Verification of Randomized Distributed Algorithms under Round-Rigid Adversaries

Josef Widder
TU Wien

Nathalie Bertrand  Igor Konnov  Marijana Lazić

MeTRiD, April 6, 2019
Fault-tolerant distributed algorithms

$n$ processes communicate by sending messages **asynchronously**

$f$ processes are faulty (unknown)

$t$ is an upper bound on $f$ (known)

**resilience condition** on $n$, $t$, and $f$, e.g., $n > 3t \land t \geq f \geq 0$
Parameterized Verification

∀n, t, f with n > 3t and t ≥ f ≥ 0.

\[
P(n, t) \parallel P(n, t) \parallel \ldots \parallel P(n, t) \parallel \text{Byz} \parallel \ldots \parallel \text{Byz} \]

\[\vdash \text{Safety} \land \text{Liveness}\]
Previous work

Verification of non-randomized distributed algorithms

(Konnov, Lazić, Veith, W, POPL 2017)
Threshold automata

\[ x_0 \geq \frac{n + t}{2} - f \iff y_0^{++} \]

\[ x_1 \geq \frac{n + t}{2} - f \iff y_1^{++} \]

send \(<x_1>\) to all

if received \(<x_1>\) from at least \(\frac{n + t}{2} - f\) distinct processes
then send \(<y_1>\) to all
Counter System as a Semantic of a TA

\[ x_0 \geq \frac{(n + t)}{2} - f \iff y_0 ++ \]

\[ x_1 \geq \frac{(n + t)}{2} - f \iff y_1 ++ \]

\[ \kappa[I_1] = 1 \]

\[ \kappa[E_1] = 2 \]

\[ \kappa[D_1] = 1 \]

count how many processes are in every location

one process decided 1
Specifications in LTL\(_{-X}\) with counters

**Agreement**: No two correct processes decide differently  

\[
F \kappa[D_v] > 0 \implies G \kappa[D_{1-v}] = 0
\]

**Termination**: Eventually all correct processes decide  

\[
F \bigwedge_{\ell \in \mathcal{L} \setminus \{D_0, D_1\}} \kappa[\ell] = 0
\]

We denote this fragment by \(\text{ELTL}_{\text{FT}}\)
Verification of Distributed Algorithms

Does $\text{Sys}(\text{TA}) \models \varphi$? (Konnov, Lazić, Veith, W, POPL’17)

Given a threshold automaton $\text{TA}$, a specification $\varphi$ in $\text{ELTL}_{\text{FT}}$, and a resilience condition $RC$, we can check whether for all parameters satisfying $RC$ holds that

$$\text{Sys}(\text{TA}) \models \varphi$$

[ [forsyte.at/software/bymc] ]
Limits of Application Domain

FLP85: There is no asynchronous consensus algorithm: Impossibility due to combination of

- faulty processes
- asynchrony
- safety requirements
- liveness requirement

Ben-Or’s randomized algorithm achieves:

- faulty processes
- asynchrony
- safety requirements
- almost sure termination

⇒ randomized distributed algorithms
Limits of Application Domain

FLP85: There is no asynchronous consensus algorithm: Impossibility due to combination of

- faulty processes
- asynchrony
- safety requirements
- liveness requirement

Ben-Or’s randomized algorithm achieves:

- faulty processes
- asynchrony
- safety requirements
- almost sure termination

⇒ randomized distributed algorithms
Limits of Application Domain

FLP85: There is no asynchronous consensus algorithm: Impossibility due to combination of
- faulty processes
- asynchrony
- safety requirements
- liveness requirement

Ben-Or’s randomized algorithm achieves:
- faulty processes
- asynchrony
- safety requirements
- almost sure termination

⇒ randomized distributed algorithms
Main Goal

Extend previous result to verify randomized distributed algorithms
bool \( v := \text{input\_value}\{0, 1\} \);
int \( r := 1 \);

while (true) do
    send (R,r,v) to all;
    wait for \( n - t \) messages (R,r,*);
    if received \( (n + t) / 2 \) messages (R,r,w)
        then send (P,r,w,D) to all;
    else send (P,r,?) to all;
    wait for \( n - t \) messages (P,r,*);
    if received at least \( t + 1 \) messages (P,r,w,D) then {
        \( v := w \); /* enough support \rightarrow update estimate */
        if received at least \( (n + t) / 2 \) messages (P,r,w,D)
            then decide w; /* strong majority \rightarrow decide */
    } else \( v := \text{random}\{0, 1\} \); /* unclear \rightarrow coin toss */
    \( r := r + 1 \);

od

[Ben-Or, PODC 1983]
Probabilistic Threshold Automata (PTA)

\[ x_0 \geq \frac{n + t}{2} - f \mapsto y_{0++} \]

\[ x_1 \geq \frac{n + t}{2} - f \mapsto y_{1++} \]
Probabilistic Threshold Automata (PTA)

