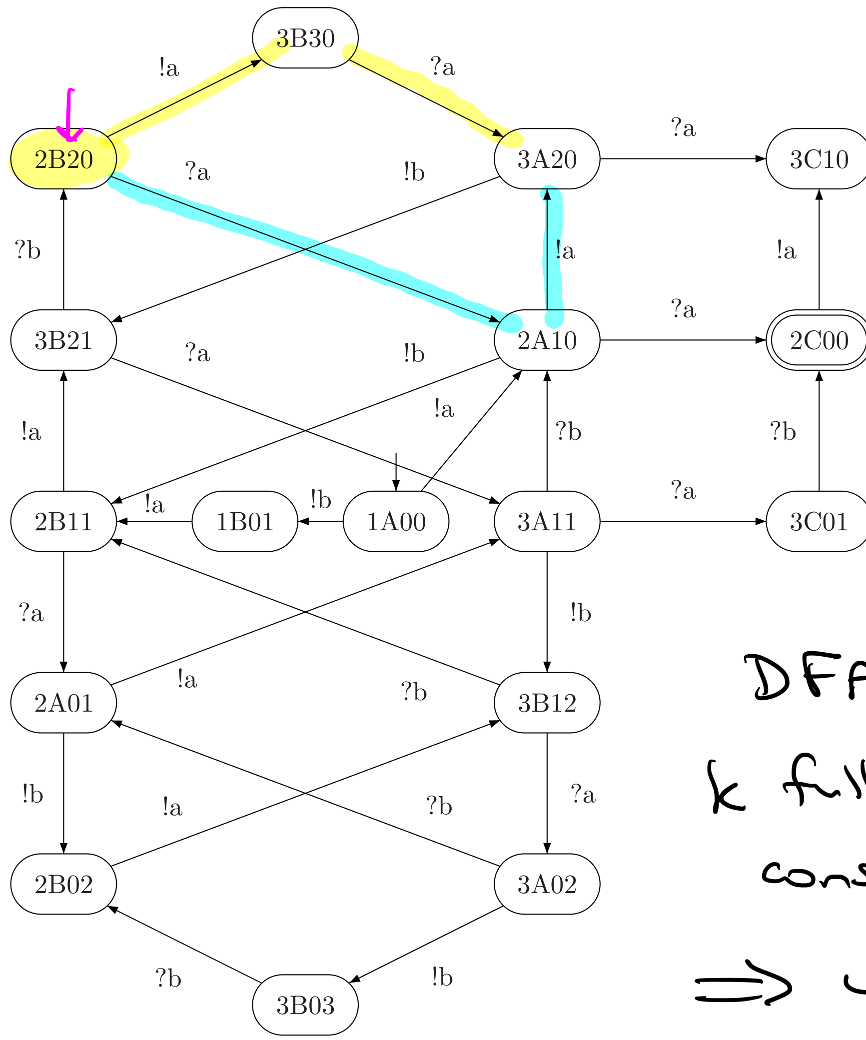


$\wedge$

α, β are not matched or $k(p,q) > 0$

$\Longrightarrow$

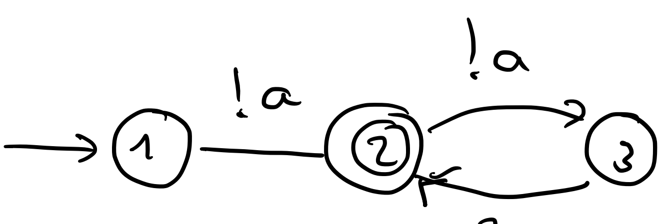




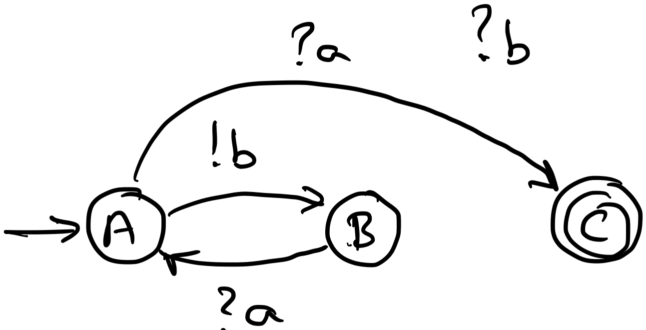
DFA

k fullfils all constraints

$\Longrightarrow$ well-formed



A_p



A_q

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. □

Overview

Regularity and realisability

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

For well-formed L, the following four statements are equivalent:

1. L is regular.
2. L is realisable by a $\forall$-bounded CFM.

cf lecture 10

3. L is realisable by a deterministic $\forall$-bounded CFM.
4. L is safe realisable by a $\forall$-bounded CFM. } cf. lecture 11

Regularity and realisability

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

For well-formed L, 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 safe 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

Regularity for MSGs is undecidable

is Lin(G) a regular language?

