A Formal Model of Asynchronous Broadcast Communication

Giorgio Delzanno Riccardo Traverso

DIBRIS, Università di Genova, Italy

Varese, September 21th, 2012

Table of Contents

Table of Contents

Ad Hoc Networks: Main features

- No fixed infrastructure.
- Selective broadcast.
- Asynchronous communication.
- ...

Verification Approaches

- **Process Algebra** for timed and untimed broadcast communication.
  [Prasad, Mezzetti-Sangiorgi, Ene-Muntean, Fehnker et al., . . .]
- **Model Checking** for fixed initial configurations.
  [Fehnker-Van Hoesel-Mader, . . .]
- **Static Analysis** for a fixed set of connection graphs.
  [Nanz-Hankin, Nanz-Nielson-Nielson]
- **Constraint-based Analysis** for arbitrary graphs of a given size.
  [Singh-Ramakrishnan-Smolka]
- **Parameterized Verification** for selective (synchronous) broadcast models.
  [Delzanno et al.]

Our Goal

Explore decidability boundaries for **parameterized verification** of **asynchronous** broadcast models:

- semantics for **incoming messages**;
- **shape** of connection graph;
- **instruction set** to model protocols.

Table of Contents

Asynchronous Broadcast Networks (ABN)

- A network of finite-state automata **distributed on a graph**.
- Topology-dependent semantics of synchronization.
- Nodes communicate via **asynchronous** broadcast messages only.
- Unread messages are kept in **local mailboxes**.
- We consider different **disciplines** for handling mailboxes (e.g. bags, FIFO queues).

Definition

A **mailbox structure** is a tuple $\mathbb{M} = \langle \mathcal{M}, del?, add, del, [] \rangle$ where:

- $\mathcal{M}$ is the set of all possible mailbox contents on some fixed finite alphabet Σ;
- for $a \in \Sigma$ and $m \in \mathcal{M}$, $del?(a, m)$, $add(a, m)$, and $del(a, m)$ are operations over mailboxes;
- $[] \in \mathcal{M}$ is the empty mailbox.

Visibility

$a \in \Sigma$ is said to be **visible** in $m \in \mathcal{M}$ when $del?(a, m)$ is true.

ABN: Mailbox

We will consider three mailbox structures:

- **bags**, to model the loss of the order of incoming messages.
- **lossy FIFO queues**, to model the loss of messages;
- **FIFO queues**, to model perfect communication;

ABN: Protocol

Definition

A **protocol** is defined by a process $\mathcal{P} = \langle Q, \Sigma, R, q_0 \rangle$, where:

- Q is a finite set of control states;
- Σ is a finite message alphabet;
- $Act = \{\tau\} \cup \{!!a, ??a \mid a \in \Sigma\}$;
- $R \subseteq Q \times Act \times Q$ is the transition relation;
- $q_0 \in Q$ is an initial control state.

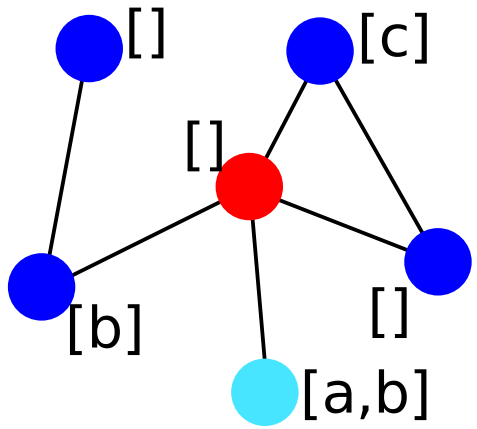
ABN: Configurations

Definition

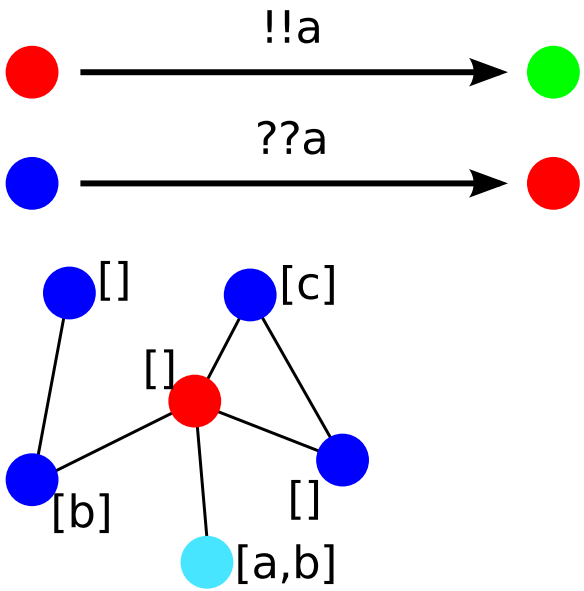
A **configuration** is an undirected graph with labels in $Q \times \mathcal{M}$.

