A Unifying Framework for Deciding Synchronizability

B. Bollig, C. Di Giusto, A. Finkel, L. Laversa, É. Lozes, and **A. Suresh**

I3S, Univ. Côte d'Azur and LMF, ENS Paris Saclay

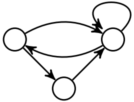
CONCUR 2021

FIFO Systems

Distributed processes

- each process is a finite state machine

P_i :

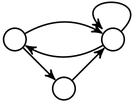


FIFO Systems

Distributed processes

- each process is a finite state machine
- fixed number

P_i :



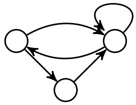
$P_1, \ldots, P_n$

FIFO Systems

Distributed processes

- each process is a finite state machine

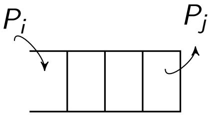
P_i :



- fixed number

$P_1, \dots, P_n$

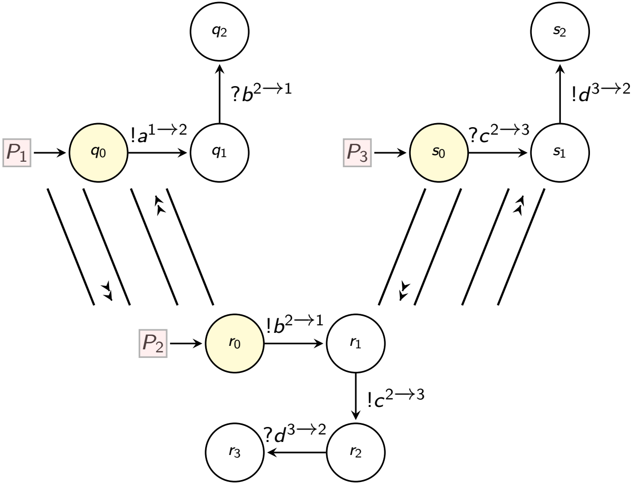
- communicate using queues



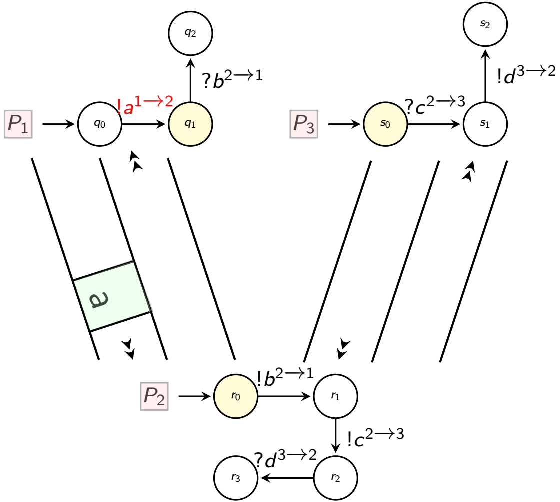
Communication architecture

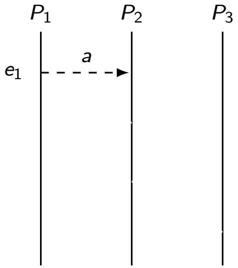
- p2p $\rightarrow$ one queue per pair of processes
- mailbox $\rightarrow$ one queue per process