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
- Regularity yields deterministic, or safe, but bounded CFMs
- But, "is MSG G regular"? is unfortunately undecidable
- Is it possible to impose simple structural conditions on MSGs that guarantee regularity?

MSG $\mathcal{G}$ satisfies the structural conditions $\xrightarrow{\text{implies}}$ $\mathcal{G}$ is realisable

Towards structural conditions for regular MSGs

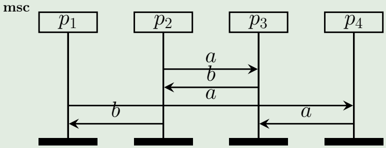
- MSG G is regular if $Lin(G)$ is a regular language
- Regularity yields deterministic, or safe, but bounded CFMs
- But, "is MSG G regular"? is unfortunately undecidable
- Is it possible to impose structural conditions on MSGs that guarantee regularity?
- Yes we can. For instance, by constraining:
 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

Definition (Communication graph)

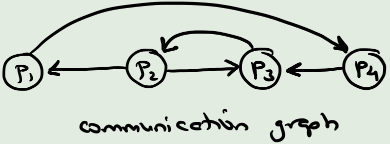
The communication graph of the MSC $M = (\mathcal{P}, E, \mathcal{C}, l, m, <)$ is the directed graph $(V, \rightarrow)$ with:

- $V = \mathcal{P} \setminus \{ p \in \mathcal{P} \mid E_p = \varnothing \}$, 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



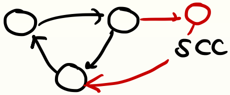
Strongly connected components

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

Strongly connected component

- $\underline{T \subseteq V}$ is strongly connected if for every $\underline{v, w \in T}$, 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.

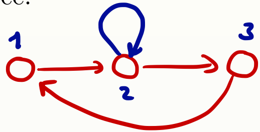
Determining the SCCs of a digraph can be done in linear time in the size of V and $\rightarrow$.

e.g. depth-first algorithm

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.



1 2 2 3 not simple

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)