Definition

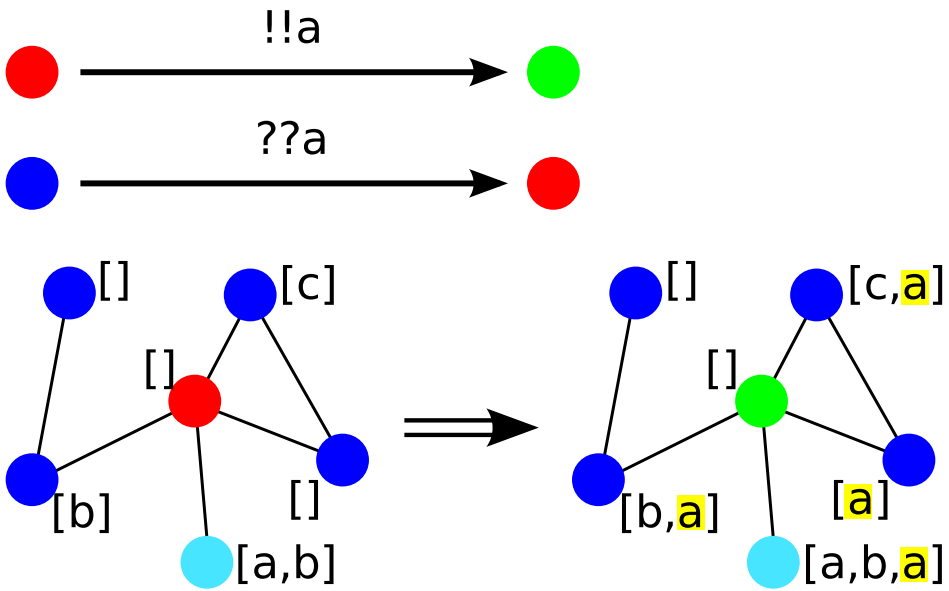
A **configuration** is an undirected graph with labels in $Q \times \mathcal{M}$.



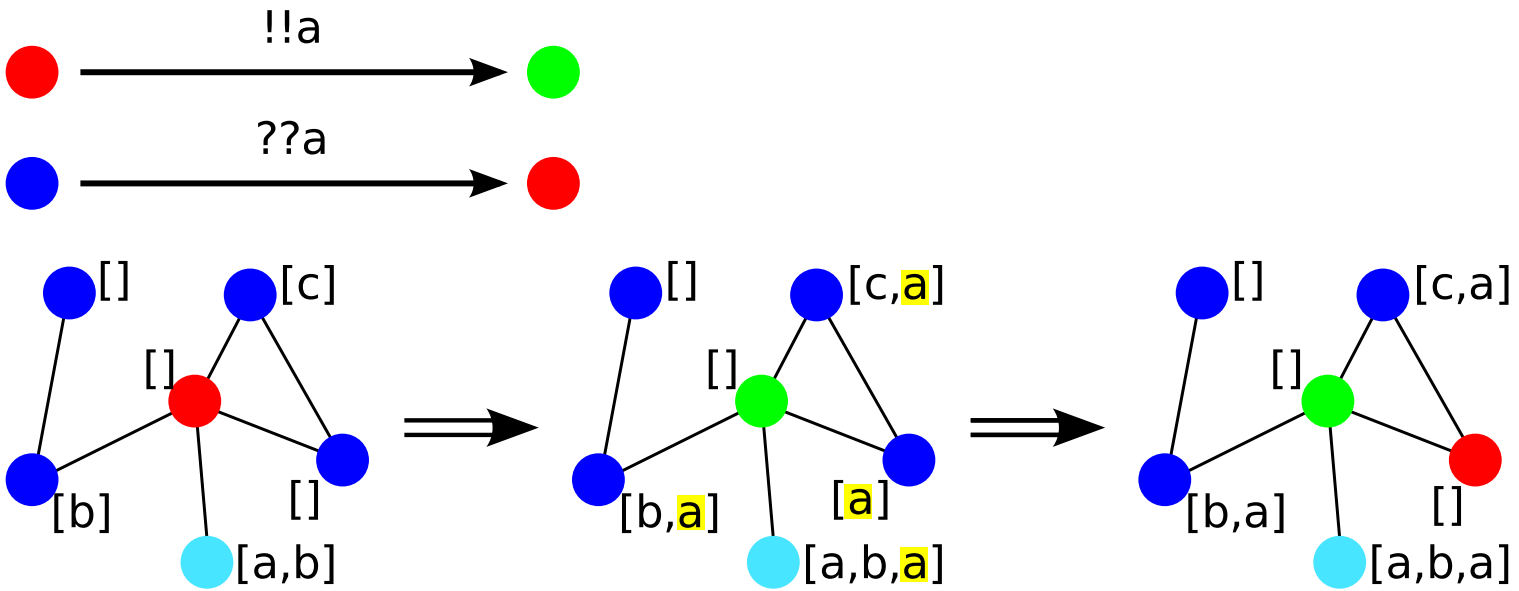
ABN: Example (with bags)



