Runtime Verification of Distributed System

Ritam Ganguly

MICHIGAN STATE UNIVERSITY

April 18, 2023
Outline of the talk

1. Motivation and Preliminaries
2. Contributions
   1. Centralized runtime verification w.r.t. LTL specifications
   2. Centralized runtime verification w.r.t. MTL specifications
   3. Decentralized runtime verification w.r.t. LTL specifications
   4. Decentralized runtime verification w.r.t. stream-based specifications
3. Conclusion and Future Work
Outline of the talk

1. Motivation and Preliminaries

2. Contributions
   1. Centralized runtime verification w.r.t. LTL specifications
   2. Centralized runtime verification w.r.t. MTL specifications
   3. Decentralized runtime verification w.r.t. LTL specifications
   4. Decentralized runtime verification w.r.t. stream-based specifications

3. Conclusion and Future Work
Distributed Database
Distributed Database

- Student(id, name, address)

- Enrollment(id, name, course)
Distributed Database

- Student(id, name, address)
- [1234, 'Leslie Lamport', '126 Spartan Dr.]

- Enrollment(id, name, course)
- [1234, 'Edsger Dijkstra', 'CSE 260']
Distributed Database

• Student(id, name, address)
  • [1234, ‘Leslie Lamport’, ‘126 Spartan Dr.’]

• Enrollment(id, name, course)
  • [1234, ‘Edsger Dijkstra’, ‘CSE 260’]
Cross-chain Transactions

Smart contracts

Bob

1000

Alice

900
Cross-chain Transactions

Smart contracts

- Smart contract is a program running on the blockchains, triggered automatically and no human intervention can stop it.
Cross-chain Transactions
Smart contracts

- Smart contract is a program running on the blockchains, triggered automatically and no human intervention can stop it.
Cross-chain Transactions
Smart contracts

- Smart contract is a program running on the blockchains, triggered automatically and no human intervention can stop it.
Cross-chain Transactions
Smart contracts

- Smart contract is a program running on the blockchains, triggered automatically and no human intervention can stop it.
Cross-chain Transactions
Smart contracts

- Smart contract is a program running on the blockchains, triggered automatically and no human intervention can stop it.
Cross-chain Transactions

Smart contracts

- Smart contract is a program running on the blockchains, triggered automatically and no human intervention can stop it.

- Enforce the (time sensitive) conditions of the smart contracts.
Industrial Control System

Water Treatment Plant

NaCl  HCl  NaOCl

UV
Industrial Control System
Water Treatment Plant

• A Programmable Logic Controller (PLC) requires time sensitive aggregated data to make well informed decisions.
Industrial Control System
Water Treatment Plant

• A Programmable Logic Controller (PLC) requires time sensitive aggregated data to make well informed decisions.
Industrial Control System
Water Treatment Plant

- A Programmable Logic Controller (PLC) requires time sensitive aggregated data to make well informed decisions.
- The data produced by the components are often the target of cyber attack putting the security of the system in jeopardy.
Common Features
Common Features

- **Safety-critical systems**: Failure could result in loss of life, significant property damage or damage to the environment.
Common Features

• **Safety-critical systems**: Failure could result in loss of life, significant property damage or damage to the environment.

• **Geographically separated**: Each component is often located in a different geographical location.
Common Features

- **Safety-critical systems**: Failure could result in loss of life, significant property damage or damage to the environment.
- **Geographically separated**: Each component is often located in a different geographical location.
- **System specification**: System should follow a set of pre-mentioned system properties.
Verifying System Properties
Verifying System Properties

- Model Checking
Verifying System Properties

• Model Checking
Verifying System Properties

• **Model Checking**
  • Exhaustive ✓
  • Costly ✗
Verifying System Properties

- **Model Checking**
  - Exhaustive ✓
  - Costly ×

- **Testing**
Verifying System Properties

- **Model Checking**
  - Exhaustive ✓
  - Costly ✗

- **Testing**

  \[a \rightarrow b \rightarrow e \rightarrow a \rightarrow c \rightarrow d \rightarrow e \rightarrow \ldots\]

  \[a \rightarrow c \rightarrow d \rightarrow g \rightarrow f \rightarrow d \rightarrow h \rightarrow \ldots\]

  \[a \rightarrow c \rightarrow d \rightarrow e \rightarrow f \rightarrow g \rightarrow d \rightarrow \ldots\]