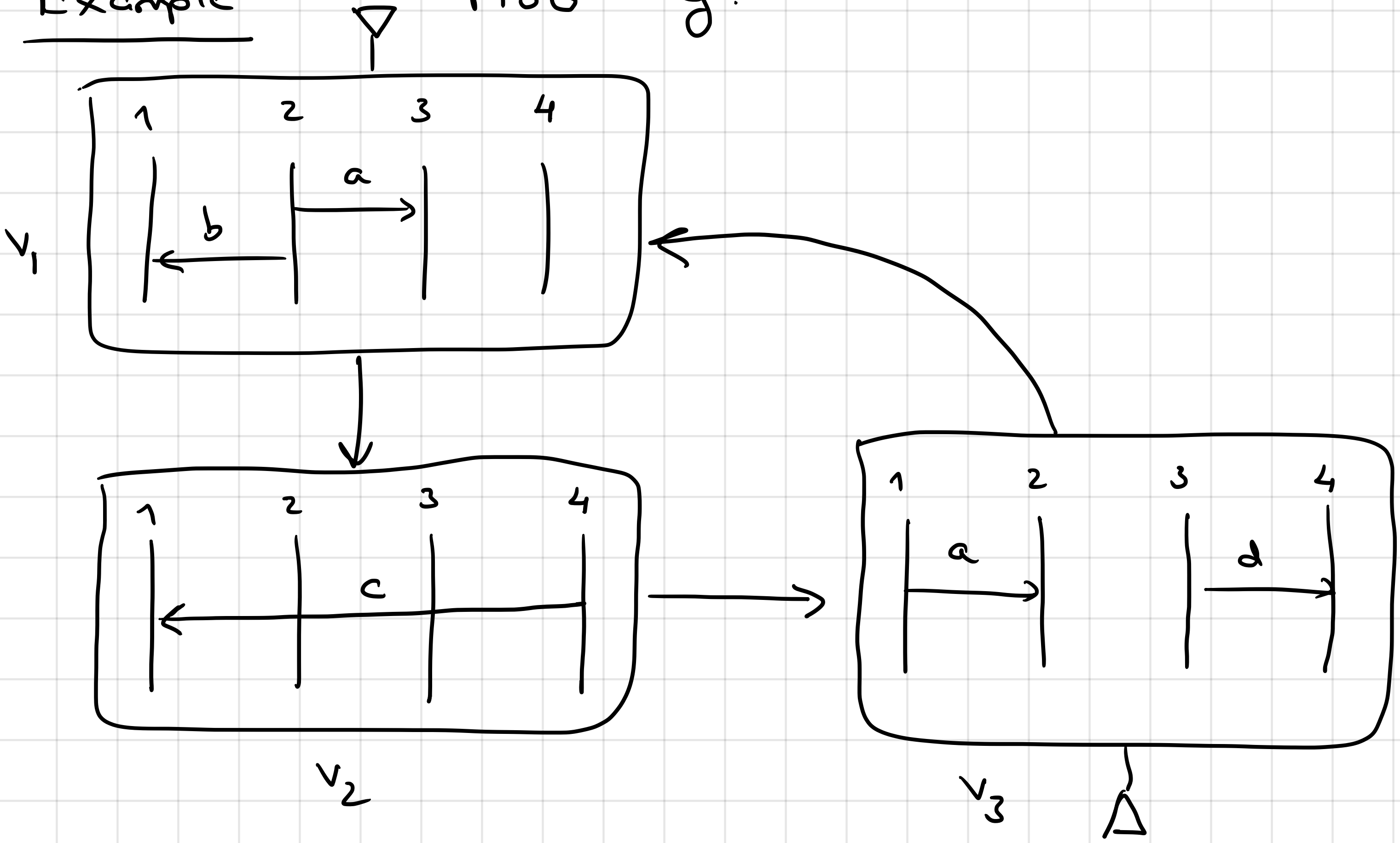
MSG G is communication-closed if for every simple loop $\pi = v_1 v_2 \ldots v_n$ (with $v_1 = v_n$) in G, the communication graph of the MSC $M(\pi) = \lambda(v_1) \bullet \lambda(v_2) \bullet \ldots \bullet \lambda(v_n)$ is strongly connected.

Example

On the black board.

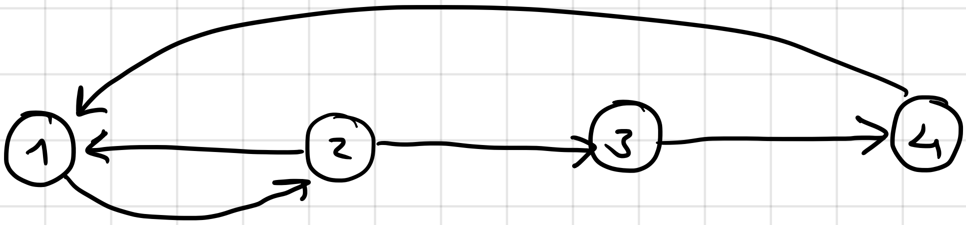
Example

MSG $\mathcal{G}$:



single loop (which is simple): $v_1\ v_2\ v_3\ v_1 = \pi$

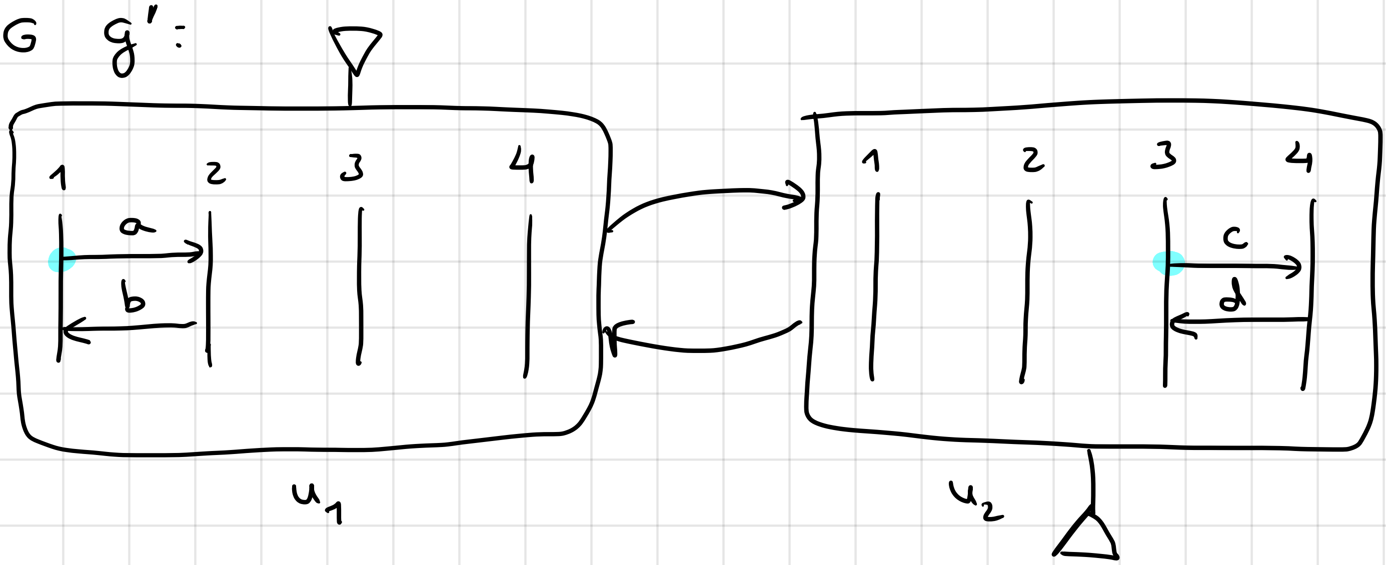
communication graph of $M(\pi)$: $\lambda(v_1) \cdot \lambda(v_2) \cdot \lambda(v_3)$



strongly connected

Thus $\mathcal{G}$ is communication closed.

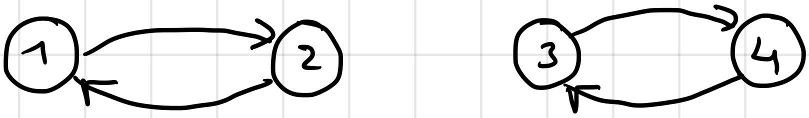
MSG $\mathcal{G}'$:



Communication graph of $\mathcal{G}'$:

$$\pi = u_1 u_2 u_1$$

$M(\pi)$:



not strongly connected

Thus, $\mathcal{G}'$ is <u>not</u> communication closed

$L(\mathcal{G}')$ is not regular. To see this, consider

$$Lin(\mathcal{G}') \upharpoonright \{!a, !c\} =$$

$$\{ \sigma \in \{!a, !c\}^* \mid \#_{!a}\sigma = \#_{!c}\sigma > 0 \}$$

is not regular. As regular languages are closed under projections, $Lin(\mathcal{G}')$ is <u>not</u> regular.

Communication-closed vs. regularity

Theorem:

Every $\underbrace{\text{communication-closed}}_{\text{sufficient criterion for being regular}}$ MSG G is regular.

Example

Example on the black board.

Note:

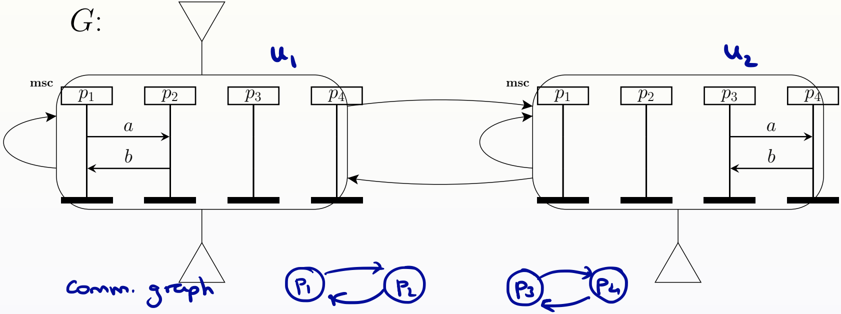
The converse does not hold (cf. next slide).

Communication-closed vs. regularity

$\underbrace{(!a\,?a\,!b\,?b)^+}_{\text{between } p_1 \text{ and } p_2}$ interleaved $\underbrace{(!a\,?a\,!b\,?b)^+}_{\text{between } p_3 \text{ and } p_4}$

Communication-closedness is not a necessary condition for regularity:

G:



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.

equally hard as checking whether a finite set of MSCs is realisable by a weak CFM (cf. lecture 10)

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. (in poly time)
2. Co-NP hardness can be shown by a polynomial reduction from the 3-SAT problem.