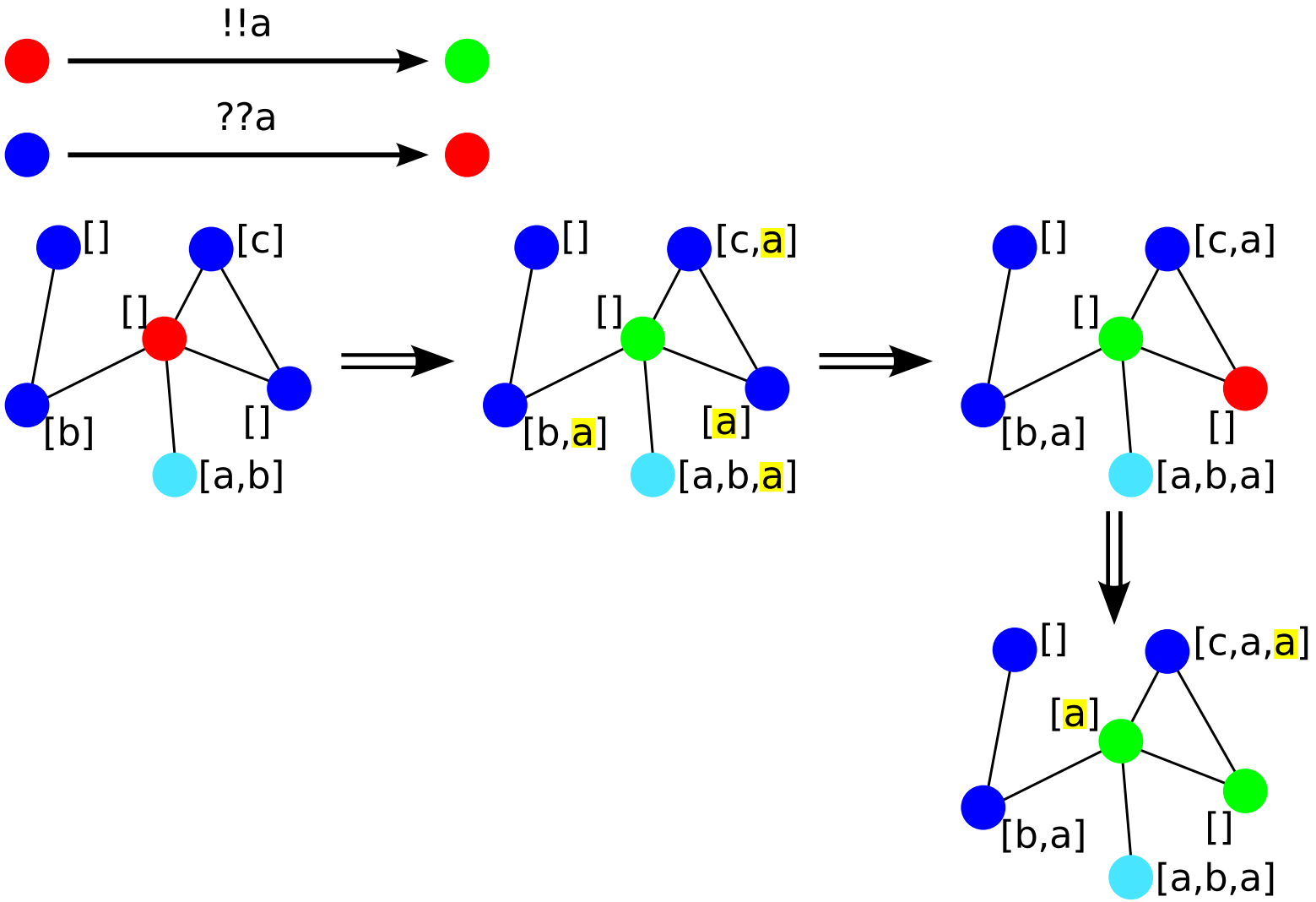
ABN: Example (with bags)