Probabilistic choice
e.g. coin toss

$x_0 \geq (n + t)/2 - f \iff y_{0^{++}}$

$x_1 \geq (n + t)/2 - f \iff y_{1^{++}}$
Probabilistic Threshold Automata (PTA)

\[ x_0 \geq \frac{n + t}{2} - f \iff y_{0++} \]

\[ x_1 \geq \frac{n + t}{2} - f \iff y_{1++} \]

Unboundedly many rounds
Probabilistic Counter System for a PTA

\begin{align*}
x_0 \geq (n + t)/2 - f & \iff y_0++ \\
x_1 \geq (n + t)/2 - f & \iff y_1++ \\
\kappa[l_1, 1] = 1 \\
\kappa[l_1, 2] = 0 \\
\kappa[l_1, 3] = 0 \\
\kappa[E_1, 1] = 2 \\
\kappa[E_1, 2] = 0 \\
\kappa[E_1, 3] = 0 \\
\kappa[D_1, 1] = 1 \\
\kappa[D_1, 2] = 0 \\
\kappa[D_1, 3] = 0 \\
\kappa[D_1, 4] = 0 \\
\end{align*}

how many processes are in every location for every round

\( CT_0 \)
Specifications in LTL-\(\mathcal{X}\) with counters

**Agreement**: No two correct processes decide differently

\[ (\forall k \in \mathbb{N}_0) \ (\forall k' \in \mathbb{N}_0) \ A \ (F \ \kappa[D_v, k] > 0 \ \rightarrow \ G \ \kappa[D_{1-v}, k'] = 0) \]

**Termination**: Under every round-rigid adversary, with probability 1 every correct process eventually decides

\[ P_s \left( \bigvee_{k \in \mathbb{N}_0} \bigvee_{v \in \{0, 1\}} \ G \ \bigwedge_{\ell \in \mathcal{L} \setminus \{D_v\}} \kappa[\ell, k] = 0 \right) = 1 \]

We denote this fragment by **multi-round ELTL\(_{FT}\)**
Challenges

Probabilistic choice

Unboundedly many rounds

Specs with multiple rounds

(∀k) (∀k') A ϕ[k, k']

Specs with probability 1

Ps(ψ[k]) = 1

The Key Idea: Reduction to POPL'17

Elimination of these brings us to the previous non-randomized setting
Two types of specifications

(A) Non-probabilistic

\[(\forall k) (\forall k') A \varphi[k, k']\]

(B) Probabilistic

\[P_s(\psi[k]) = 1\]

Two different strategies for checking them
(A) Non-probabilistic properties

Probabilistic choice

Unboundedly many rounds

Specs with multiple rounds

\((\forall k) (\forall k') A \varphi[k, k']\)
(A) Non-probabilistic properties

Probabilistic choice

Unboundedly many rounds

Specs with multiple rounds

\[ (\forall k) \ (\forall k') \ A \ \varphi[k, k'] \]
(A) Non-probabilistic properties

Probabilistic choice

Unboundedly many rounds

Specs with multiple rounds

\((\forall k) (\forall k') A \varphi[k, k']\)

Non-determinism

One-round system
(A) Non-probabilistic properties

Probabilistic choice

Unboundedly many rounds

Specs with multiple rounds

Non-determinism

One-round system

One-round specs

\((\forall k) (\forall k') A \varphi[k, k']\)

\((\forall k) A \varphi'[k]\)
Reduction to one-round specs

Agreement: if $F$ decision $v$ in $k$ then $G$ no decision $1 - v$ in $k'$

$$(\forall k \in \mathbb{N}_0) \ (\forall k' \in \mathbb{N}_0) \ A \ (F \ \kappa[D_v, k] > 0 \ \rightarrow \ G \ \kappa[D_{1-v}, k'] = 0)$$
Reduction to one-round specs