Theorem Checking whether MSG $\mathcal{G}$ is comm.-closed is coNP-hard.

Proof: Polynomial reduction from the 3SAT-problem.

3SAT: consider the Boolean formula

$$\phi = \underbrace{C_1 \wedge \ldots\ldots \wedge C_m}_{\text{clauses}}$$

over the variables $\{x_1, \ldots, x_n\}$ such that clause

$$C_j = l_j^1 \vee l_j^2 \vee l_j^3$$

literals equals x_k or $\bar{x}_k$ for some $k \in \{1, \ldots, n\}$

ϕ is satisfiable if $\exists$ valuation for x_1 through x_n. for every m, C_m is true.

Fact: 3SAT is NP-complete. Its complement is also NP-complete, thus 3SAT is also coNP-complete.

Reduction:

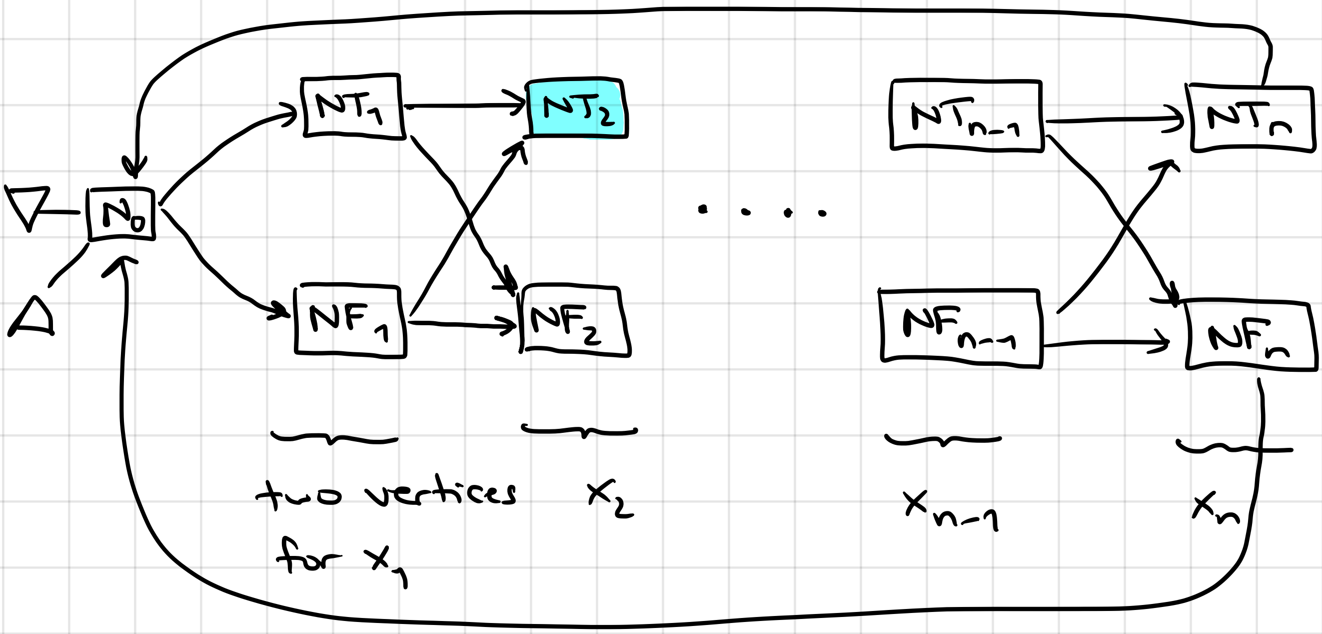
3SAT-formula $\longmapsto$ MSG $\mathcal{G}$

$\phi = C_1 \wedge \ldots \wedge C_m$ over $\{x_1, \ldots, x_n\}$

such that ϕ is satisfiable iff $\mathcal{G}$ has a simple loop that is not strongly connected.

$\mathcal{G}$ is not comm. closed.