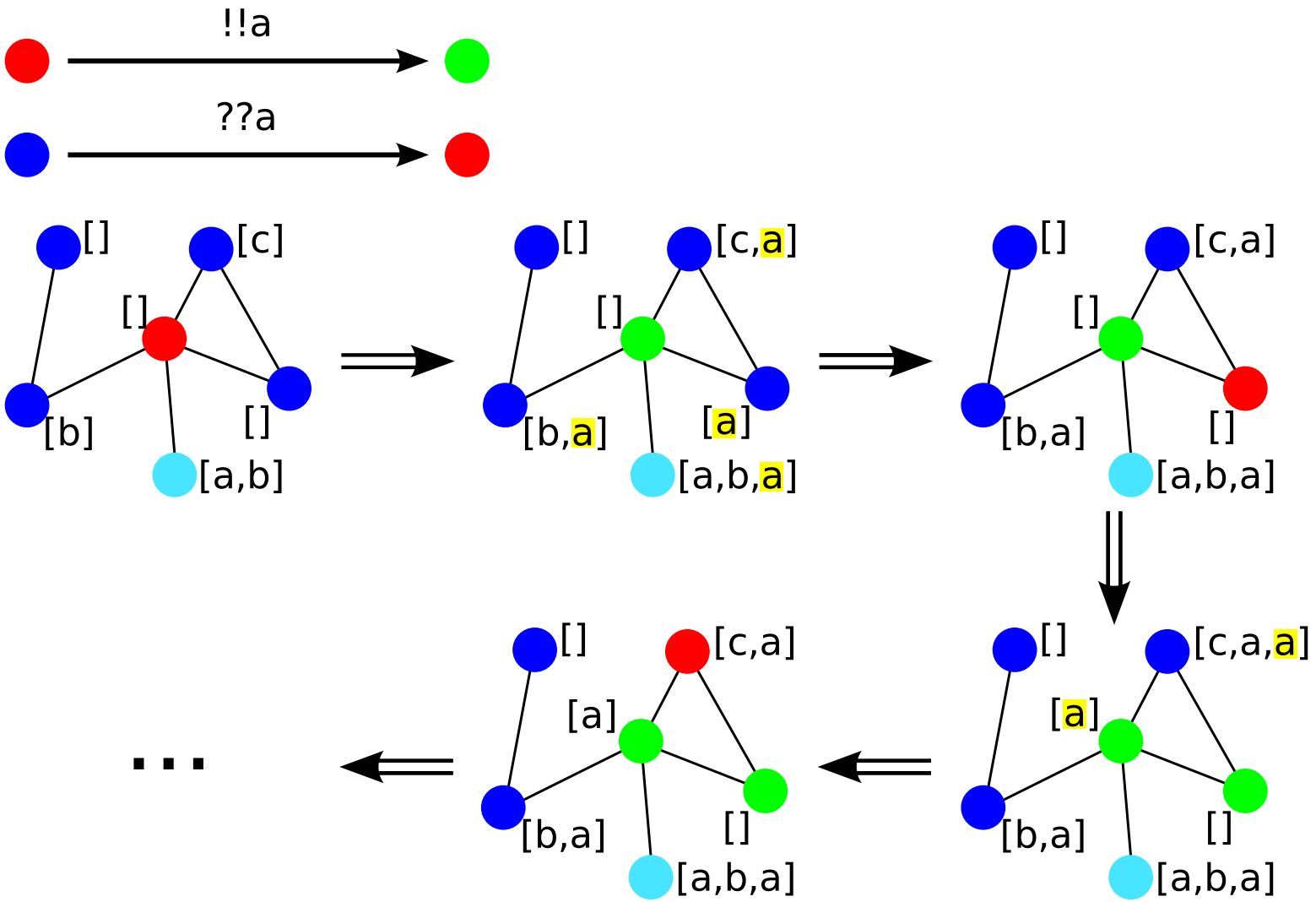
ABN: Example (with bags)



