

# Safety Verification of Communicating One-Counter Machines

A. Heußner

Univ. of Bamberg

T. Le Gall

CEA Saclay

G. Sutre

LaBRI Bordeaux

FSTTCS 2012, Hyderabad

# Agenda

⇨ Example: Sliding Window Protocol

⇨ Systems of Communicating One-Counter Machines and their Topology Parametrized Reachability Problem

⇨ Main Theorem: Proof of “Only If” Direction

⇨ Main Theorem: Proof of “If” Direction

⇨ On-going/Future work

# Agenda

⇒ Example: Sliding Window Protocol

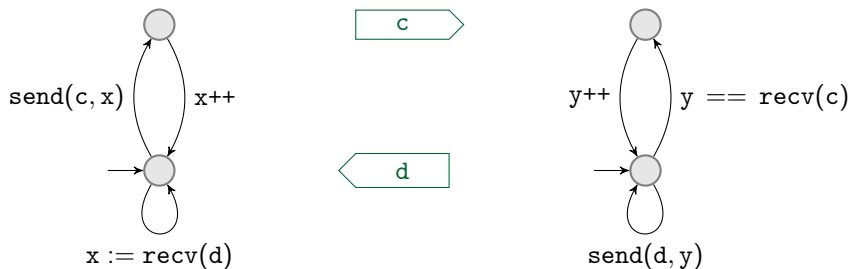
⇒ Systems of Communicating One-Counter Machines and their Topology Parametrized Reachability Problem

⇒ Main Theorem: Proof of “Only If” Direction

⇒ Main Theorem: Proof of “If” Direction

⇒ On-going/Future work

# Example Sliding Window Protocol

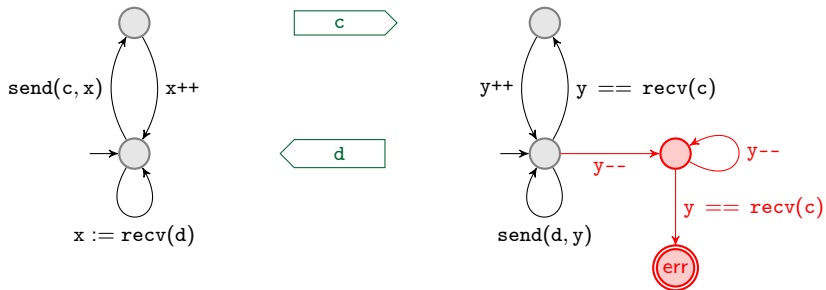


- ⇒ One local counter for each process ( $x, y$ )
- ⇒ Asynchronous communication via perfect channels ( $c, d$ )
  - send the counter's value
  - receive and test/overwrite the counter
- ⇒ Sources of **infinity**: messages, channel length, local counters

# Safety Verification of Example

Goal:

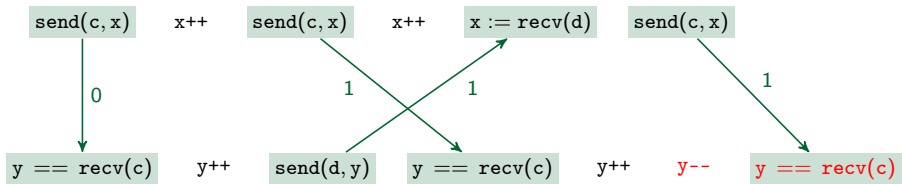
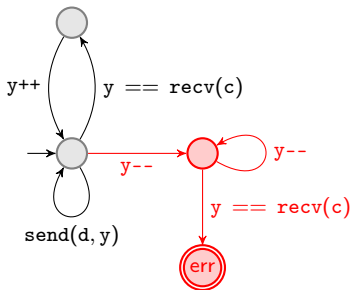
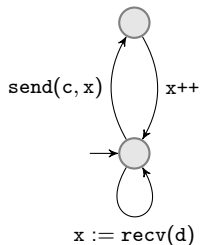
Check absence of unspecified receptions due to  $y$  being too large



Goal:

Check that  $\text{err}$  is not **reachable**

# An Erroneous Execution



# Agenda

⇒ Example: Sliding Window Protocol

⇒ Systems of Communicating One-Counter Machines and their Topology Parametrized Reachability Problem

⇒ Main Theorem: Proof of “Only If” Direction

⇒ Main Theorem: Proof of “If” Direction

⇒ On-going/Future work

# Communication Topologies

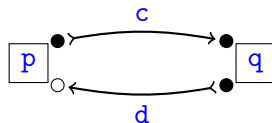
## Communication types

- **strong** : standard CFSM-style communication ( $==$ )
- **weak** : counter lost by communication ( $:=$ )

### Definition:

A **topology** is  $\mathcal{T} = \langle P, C, \text{src}, \text{dst} \rangle$  where