  \[\vdots\]
Verifying System Properties

- **Model Checking**
  - Exhaustive ✓
  - Costly ✗

- **Testing**
  - Not Exhaustive ✗
  - Cheap ✓

\[
a \rightarrow b \rightarrow e \rightarrow a \rightarrow c \rightarrow d \rightarrow e \rightarrow \cdots
\]
\[
a \rightarrow c \rightarrow d \rightarrow g \rightarrow f \rightarrow d \rightarrow h \rightarrow \cdots
\]
\[
a \rightarrow c \rightarrow d \rightarrow e \rightarrow f \rightarrow g \rightarrow d \rightarrow \cdots
\]
\[\vdots\]
Runtime Verification
Runtime Verification (RV)
Runtime Verification (RV)

- A lightweight technique where a monitor continually inspects the health of a system under inspection at run time with respect to a formal specification.
Runtime Verification (RV)

- A lightweight technique where a monitor continually inspects the health of a system under inspection at run time with respect to a formal specification.

- In distributed RV, one or more monitors observe the behavior of a distributed system at run time and collectively verify its correctness with respect to its specification.
Distributed System
Distributed System

$P_1$ → $P_2$
Distributed System

$P_1$  $e_1^0$  $e_1^1$  $e_1^2$  $e_1^3$  $e_1^4$

$P_2$  $e_2^0$  $e_2^1$  $e_2^2$  $e_2^3$  $e_2^4$

Local State
Distributed System

\[ P_1 \]
\[ e_0^1 \quad e_1^1 \quad e_2^2 \quad e_3^3 \quad e_4^4 \]

\[ P_2 \]
\[ e_0^2 \quad e_1^1 \quad e_2^2 \quad e_3^3 \quad e_4^4 \]

Time of occurrence

p: false
r: false
Distributed System

P_1

(1, \{p\})
(2, \emptyset)

e_1^0

e_1^1

\rightarrow

e_1^2

4

(7, \{p\})

P_2

(1, \emptyset)

\rightarrow

P_2

(1, \emptyset)

\rightarrow

(3, 4, \emptyset)

\rightarrow

(7, \emptyset)

\rightarrow

(8, \{r\})

Send

Receive
Distributed System

\[ e_1^0 \rightarrow e_1^1 \rightarrow e_1^2 \rightarrow e_1^3 \rightarrow e_1^4 \]

\[ e_2^0 \rightarrow e_2^1 \rightarrow e_2^2 \rightarrow e_2^3 \rightarrow e_2^4 \]
Distributed System

$e_1^0 \leadsto e_1^1$

$e_1^3 \leadsto e_1^2$

$e_1^0 \leadsto e_1^1$

$e_1^3 \leadsto e_1^2$

$P_1$

$P_2$

happened-before
Distributed System

$P_1$ (1, \{p\})  (2, \emptyset)  4  6  (7, \{p\})

$P_2$ (1, \emptyset)  3  7 (8, \{r\})
Distributed System

Consistent cut

\((e \sim f) \land (f \in C) \rightarrow (e \in C)\)
Distributed System

Consistent cut

\((e \sim f) \land (f \in C) \rightarrow (e \in C)\)
Distributed System

Consistent cut

\[ ((e \sim f) \land (f \in C)) \rightarrow (e \in C) \]
Consistent cut

\[ ((e \sim f) \land (f \in C)) \rightarrow (e \in C) \]
Distributed System

P1

(1, \{p\})

\(e_1^0\)

\(e_1^1\)

(2, \emptyset)

\(e_1^2\)

4

\(e_1^3\)

6

\(e_1^4\)

(7, \{p\})

\(e_1^5\)

P2

(1, \emptyset)

\(e_2^0\)

\(e_2^1\)

3

\(e_2^2\)

(4, \emptyset)

\(e_2^3\)

7

\(e_2^4\)