**Agreement:** if \( F \) decision \( \nu \) in \( k \) then \( G \) no decision \( 1 - \nu \) in \( k' \)

\[
(\forall k \in \mathbb{N}_0) \ (\forall k' \in \mathbb{N}_0) \ \mathbf{A} \ \ (F \ \kappa[D_{\nu}, k] > 0 \ \rightarrow \ \ G \ \kappa[D_{1-\nu}, k'] = 0)
\]

if \( F \) decision \( \nu \) in \( k \) then \( G \) empty final states with \( 1 - \nu \) in \( k \)

\[
(\forall k \in \mathbb{N}_0) \ \mathbf{A} \ \ (F \ \kappa[D_{\nu}, k] > 0 \ \rightarrow \ \ G \ \bigwedge_{\ell \in \mathcal{F}_{1-\nu}} \kappa[\ell, k] = 0) \quad (1)
\]

if \( G \) empty initial with \( 1 - \nu \) in \( k \) then \( G \) empty final with \( 1 - \nu \) in \( k \)

\[
(\forall k \in \mathbb{N}_0) \ \mathbf{A} \ \ (G \ \bigwedge_{\ell \in \mathcal{I}_{1-\nu}} \kappa[\ell, k] = 0 \ \rightarrow \ \ G \ \bigwedge_{\ell \in \mathcal{F}_{1-\nu}} \kappa[\ell, k] = 0) \quad (2)
\]

\((1) \land (2) \rightarrow \text{Agreement} \quad \text{Both are one-round specs}\)
Reduction to one-round specs

**Agreement:** if $F$ decision $v$ in $k$ then $G$ no decision $1 - v$ in $k'$

$$\forall k \in \mathbb{N}_0 \ (\forall k' \in \mathbb{N}_0) \ A \ (F \ \kappa[D_v, k] > 0 \ \rightarrow \ G \ \kappa[D_{1-v}, k'] = 0)$$

if $F$ decision $v$ in $k$ then $G$ empty final states with $1 - v$ in $k$

$$\forall k \in \mathbb{N}_0 \ A \ (F \ \kappa[D_v, k] > 0 \ \rightarrow \ G \ \bigwedge_{\ell \in F_{1-v}} \ \kappa[\ell, k] = 0) \quad (1)$$

if $G$ empty initial with $1 - v$ in $k$ then $G$ empty final with $1 - v$ in $k$

$$\forall k \in \mathbb{N}_0 \ A \ (G \ \bigwedge_{\ell \in I_{1-v}} \ \kappa[\ell, k] = 0 \ \rightarrow \ G \ \bigwedge_{\ell \in F_{1-v}} \ \kappa[\ell, k] = 0) \quad (2)$$

$(1) \wedge (2) \rightarrow$ Agreement

Both are one-round specs
Reduction to one-round system

restrictions on the communication

- originally CSP [Elrad, Francez, 1982]: rendezvous synchronization only within the same round

- message passing variant:
  - messages from past rounds are dropped
  - messages from future rounds are buffered (or dropped)
Reduction to one-round system (cont.)

Solution: reordering transitions and analyzing separate rounds

Round 1

Round 2

Round 3
Problem for analysis:
Fast processes enter a round before slow ones leave the previous round.
Reduction to one-round system (cont.)

Solution: reordering transitions and analyzing separate rounds
Reasoning about round boundaries

Original System:

\[ P_1 \parallel P_2 \parallel \cdots \parallel P_n, \text{ with } P_i = R_i^1 ; R_i^2 ; R_i^3 ; \ldots \]

reduced to

\[ R_1^1 \parallel R_2^1 \parallel \cdots \parallel R_n^1 ; R_1^2 \parallel R_2^2 \parallel \cdots \parallel R_n^2 ; \ldots \]

⇒ Reason about round boundaries only!!

\[ \{ \text{init} \} \ R_1^1 \parallel R_2^1 \parallel \cdots \parallel R_n^1 \ \{ \phi_1 \} ; \ R_1^2 \parallel R_2^2 \parallel \cdots \parallel R_n^2 \ \{ \phi_2 \} ; \ldots \]
One-round system

Reduction preserves \((\forall k) A \varphi'[k]\)

Multi-round and one-round systems satisfy the same one-round specs
Two types of specifications

(A) Non-probabilistic

\((\forall k) (\forall k') A \varphi[k, k']\)

(B) Probabilistic

\(P_s(\psi[k]) = 1\)

Two different strategies for checking them
Almost sure termination: Under every round-rigid adversary, with probability 1 every correct process eventually decides.

\[ \mathbb{P}_s \left( \bigvee_{k \in \mathbb{N}_0} \bigvee_{v \in \{0,1\}} G \bigwedge_{\ell \in \mathcal{L}\setminus\{D_v\}} \kappa[\ell, k] = 0 \right) = 1 \]
Reduction to one-round specs

**Almost sure termination:** Under every round-rigid adversary, with probability 1 every correct process eventually decides.