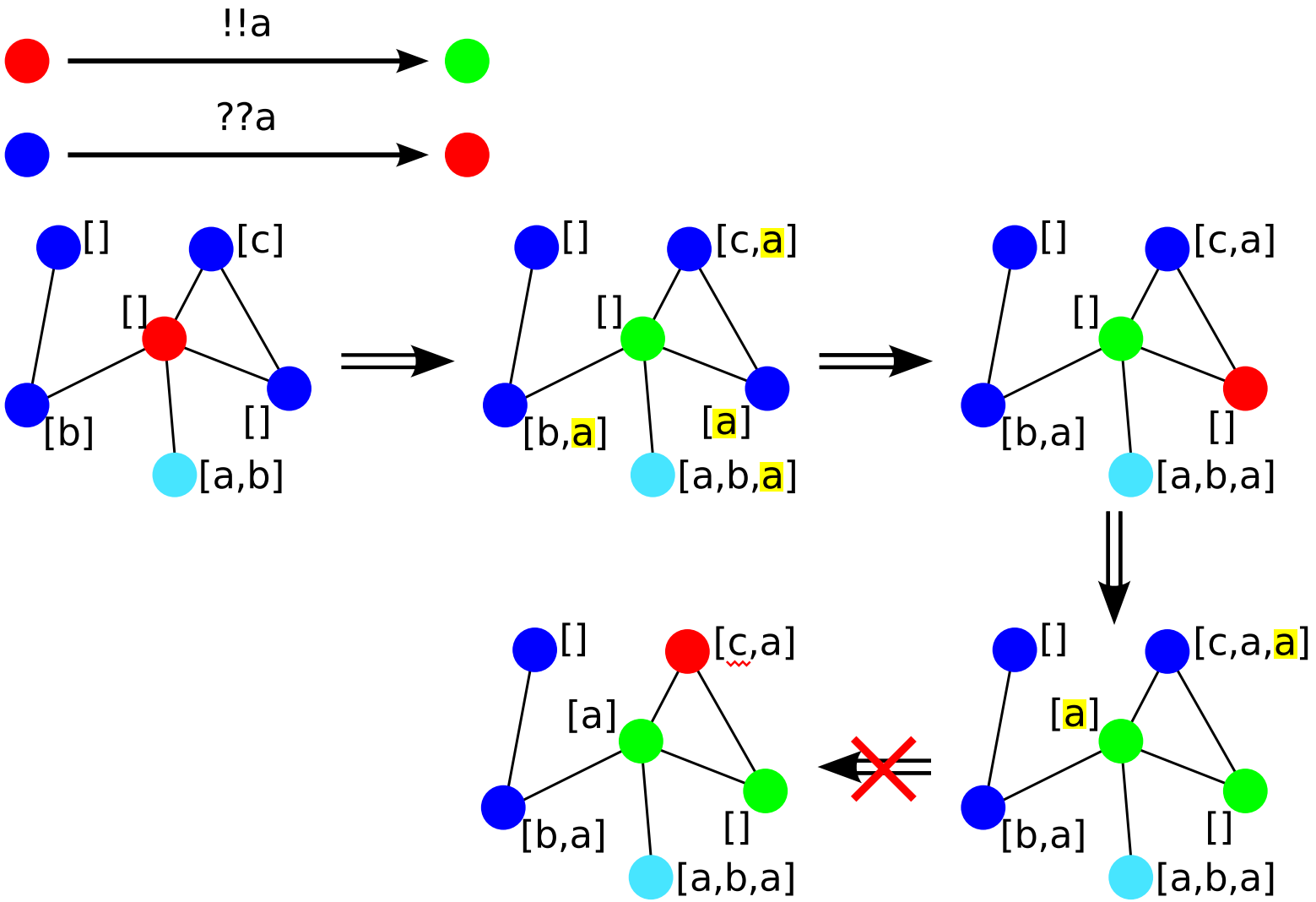
ABN: Example (with bags)



ABN: Example (with bags)



ABN: Example (with FIFO queues)



ABN: Example (with lossy FIFO queues)