- ⇒  $P$  : finite set of **processes**
- ⇒  $C$  : finite set of **channels**
- ⇒  $\text{src}, \text{dst} : C \rightarrow P \times \{\bullet, \circ\}$





# Communicating One-counter Machines

## Definition:

A **communicating one-counter machine** is  $\langle S, I, F, A, \Delta \rangle$  where

- $\Leftrightarrow S$  : finite set of **states**
- $\Leftrightarrow I, F \subseteq S$  : **initial** and **final** states
- $\Leftrightarrow A$  : finite set of **actions**
- $\Leftrightarrow \Delta \subseteq S \times A \times S$  : finite set of transition **rules**

**Actions** :  $\text{add}(k) \mid \text{test}(\varphi) \mid c! \mid c? \quad (k \in \mathbb{Z}, \varphi \in \mathcal{P}_1, c \in C)$

## Definition:

A **system of comm. one-counter machines** is  $\langle \mathcal{T}, (\mathcal{M}^p)_{p \in P} \rangle$  where

- $\Leftrightarrow \mathcal{T}$  : topology
- $\Leftrightarrow \mathcal{M}^p$  : communicating one-counter machine

# SC1CM Semantics: Configurations

Recall:

A SC1CM is  $\langle \mathcal{T}, (\mathcal{M}^P)_{P \in P} \rangle$  where  $\mathcal{M}^P = \langle S^P, I^P, F^P, A^P, \Delta^P \rangle$

A configuration is  $( \begin{matrix} \prod_{P \in P} S^P \\ \cup \\ \mathbf{s} \end{matrix} , \begin{matrix} \mathbb{N}^P \\ \cup \\ \mathbf{x} \end{matrix} , \begin{matrix} (\mathbb{N}^*)^C \\ \cup \\ \mathbf{w} \end{matrix} )$

*initial*  $\stackrel{\text{def}}{\iff} s^P \in I^P \wedge \mathbf{x} = \mathbf{0} \wedge \mathbf{w} = \epsilon$   
*final*  $\stackrel{\text{def}}{\iff} s^P \in F^P$

# SC1CM Semantics: Transitions

Recall:

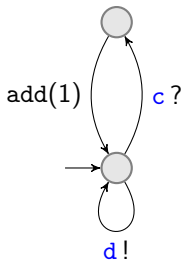
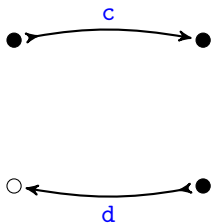
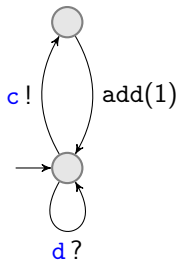
A SC1CM is  $\langle \mathcal{T}, (\mathcal{M}^P)_{P \in \mathcal{P}} \rangle$  where  $\mathcal{M}^P = \langle S^P, I^P, F^P, A^P, \Delta^P \rangle$

The **transition** relation  $(s, x, w) \xrightarrow{a} (s', x', w')$  is defined by

- ⇒ exactly one process moves
- ⇒ counter actions behave as expected
- ⇒ communication actions depend on the endpoint's type

$$\begin{array}{ll} \bullet \xrightarrow{c} c! & \equiv c!x & \circ \xrightarrow{c} c! & \equiv c!x ; x := \text{any} \\ \xrightarrow{c} \bullet c? & \equiv c?x & \xrightarrow{c} \circ c? & \equiv x := \text{any} ; c?x \end{array}$$

# Example Sliding Window Protocol



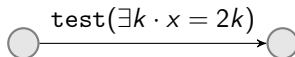
# On Presburger Tests versus Zero Tests

**Idea:**

Simulation of  $\text{test}(\varphi)$  by zero-tests in absence of communication

- ⇒  $\llbracket \varphi \rrbracket = A \cup (B + m\mathbb{N})$  where  $A, B \subseteq \mathbb{N}$  finite, and  $m \in \mathbb{N}$ .
- ⇒ Maintain  $(x \bmod m)$  in the state

Does not work when the initial counter value is unknown ( $c?$  for  $\xrightarrow{c}\circ$ )



**Idea:**

Simulation of  $\text{test}(\varphi)$  by communication to a slave process

- ⇒ delegate the test with a send over channel  $\bullet \xrightarrow{\quad} \circ$
- ⇒ the slave checks that its input messages satisfy  $\varphi$

# Parametrized Reachability Problem

**Definition:**

Given a topology  $\mathcal{T}$ , the decision problem  $\text{RP-SC1CM}(\mathcal{T})$  is

**Input:** a system of communicating one-counter machines  $\mathcal{S}$   
with topology  $\mathcal{T}$

**Output:** whether there exists a full run in  $\llbracket \mathcal{S} \rrbracket$

A run  $(s, x, w) \xrightarrow{*} (s', x', w')$  is **full** when  $\begin{cases} (s, x, w) \text{ is initial} \\ (s', x', w') \text{ is final} \end{cases}$

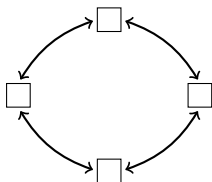
**Goal:**

Characterize the topologies  $\mathcal{T}$  where  $\text{RP-SC1CM}(\mathcal{T})$  is decidable.