The structure of MSG G is as follows: n variables



if $x_i = \text{true}$ then traverse vertex NT_i

if $x_i = \text{false}$ ($\overline{x_i} = \text{true}$) then traverse NF_i

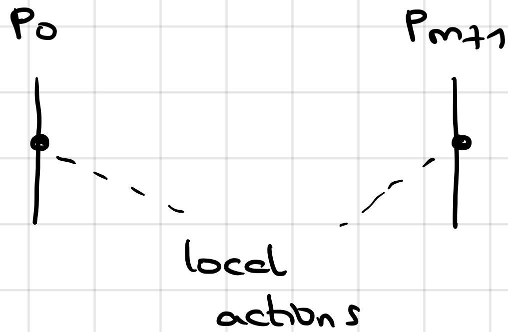
$\phi = C_1 \wedge \dots \wedge C_m \qquad C_j = l_j^1 \vee l_j^2 \vee l_j^3$

Processes of the MSCs:

$$\{ P_0 , \underbrace{P_1^1, P_1^2, P_1^3}_{\substack{\text{correspond}\\ \text{to literals}\\ l_1^1\ l_1^2\ l_1^3}} , \dots\dots , P_m^1, P_m^2, P_m^3 , P_{m+1} \}$$

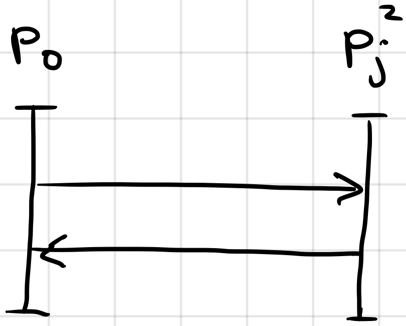
$|P| = 3 \times \#\text{clauses} + 2$

$\lambda(N_0) =$



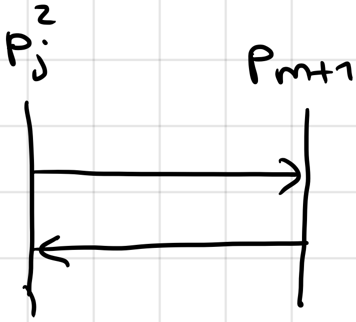
Template MSCs: $\underbrace{LT_j^1 \, , \, LT_j^2}_{\text{empty}} \, , \, LT_j^3$ (true)

LT_j^3 ::

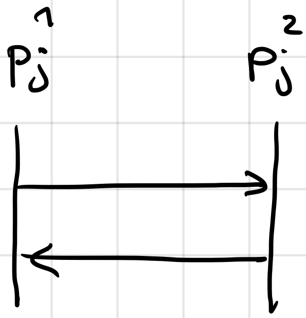


Also have templates LF_j^1, LF_j^2, LF_j^3

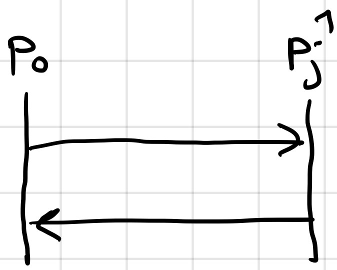
LF_j^3



LF_j^2



LT_j^1



$\lambda(NT_i) =$ the concatenation of all template MSCs LT_j^k with $l_j^k = x_i$

$+$ $(k = 1, 2, 3)$

all template MSCs LF_j^k with $l_j^k = \overline{x_i}$

$\lambda(NF_i) =$ concatenate:

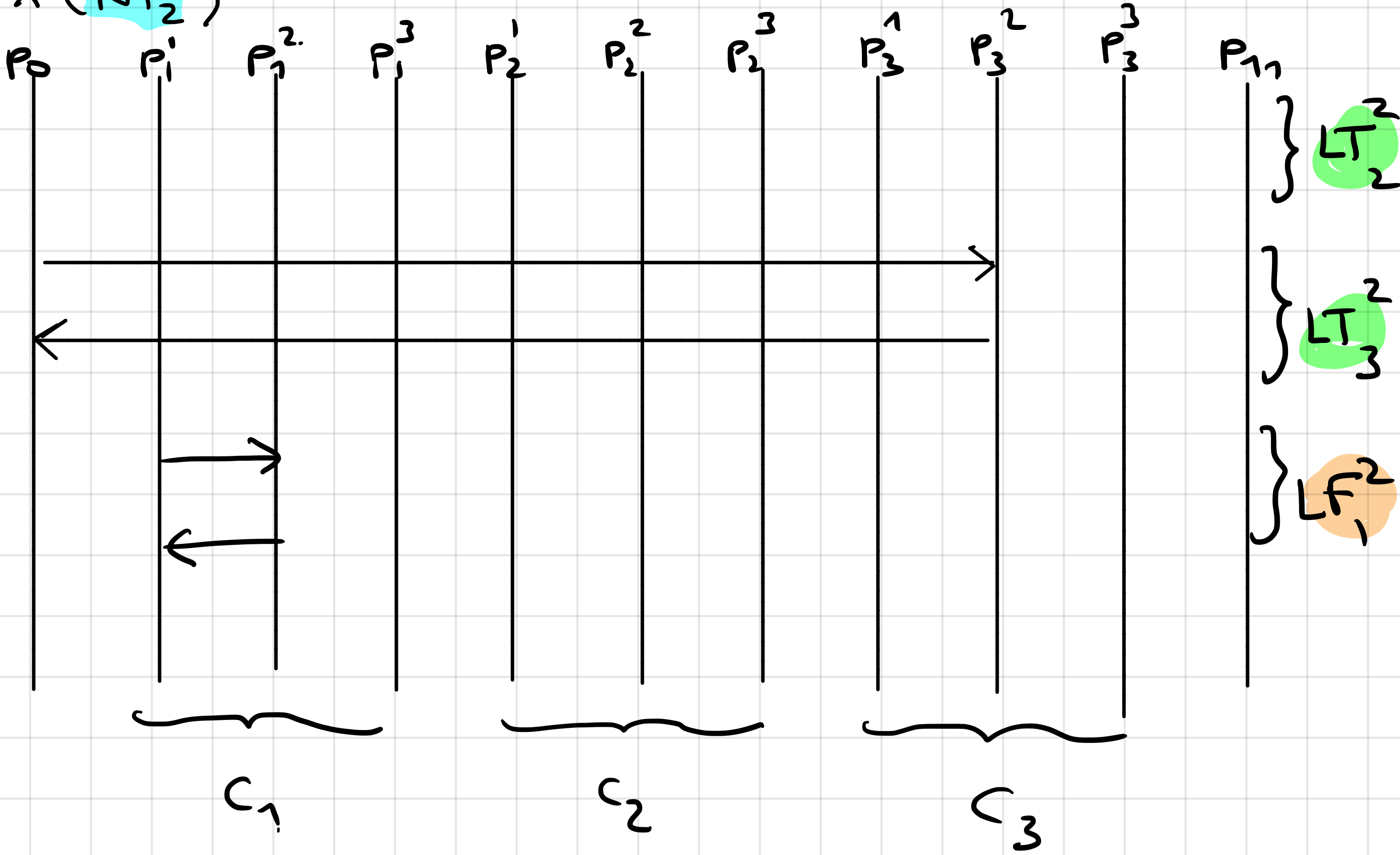
LT_j^k with $l_j^k = \overline{x_i}$

and LF_j^k with $l_j^k = x_i$

Example

$$
\begin{aligned}
\phi = \ & (x_1 \vee \overline{x_2} \vee x_3) && C_1 \\
\wedge \ & (\overline{x_1} \vee x_2 \vee x_3) && C_2 \\
\wedge \ & (x_1 \vee x_2 \vee \overline{x_3}) && C_3
\end{aligned}
$$

$\lambda(NT_2)$



$$\lambda(NT_2) = LT_2^2 \cdot LT_3^2 \cdot LF_1^2$$

Claim processes P_0 and P_{m+1} are connected in the communication graph of MSG G iff there exists a clause in ϕ for which all literals are false.

Asynchronous iteration

Definition

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 $\mathcal{M} \subseteq \widehat{\mathcal{M}}^*$.

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:

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

iff

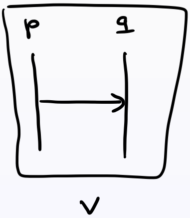
$$\underline{\mathcal{M}} = \underline{\mathcal{L}(G)} \text{ for some } \underline{\text{communication-closed MSG } G}.$$

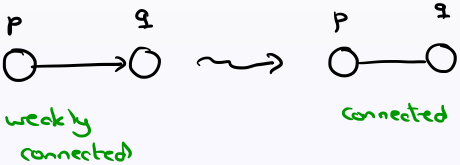
Local communication-closedness

Definition (Local communication-closedness)

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

communication graph,
ignore the direction of edges
$\longrightarrow$ undirected graph





Local communication-closedness

Definition (Local communication-closedness)

MSG G is locally communication-closed if for each edge (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)