\[
P_s \left( \bigvee_{k \in \mathbb{N}_0} \bigvee_{v \in \{0,1\}} G \land \bigwedge_{\ell \in \mathcal{L} \setminus \{D_v\}} \kappa[\ell, k] = 0 \right) = 1
\]

with positive probability \( p \), empty final with \( 1 - v \) in \( k \)

\[
P_s^\sigma \left( \bigvee_{v \in \{0,1\}} G \land \bigwedge_{\ell \in \mathcal{F}_1} \kappa[\ell, k] = 0 \right) > p > 0 \quad (3)
\]

if \( G \) empty initial with \( 1 - v \) in \( k \) then \( F \) all decide \( v \) in \( k \)

\[
(\forall k \in \mathbb{N}_0) \text{ A } \left( G \land \bigwedge_{\ell \in \mathcal{I}_{1-v}} \kappa[\ell, k] = 0 \right) \rightarrow \left( G \land \bigwedge_{\ell \in \mathcal{F} \setminus D_v} \kappa[\ell, k] = 0 \right) \quad (4)
\]

\((3) \land (4) \rightarrow \text{ Almost-sure termination} \quad \text{Both are one-round specs}\)
Reduction to one-round specs

**Almost sure termination:** Under every round-rigid adversary, with probability 1 every correct process eventually decides.

\[
\mathbb{P}_s \left( \bigvee_{k \in \mathbb{N}_0} \bigvee_{v \in \{0,1\}} G \land \bigwedge_{\ell \in \mathcal{L} \setminus \{D_v\}} \kappa[\ell, k] = 0 \right) = 1
\]

with positive probability \( p \), empty final with \( 1 - v \) in \( k \)

\[
\mathbb{P}_s^\sigma \left( \bigvee_{v \in \{0,1\}} G \land \bigwedge_{\ell \in \mathcal{F}_{1-v}} \kappa[\ell, k] = 0 \right) > p > 0 \quad \text{(3)}
\]

if \( G \) empty initial with \( 1 - v \) in \( k \) then \( F \) all decide \( v \) in \( k \)

\[
(\forall k \in \mathbb{N}_0) \ A \ (G \land \bigwedge_{\ell \in \mathcal{I}_{1-v}} \kappa[\ell, k] = 0) \rightarrow G \land \bigwedge_{\ell \in \mathcal{F} \setminus D_v} \kappa[\ell, k] = 0 \quad \text{(4)}
\]

\((3) \land (4) \rightarrow \text{Almost-sure termination} \quad \text{Both are one-round specs}\)
Round-rigid adversaries

Adversary: prefix $\mapsto$ or
branching for coin toss

Round-rigid adversary

- respects round order
- branching at the end of each round
Checking $\mathbb{P}_{\sigma} (\varphi[k]) > 0$

All outcomes have probability $p \geq (1/2)^n > 0$
Checking $\mathbb{P}_s^\sigma (\varphi[k]) > 0$

Capture that some processes toss a coin

All outcomes have probability $p \geq (1/2)^n > 0$
Abstracting Coin-Toss Outcomes

A non-probabilistic threshold automaton
Experimental evaluation

We have verified 6 parameterized randomized distributed algorithms:

<table>
<thead>
<tr>
<th>Automaton</th>
<th>S1/S1'/S1&quot;</th>
<th>S2</th>
<th>S3</th>
<th>S4</th>
<th>S5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>L</td>
<td>R</td>
<td>S</td>
<td>Time</td>
<td>L</td>
</tr>
<tr>
<td>ben-or-cc</td>
<td>10</td>
<td>27</td>
<td>9</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>ben-or-dc</td>
<td>11</td>
<td>32</td>
<td>9</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>ben-or-byz</td>
<td>9</td>
<td>18</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>rabc-cr</td>
<td>11</td>
<td>31</td>
<td>9</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>kset</td>
<td>13</td>
<td>58</td>
<td>65</td>
<td>3</td>
<td>65</td>
</tr>
<tr>
<td>rs-bosco</td>
<td>19</td>
<td>48</td>
<td>156M</td>
<td>3:21</td>
<td>156M</td>
</tr>
</tbody>
</table>

[forseyte.at/software/bymc/artifact82]
Conclusions

We can efficiently verify **randomized distributed algorithms** that are:

- asynchronous and parameterized, fault-tolerant
- counting messages and comparing to **threshold guards**
- non-probabilistic specs,
- specs with probability 1, under **round-rigid adversaries**

Future work

- more general adversaries

[https://hal.inria.fr/hal-01925533]
Conclusions

We can efficiently verify randomized distributed algorithms that are:

- asynchronous and parameterized, fault-tolerant
- counting messages and comparing to threshold guards
- non-probabilistic specs,
- specs with probability 1, under round-rigid adversaries

Future work

- more general adversaries

Thank you!

[ https://hal.inria.fr/hal-01925533 ]