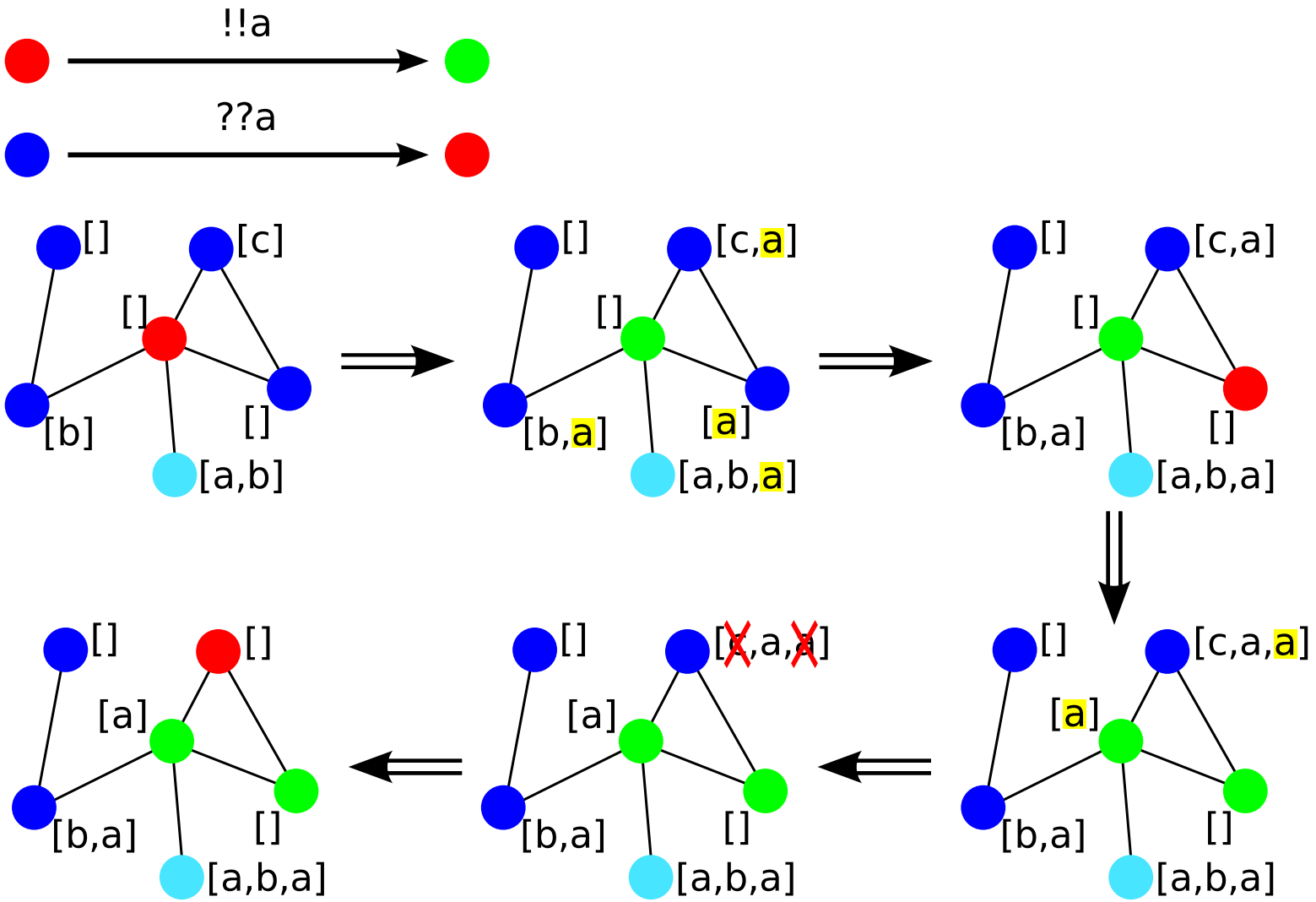
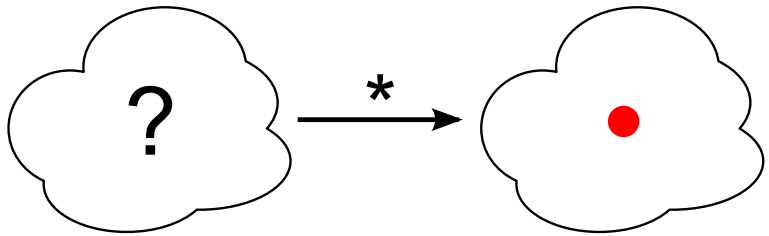


Table of Contents

COVER($\mathbb{M}$): Problem Definition

Given a protocol $\mathcal{P}$, a mailbox structure $\mathbb{M}$ and a control state q, the coverability problem *COVER*($\mathbb{M}$) states: is there an initial configuration of the resulting ABN such that it may evolve into a configuration exposing the state q?