Example: a P2P System



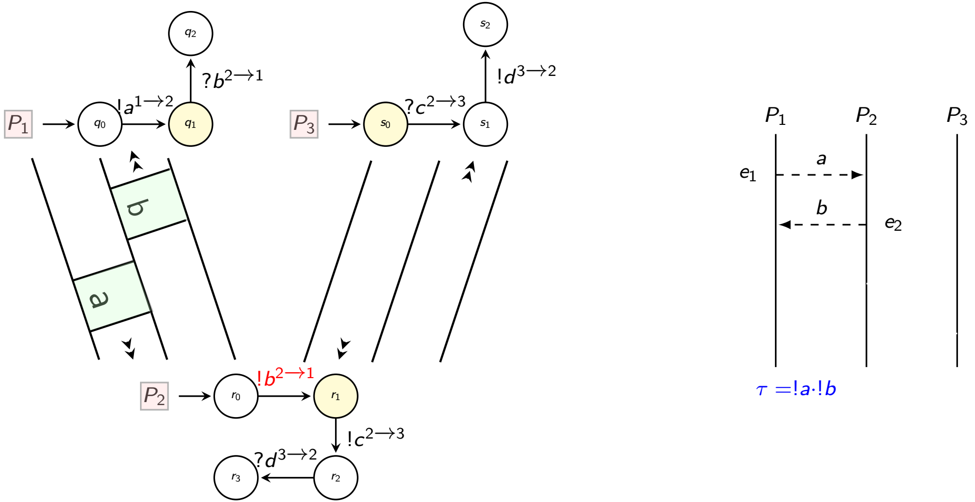
Example: a P2P System





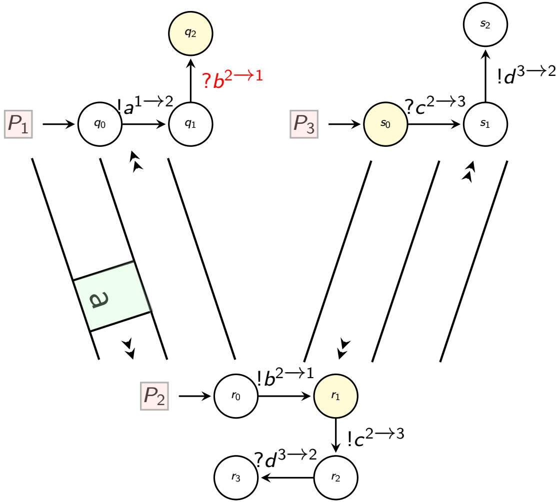
$\tau = !a$

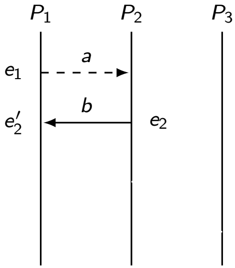
Example: a P2P System



$\tau = !a \cdot !b$

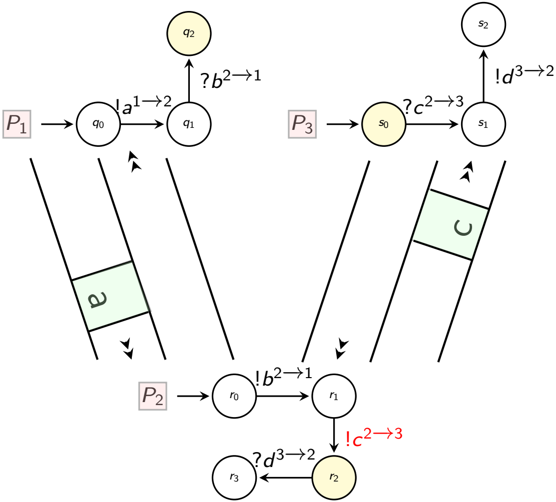
Example: a P2P System

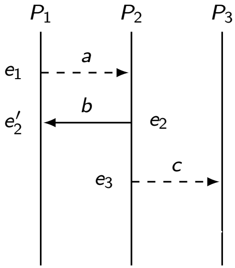




$\tau = !a \cdot !b \cdot ?b$

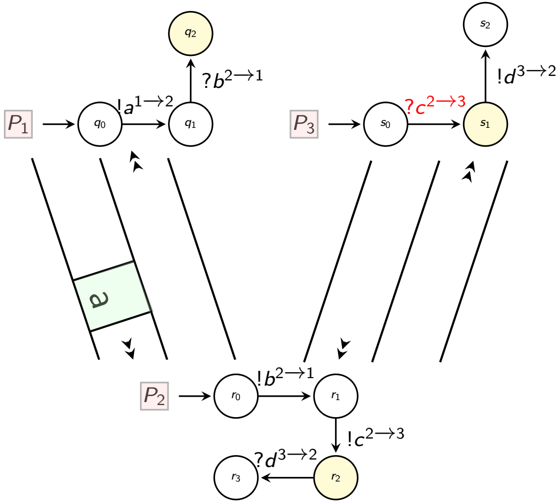
Example: a P2P System

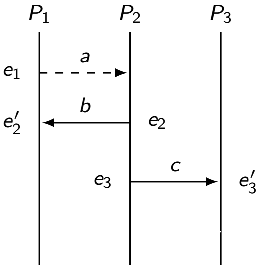




$\tau = !a \cdot !b \cdot ?b \cdot !c$

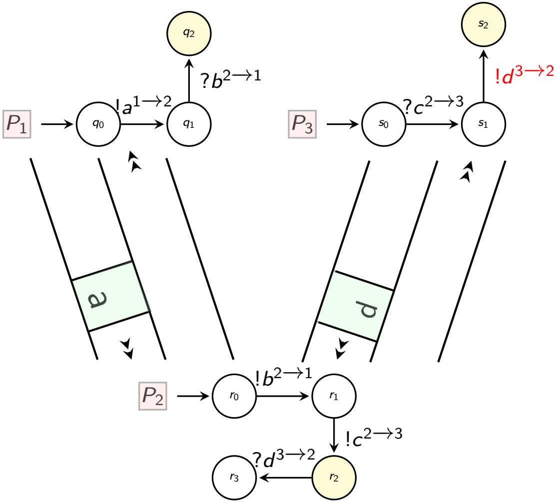
Example: a P2P System

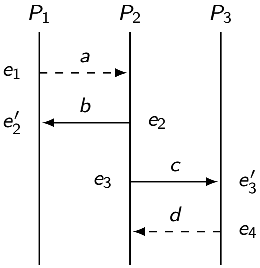




$\tau = !a \cdot !b \cdot ?b \cdot !c \cdot ?c$

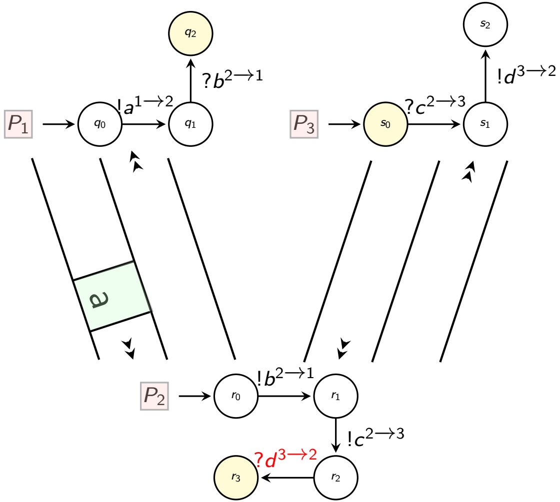
Example: a P2P System

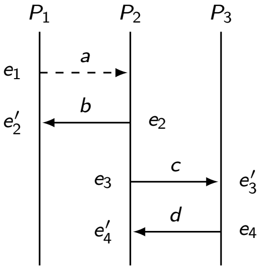




$\tau = !a \cdot !b \cdot ?b \cdot !c \cdot ?c \cdot !d$

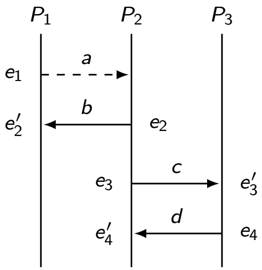
Example: a P2P System





$\tau = !a \cdot !b \cdot ?b \cdot !c \cdot ?c \cdot !d \cdot ?d$

Example: a Mailbox system



- We cannot have same trace as before!
- MSC still valid

New trace $\tau = !b \cdot !c \cdot ?c \cdot !d \cdot !a \cdot ?b \cdot ?d$

Boundedness

Boundedness Problem

Is there a bound on the size of the queues for all runs?

Boundedness

Boundedness Problem

Is there a bound on the size of the queues for all runs?

UNDECIDABLE in general FIFO systems [1]

[1] Brand and Zafiropulo, *On communicating finite-state machines*, 1983

Boundedness

Underapproximations

- Restrict to k-bounded channels.

Boundedness

Underapproximations

- Restrict to k-bounded channels. Too restricting!

Boundedness

Underapproximations

- Restrict to k-bounded channels. Too restricting!
- Every unbounded execution is equivalent to a bounded execution.

Synchronizability

- *existentially k-bounded* systems [1] [2] - all accepting executions re-ordered to a k-bounded execution.

[1] Lohrey and Muscholl, *Bounded MSC communication*, 2002

[2] Genest et al., *A Kleene theorem for a class of communicating automata with effective algorithms*, 2004

Synchronizability

- *existentially k-bounded* systems [1] [2]
- *synchronizable systems* [3] - send projection equivalent to rendezvous.