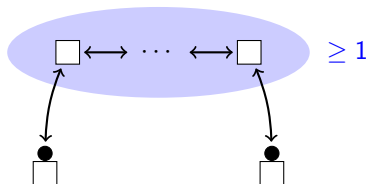
Note:  $\circ$  can be simulated by  $\bullet$

# Main Result

Simple Undirected Cycle



Simple Undirected Shunt



**Theorem:**

$\text{RP-SC1CM}(\mathcal{T})$  is decidable iff  $\mathcal{T}$  is cycle-free and shunt-free

⇔ **cycle-free**: no simple undirected cycle

⇔ **shunt-free**: no simple undirected shunt

# Agenda

⇒ Example: Sliding Window Protocol

⇒ Systems of Communicating One-Counter Machines and their Topology Parametrized Reachability Problem

⇒ Main Theorem: Proof of “Only If” Direction

⇒ Main Theorem: Proof of “If” Direction

⇒ On-going/Future work

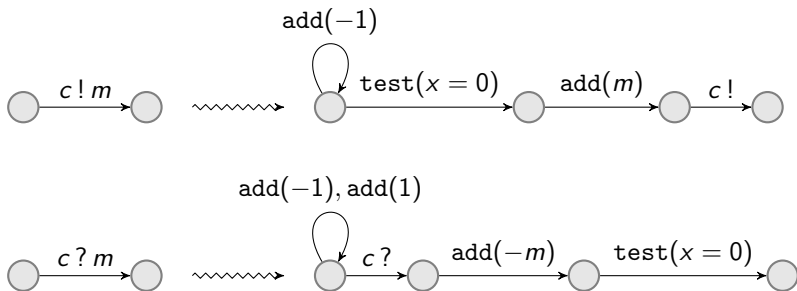


# Cycle-freeness of Decidable Topologies

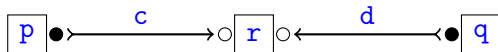
Idea:

Simulation of communicating finite-state machines (CFSM)

Message alphabet:  $M \subseteq \mathbb{N}$  finite



# Shunt-freeness of Decidable Topologies

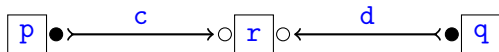


Idea:

Simulation of two-counters Minsky machines

- ⇒ p and q maintain, each, one counter and ignore the other
- ⇒ r checks that they take the same rules of the Minsky machine
- ⇒ p and q need to **send a message  $\delta \in \Delta$  without losing their counter!**

# Shunt-freeness of Decidable Topologies



Idea:

Simulation of two-counters Minsky machines

- ⇒ p and q maintain, each, one counter and ignore the other
- ⇒ r checks that they take the same rules of the Minsky machine
- ⇒ p and q need to **send a message  $\delta \in \Delta$  without losing their counter!**

Idea:

Multiply by  $|\Delta|$  the counters of p and q

- ⇒ p and q send  $x + \delta$
- ⇒ r receives and computes  $\delta = x \bmod |\Delta|$

# Agenda

⇒ Example: Sliding Window Protocol

⇒ Systems of Communicating One-Counter Machines and their Topology Parametrized Reachability Problem

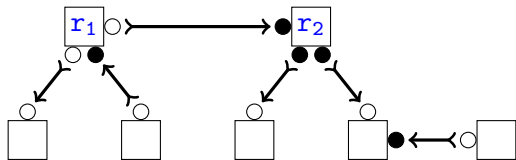
⇒ Main Theorem: Proof of “Only If” Direction

⇒ Main Theorem: Proof of “If” Direction

⇒ On-going/Future work

# Cycle-free & Shunt-free Topologies

Form of topologies that are weakly-connected, cycle-free and shunt-free

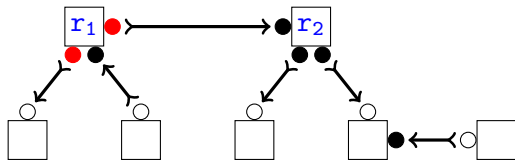


⇒ Two “roots”  $r_1 \rightsquigarrow r_2$

⇒ Every simple undirected path from  $\{r_1, r_2\}$  to  $p \notin \{r_1, r_2\}$  ends with  $\dots \longleftrightarrow \circ p$