$COVER^{\mathcal{K}}(\mathbb{M})$: Fully Connected Graphs

We use $COVER^{\mathcal{K}}(\mathbb{M})$ to denote the restriction of $COVER(\mathbb{M})$ to fully connected configurations only.

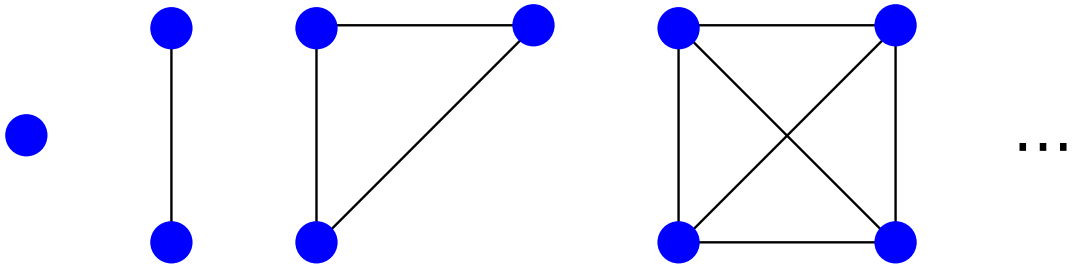


Table of Contents

ABN: $COVER^{\mathcal{K}}(Bag)$

Bags

The mailbox structure *Bag* is instantiated in such a way that the elements of $\mathcal{M}$ are multisets over Σ.

$COVER^{\mathcal{K}}(Bag)$ is decidable

Proof idea: we can resort to the theory of well-structured transition systems (wsts) to build an algorithm that solves the problem.

ABN: $COVER^{\mathcal{K}}(Bag)$

WSTS

- Symbolic configurations are multisets of pairs formed by a state in Q and a multiset over Σ.
- The well-quasi order is defined by exploiting identity over Q and dominance between multisets.
- The target set is upward closed w.r.t. $\preceq$.
- We prove monotonicity for the transition relation and define an algorithm to compute the predecessors.
- Backward reachability algorithm to solve $COVER^{\mathcal{K}}(Bag)$.

ABN: *COVER*(*Bag*)

$COVER(Bag)$ is decidable

Coverability for arbitrary topologies can be reduced to the fully connected case.

Proof idea: Given an arbitrary configuration, we can simulate it with a fully connected one by ignoring all extra messages.

ABN: $COVER^{\mathcal{K}}(LFIFO)$

Lossy FIFO queues

The mailbox structure *LFIFO* handles incoming messages with FIFO queues that may lose data arbitrarily.

$COVER^{\mathcal{K}}(LFIFO)$ is decidable

As in the *Bag* case, we can resort to the theory of wsts.

ABN: $COVER^{\mathcal{K}}(LFIFO)$

Lossy FIFO queues

The mailbox structure *LFIFO* handles incoming messages with FIFO queues that may lose data arbitrarily.

$COVER^{\mathcal{K}}(LFIFO)$ is decidable

As in the *Bag* case, we can resort to the theory of wsts.

WSTS

- Symbolic configurations are multisets of pairs in $Q \times \Sigma^*$.
- The well-quasi order $\preceq$ over configurations is defined by exploiting string embedding over mailboxes.

ABN: *COVER*(*LFIFO*)

$COVER(LFIFO)$ is decidable

There is a reduction from $COVER(LFIFO)$ to $COVER^{\mathcal{K}}(LFIFO)$.
Proof idea: instead of ignoring extra messages, we can assume they are going to be deleted by lossy steps.

ABN: $COVER^{\mathcal{K}}(FIFO)$ and $COVER(FIFO)$

FIFO queues

The mailbox structure *FIFO* makes local mailboxes behave like (perfect) FIFO queues.

$COVER^{\mathcal{K}}(FIFO)$ and $COVER(FIFO)$ are undecidable

We can build a reduction from the halting problem for two-counter machines. The same construction works for both problems: it does not assume anything about the underlying topology.