[1] Lohrey and Muscholl, *Bounded MSC communication*, 2002
[2] Genest et al., *A Kleene theorem for a class of communicating automata with effective algorithms*, 2004
[3] Basu and Bultan, *Choreography conformance via synchronizability*, 2011

Synchronizability

- *existentially k-bounded* systems [1] [2]
- *synchronizable systems* [3]
- *k-synchronizable systems* [4] - if every MSC admits a linearization that can be divided into "blocks" of at most k messages.

[1] Lohrey and Muscholl, *Bounded MSC communication*, 2002
[2] Genest et al., *A Kleene theorem for a class of communicating automata with effective algorithms*, 2004
[3] Basu and Bultan, *Choreography conformance via synchronizability*, 2011
[4] Bouajjani et al., *On the completeness of verifying message passing programs under bounded asynchrony*, 2018

Weakly k-synchronous MSCs

A k-exchange is an MSC that allows one to schedule all sends before all receives, and there are at most k sends.

Weakly k-synchronous MSCs

A k-exchange is an MSC that allows one to schedule all sends before all receives, and there are at most k sends.

Definition

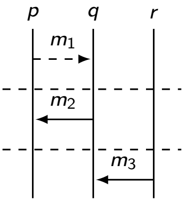
M is weakly k-synchronous if it is of the form $M = M_1 \cdot \ldots \cdot M_n$ such that every M_i is a k-exchange.

Weakly k-synchronous MSCs

A k-exchange is an MSC that allows one to schedule all sends before all receives, and there are at most k sends.

Definition

M is weakly k-synchronous if it is of the form $M = M_1 \cdot \ldots \cdot M_n$ such that every M_i is a k-exchange.



Weakly k-synchronous MSCs

An exchange is an MSC that allows one to schedule all sends before all receives ~~and there are at most k sends~~.

Definition

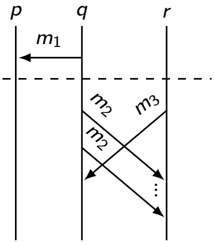
M is weakly synchronous if it is of the form $M = M_1 \cdot \ldots \cdot M_n$ such that every M_i is an exchange.

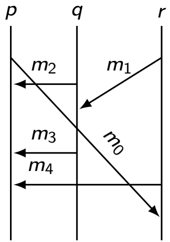
Weakly k-synchronous MSCs

An exchange is an MSC that allows one to schedule all sends before all receives ~~and there are at most k sends~~.

Definition

M is weakly synchronous if it is of the form $M = M_1 \cdot \ldots \cdot M_n$ such that every M_i is an exchange.





MSO definability

Condition 1

The set of MSCs are MSO-definable.

MSO definability

First-order variables

MSO Logic

- $x \to y$ x precedes y in the process order
- $x \lhd y$ x and y are matched send-receive events
- $\lambda(x) = a$ x has the label a
- $x = y$ Second-order variable
- $\exists x.\phi$ there is an event x such that ϕ
- $\exists X.\phi$ there is a unary relation X such that ϕ holds
- $\phi \vee \phi$, $\neg\phi$, $x \in X$, etc.

MSO definability

First-order variables

MSO Logic

- $x \rightarrow y$ x precedes y in the process order
- $x \lhd y$ x and y are matched send-receive events
- $\lambda(x) = a$ x has the label a
- $x = y$
- $\exists x.\phi$ there is an event x such that ϕ
- $\exists X.\phi$ there is a unary relation X such that ϕ holds
- $\phi \vee \phi$, $\neg\phi$, $x \in X$, etc.

Second-order variable

$matched(x) = \exists y.x \lhd y$ indicates that x is a matched send.

Special tree width

Condition 2

The set of MSCs have bounded special tree-width.

Special tree width

Condition 2

The set of MSCs have bounded special tree-width.

Special tree width

Condition 2

The set of MSCs have bounded special tree-width.

- Adam-Eve play the *decomposition game*.

Special tree width

Condition 2

The set of MSCs have bounded special tree-width.

- Adam-Eve play the *decomposition game*.
- Eve "colours" some events on the MSC, removes edges between coloured events.

Special tree width

Condition 2

The set of MSCs have bounded special tree-width.

- Adam-Eve play the *decomposition game*.
- Eve "colours" some events on the MSC, removes edges between coloured events.
- Adam chooses one of the resulting connected components.

Special tree width

Condition 2

The set of MSCs have bounded special tree-width.

- Adam-Eve play the *decomposition game*.
- Eve "colours" some events on the MSC, removes edges between coloured events.
- Adam chooses one of the resulting connected components.
- Bounded special tree-width k if Eve can win (colour all vertices) with $k + 1$ colours.