(8, \{r\})
Distributed System

Sequence of consistent cuts

\[ C_0 C_1 C_2 \cdots \]
Distributed System

Sequence of consistent cuts

$C_0 C_1 C_2 \cdots$
Distributed System

Sequence of consistent cuts

$C_0 C_1 C_2 \cdots$
Distributed System

Sequence of consistent cuts

\[ C_0 C_1 C_2 \cdots \]
Distributed System

Sequence of consistent cuts

$C_0 C_1 C_2 \cdots$
Distributed System

Sequence of consistent cuts $C_0 C_1 C_2 \cdots$
Distributed System

Sequence of consistent cuts

$C_0 C_1 C_2 \cdots$
Distributed System

Sequence of consistent cuts

\( C_0 C_1 C_2 \cdots \)
Distributed System

Sequence of consistent cuts
\( C_0 C_1 C_2 \ldots \)
Distributed System

Sequence of consistent cuts

\[ C_0C_1C_2\ldots \]
Distributed System

Sequence of consistent cuts

$C_0 C_1 C_2 \cdots$
Distributed System

 Sequence of consistent cuts

$C_0C_1C_2\ldots$
Challenges in Distributed RV

\[\bigcirc (\neg p \rightarrow (\neg p \cup r))\]
Challenges in Distributed RV

Next

If-then-else

Until
Challenges in Distributed RV

P1

\( e_1^0 \)  (1, \( p \land \neg r \))  \( e_1^1 \)  (2, \( \neg p \land \neg r \))  \( e_1^2 \)  (4)  \( e_1^3 \)  (6)  \( e_1^4 \)  (7, \( p \land \neg r \))

P2

\( e_2^0 \)  (1, \( \neg p \land \neg r \))  \( e_2^1 \)  (3)  \( e_2^2 \)  (4, \( \neg p \land \neg r \))  \( e_2^3 \)  (7)  \( e_2^4 \)  (8, \( \neg p \land r \))

\( \bigcirc \left( \neg p \to (\neg p \cup r) \right) \)
Challenges in Distributed RV

\[
\bigcirc (\neg p \rightarrow (\neg p \cup r))
\]
Challenges in Distributed RV

\[ P_1 \]
\[ e_1^0 \]  (1, \( p \land \neg r \))  \[ e_1^1 \]  (2, \( \neg p \land \neg r \))  \[ e_1^2 \]  (4, \( \neg p \land \neg r \))  \[ e_1^3 \]  (6, \( p \land \neg r \))  \[ e_1^4 \]  (7, \( p \land \neg r \))

\[ P_2 \]
\[ e_2^0 \]  (1, \( \neg p \land \neg r \))  \[ e_2^1 \]  (3, \( \neg p \land \neg r \))  \[ e_2^2 \]  (4, \( \neg p \land \neg r \))  \[ e_2^3 \]  (7, \( \neg p \land r \))  \[ e_2^4 \]  (8, \( \neg p \land r \))

\[ \bigcirc (\neg p \rightarrow (\neg p \cup r)) \]
Challenges in Distributed RV

\[
\bigcirc (\neg p \rightarrow (\neg p \cup r))
\]

true
Challenges in Distributed RV

\[ \circlearrowleft (\neg p \rightarrow (\neg p \vee r)) \]

true
Challenges in Distributed RV

\[ \bigcirc (\neg p \rightarrow (\neg p \cup r)) \]

true \hspace{5cm} false
Present Solutions
Present Solutions

• Synchronous
  • A. Bauer and Y. Falcone. Decentralized LTL Monitoring (FM 2012).
  • L. M. Danielsson and C. Sanchez. Decentralized stream runtime verification (RV 2019)
Present Solutions

• Synchronous
  • A. Bauer and Y. Falcone. Decentralized LTL Monitoring (FM 2012).
  • L. M. Danielsson and C. Sanchez. Decentralized stream runtime verification (RV 2019)
• Asynchronous
Present Solutions

• Synchronous
  • A. Bauer and Y. Falcone. Decentralized LTL Monitoring (FM 2012).
  • L. M. Danielsson and C. Sanchez. Decentralized stream runtime verification (RV 2019)

• Asynchronous

• Partially-synchronous
Distributed System

Partially-Synchronous

\[ (1, \{p\}) \]
\[ (2, \emptyset) \]
\[ 4 \]
\[ 6 \]
\[ (7, \{p\}) \]

\[ (1, \emptyset) \]
\[ 3 \]
\[ 4 \]
\[ 6 \]
\[ 7 \]
\[ (8, \{r\}) \]
Partially-Synchronous Distributed System
Partially-Synchronous

Distributed System

\[ (1, \{p\}) \quad (2, \emptyset) \quad 4 \quad (7, \{p\}) \]

\[ (1, \emptyset) \quad (4, \emptyset) \quad 7 \quad (8, \{r\}) \]
Partially-Synchronous