ABN with Emptiness Test (ABN_ϵ)

We enrich the ABN model in order to enable individual processes to check whether the local mailbox is empty.

ABN_ϵ Model

The set of actions *Act* is extended to $\{\tau, \epsilon\} \cup \{!!a, ??a \mid a \in \Sigma\}$, and the semantics is modified accordingly, i.e. such that a ϵ-transitions can be fired if and only if the local mailbox is empty.

ABN_ϵ: FIFO and LFIFO

$COVER^{\mathcal{K}}(FIFO)$ and $COVER(FIFO)$ are undecidable

The possibility to test the emptiness of the mailbox does not affect the reduction from two-counter machines.

$COVER^{\mathcal{K}}(LFIFO)$ and $COVER(LFIFO)$ are decidable

The wsts built for the ABN case still works with ABN_ϵ models.
Proof idea: ϵ-transitions are somewhat like internal transitions, because we can always empty the local mailbox to pass the emptiness test.

ABN_ϵ: $COVER^{\mathcal{K}}(Bag)$ and $COVER(Bag)$

$COVER^{\mathcal{K}}(Bag)$ and $COVER(Bag)$ are undecidable

The transition system is not well-structured anymore, because ϵ-transitions add enough expressive power to the model to reach Turing-completeness (we cannot arbitrarily delete incoming messages anymore with bags).

Proof idea: reduction from halting problem for two-counter machines. The emptiness test can be used both for zero-testing and interference detection.

ABN_ϵ: From two-counter machines to *COVER*(*Bag*)

Two-counter machines

A **two-counter machine** is defined by a set of *control locations*, a set of *instructions* $Inst \subseteq Loc \times Op \times Loc$ over two natual counters (increment, decrement, zero-test), and an *initial location*.

- The encoded protocol is split in two phases: **election** and **simulation**.
- During the election processes choose their role.
- The simulation requires a leader process directly connected to two slaves (one per counter).

ABN_ϵ: From two-counter machines to *COVER*(*Bag*)

Election

- Each node chooses a role and searches for appropriate neighbors accordingly.
- Election only tests for the presence of the required links between nodes with the various roles.
- Messages exchanged during the election can never be consumed afterwards.
- A successful election ends leaving empty mailboxes in the involved nodes.

ABN_ϵ: From two-counter machines to *COVER*(*Bag*)

Simulation

- Counters are encoded (in unary) through messages in the mailboxes.
  - For increment it is sufficient to send a broadcast with a unit.
  - A decrement forces the removal of a unit from the mailbox of the slave.
  - Tests for zero are performed by exploiting ϵ-transitions.

ABN_ϵ: From two-counter machines to *COVER*(*Bag*)

Warning

What if other neighbors wake up and start a simulation leading to interferences with the current one?

ABN_ϵ: From two-counter machines to *COVER*(*Bag*)

Warning

What if other neighbors wake up and start a simulation leading to interferences with the current one?

Finalization

- Reminder: we cannot consume messages from the election during the simulation.
- After reaching the target control state, we reset both counters to zero in order to try to empty all mailboxes.
- If all mailboxes are empty the simulation ends successfully, otherwise it blocks just before completing.

Table of Contents

Summary of our results

	$COVER^{\mathcal{K}}(\mathbb{M})$		$COVER(\mathbb{M})$	
	ABN	ABN_ϵ	ABN	ABN_ϵ
LFIFO	✓	✓	✓	✓
Bag	✓	undec.	✓	undec.
FIFO	undec.	undec.	undec.	undec.

Comparison w.r.t. (synchronous) Ad Hoc Networks

		ABN / ABN_ϵ		
	AHN[1]	LFIFO	Bag	FIFO
Fully connected graphs	✓	✓	✓/undec.	undec.
Arbitrary graphs	undec.	✓	✓/undec.	undec.

[1] Parameterized Verification of Ad Hoc Networks. Delzanno, Sangnier, Zavattaro.
On the Power of Cliques in the Parameterized Verification of Ad Hoc Networks. Delzanno, Sangnier, Zavattaro.

Future work

- Complexity of decidable cases *(work in progress)*.
  - They are all decidable in PTime.
  - We still lack proofs for PTime-hardness.
- Extend the model with a notion of process identifiers.
  - Coverability may be decidable under some restrictions on the operations.
- Study a variant with asynchronous rendez-vous communication.

Thank you for your attention!