# Cycle-free & Shunt-free Topologies

Form of topologies that are weakly-connected, cycle-free and shunt-free

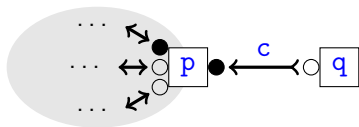


⇒ Two “roots”  $r_1 \bullet \rightarrow \bullet r_2$

⇒ Every simple undirected path from  $\{r_1, r_2\}$  to  $p \notin \{r_1, r_2\}$  ends with  $\dots \bullet \leftrightarrow \circ p$

[ Recall:  $\circ$  can be simulated by  $\bullet$  ]

# Merging Leaf Processes



**Idea:**

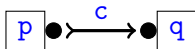
Merge leaf process  $q$  into  $p$  by summarizing  $q$ 's behavior

- ⇒ Schedule  $q$  last :  $q$  moves only when  $p$  attempts to receive from  $c$
- ⇒ Communications between  $p$  and  $q$  become synchronizations  $c! \cdot c?$
- ⇒ States of  $p$  become pairs  $(s^p, s^q)$
- ⇒ Rules  $(s^p, c?, t^p)$  of  $p$  become  $((s^p, s^q), \text{test}(\varphi), (t^p, t^q))$  where

$$\varphi = \exists u \exists z \cdot (u, c!, t^q) \in \Delta^q \wedge (s^q, z) \xrightarrow{*}_q (u, x)$$

- ⇒ Use Presburger-definability of  $post^*$  for one-counter machines

# Case of Two Processes



Idea:

Intersect reachability relations of  $p$  and  $q$  between synchronizations

$$\dots \xrightarrow{c!c?} \underbrace{(s, x) \xrightarrow{*} (t, y)}_{\chi_{s,t}} \xrightarrow{c!c?} \dots$$

$$\chi_{s,t}(x, y) = (s^p, x) \xrightarrow{*}_p (t^p, y) \wedge (s^q, x) \xrightarrow{*}_q (t^q, y) \in \mathcal{P}_2$$

Issue:

Reachability is **undecidable** for the class of one-counter machines with Presburger-definable updates



# One-Counter Reachability Relations

Fix two distinguished Presburger variables  $x$  and  $y$

The class of **one-counter Presburger predicates** is generated by

$$\psi ::= \varphi(x) \mid \varphi(y) \mid \varphi(x - y) \mid \varphi(y - x) \mid \psi \wedge \psi \mid \psi \vee \psi \mid \mathbf{tt} \mid \mathbf{ff}$$

where  $\varphi$  ranges over unary Presburger predicates

**Theorem:**

For every binary relation  $R \subseteq \mathbb{N} \times \mathbb{N}$ , the two following assertions are equivalent:

- i)  $R = \{(x, y) \mid (s, x) \xrightarrow{*} (t, y)\}$  for some one-counter machine
- ii)  $R = \llbracket \psi \rrbracket$  for some one-counter Presburger predicate  $\psi$

$\Leftrightarrow \chi_{s,t}(x, y)$  can be translated into a one-counter machine

# Agenda

⇒ Example: Sliding Window Protocol

⇒ Systems of Communicating One-Counter Machines and their Topology Parametrized Reachability Problem

⇒ Main Theorem: Proof of “Only If” Direction

⇒ Main Theorem: Proof of “If” Direction

⇒ On-going/Future work

# Decidability of Eager Reachability

**Definition:**

A full run  $\rho$  is **eager** if matching  $(c!, c?)$  pairs are consecutive in  $\rho$

If  $\mathcal{T}$  is cycle-free,

⇒ Full runs can be re-ordered into eager ones

⇒  $\text{RP-SC1CM-EAGER}(\mathcal{T})$  is decidable iff  $\mathcal{T}$  is shunt-free

**Proposition:**

If  $\mathcal{T}$  is strongly connected, then  $\text{RP-SC1CM-EAGER}(\mathcal{T})$  is decidable iff  $\mathcal{T}$  contains at most two processes

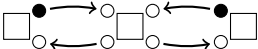
**Open:** full characterization of decidable topologies (for eager reachability)

# Perspectives

Complexity of  $RP\text{-}SC1CM$  for decidable topologies

⇒ At least PSPACE-hard

Lossy channels

⇒ Undecidable for  using acknowledgments

The diagram shows two nodes, each represented by a square with two circles below it. The left node has a black dot on its top circle. The right node has a black dot on its top circle. There are two channels between the nodes: one connecting the top circles and one connecting the bottom circles. Arrows indicate bidirectional communication between the top circles and between the bottom circles.

Extension from counters to stacks

⇒ Conjecture: same characterization (cycle-free and shunt-free)