Crucial observation

Theorem

Let $\mathcal{C}$ be a class of MSCs. If $\mathcal{C}$ is MSO-definable and STW-bounded class, the following problem is decidable: Given a communicating system S, do we have $L(S) \subseteq \mathcal{C}$?

Crucial observation

Theorem

Let $\mathcal{C}$ be a class of MSCs. If $\mathcal{C}$ is MSO-definable and STW-bounded class, the following problem is decidable: Given a communicating system S, do we have $L(S) \subseteq \mathcal{C}$?

- Synchronizability for an STW-bounded class $\xrightarrow{\text{reduces to}}$ *bounded model-checking*
- Bounded model-checking $\rightarrow$ known to be decidable [5]

[5] c.f. Bollig and Gastin, *Non-sequential theory of distributed systems*, 2019

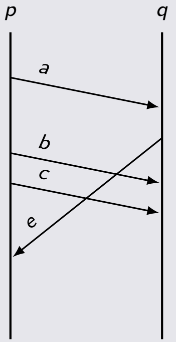
Applying the framework to Weakly synchronous MSCs

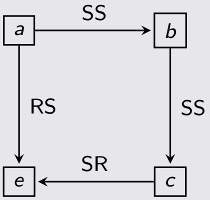
Result

The set of weakly synchronous MSCs are MSO-definable.

Applying the framework to Weakly synchronous MSCs

Conflict graph





Applying the framework to Weakly synchronous MSCs

Result

The set of weakly synchronous MSCs are MSO-definable.

Graphical characterization of weakly synchronous MSCs

- No RS edge along any cycle

MSO definable!

Applying the framework to Weakly synchronous MSCs

Result

The set of weakly synchronous MSCs has bounded STW.

- Eve's strategy - isolate each exchange, then remove message pairs
- Uses at most $4n + 1$ colours

Summary of results

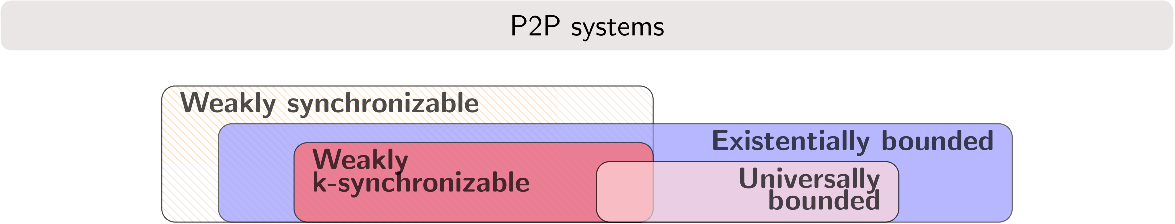
Class of MSCs	Peer-to-Peer	Mailbox
Weakly synchronous	Undecidable	EXPTIME
Weakly k-synchronous	Decidable [6], [7]	
Strongly k-synchronous	—	Decidable
Existentially k-p2p-bounded	Decidable [8]	
Existentially k-mailbox-bounded	—	Decidable

[6] Bouajjani et al., *On the completeness of verifying message passing programs under bounded asynchrony*, 2018

[7] Di Giusto et al., *On the k-synchronizability of systems*, 2020

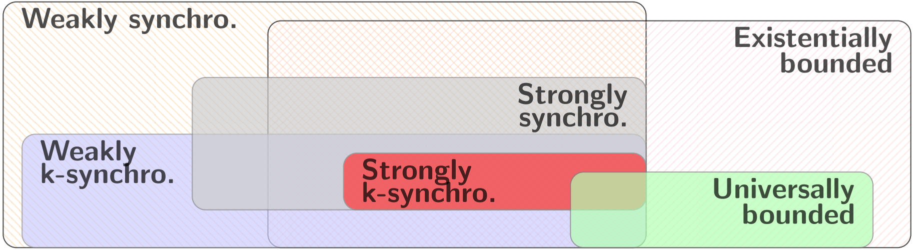
[8] Genest et al., *On communicating automata with bounded channels*, 2007

Comparison of classes



Comparison of classes

Mailbox systems



Contributions

- Unifying framework for various notions of synchronizability.
- Applicable to both mailbox and p2p communications.
- LCPDL for better complexity.

Thank you!