Distributed System

happened-before

\[
\begin{align*}
(1, \{p\}) & \quad e_1^0 & \quad e_1^1 & \quad e_2^0 & \quad e_2^1 & \quad e_1^2 & \quad e_1^3 & \quad e_1^4 \\
(2, \emptyset) & \quad e_1^0 & \quad e_1^1 & \quad e_2^0 & \quad e_2^1 & \quad e_1^2 & \quad e_1^3 & \quad e_1^4 \\
4 & \quad e_1^2 & \quad e_1^3 & \quad e_1^4 & \quad e_1^3 & \quad e_1^4 & \quad e_1^3 & \quad e_1^4 \\
6 & \quad e_1^2 & \quad e_1^3 & \quad e_1^4 & \quad e_1^3 & \quad e_1^4 & \quad e_1^3 & \quad e_1^4 \\
(7, \{p\}) & \quad e_1^2 & \quad e_1^3 & \quad e_1^4 & \quad e_1^3 & \quad e_1^4 & \quad e_1^3 & \quad e_1^4 \\
(8, \{r\}) & \quad e_1^2 & \quad e_1^3 & \quad e_1^4 & \quad e_1^3 & \quad e_1^4 & \quad e_1^3 & \quad e_1^4 \\
\end{align*}
\]
Partially-Synchronous

Distributed System

$P_1$

$P_2$

$e^0_1 \leadsto e^1_2$

$happened-before$

$e^3_1 \leadsto e^4_2$

$e^3_1 \leadsto e^4_2$

$(1, \{p\})$

$(2, \emptyset)$

$4$

$6$

$(7, \{p\})$

$(1, \emptyset)$

$3$

$(4, \emptyset)$

$7$

$(8, \{r\})$
Consistent cut

\((e \sim f) \land (f \in C) \rightarrow (e \in C)\)
Partially-Synchronous

Distributed System

Consistent cut

\((e \leadsto f) \land (f \in C) \rightarrow (e \in C)\)
Consistent cut

\[ ((e \sim f) \land (f \in C)) \rightarrow (e \in C) \]
Consistent cut

\((e \sim f) \land (f \in C) \rightarrow (e \in C)\)
Consistent cut

\[(e \rightsquigarrow f) \land (f \in C) \rightarrow (e \in C)\]
Runtime Verification of a partially-synchronous distributed system in real-time is feasible.
Contributions
A graphical representation

Runtime Verification of a partially-synchronous distributed system in real-time is feasible.
Contributions
A graphical representation

Runtime Verification of a partially-synchronous distributed system in real-time is feasible.
Contributions
A graphical representation

Runtime Verification of a partially-synchronous distributed system in real-time is feasible.
Contributions
A graphical representation

Runtime Verification of a partially-synchronous distributed system in real-time is feasible.
Contributions
A graphical representation

Runtime Verification of a partially-synchronous distributed system in real-time is feasible.

1. Centralized monitor
   LTL spec.

2. Centralized monitor
   MTL spec.
Contributions
A graphical representation

Runtime Verification of a partially-synchronous distributed system in real-time is feasible.
Contributions
A graphical representation

Runtime Verification of a partially-synchronous distributed system in real-time is feasible.
Contributions

1. Monitoring Distributed System under Partial Synchrony
   Ritam Ganguly, Anik Momtaz, Borzoo Bonakdarpour; OPODIS 2020

2. Runtime Verification of Partially-Synchronous Distributed System
   Ritam Ganguly, Anik Momtaz, Borzoo Bonakdarpour; FMSD (minor revision)

3. Distributed Runtime Verification of Metric Temporal Properties for Cross-Chain Protocols
   Ritam Ganguly, Yingjie Xue, Aaron Jonckheere, Parker Ljung, Benjamin Schornstein, Borzoo Bonakdarpour, Maurice Herlihy; ICDCS 2022

4. Crash-Resilient Decentralized Synchronous Runtime Verification
   Ritam Ganguly, Shokufeh Kazemloo, Borzoo Bonakdarpour, TDSC

5. Decentralized Runtime Verification of Stream-based Partially-Synchronous Distributed System
   Ritam Ganguly, Borzoo Bonakdarpour, EMSOFT 2023 (in review)
## Contributions

<table>
<thead>
<tr>
<th></th>
<th>Title</th>
<th>Authors</th>
<th>Conference/Poster Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Monitoring Distributed System under Partial Synchrony</td>
<td>Ritam Ganguly, Anik Momtaz, Borzoo Bonakdarpour; OPODIS 2020</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Runtime Verification of Partially-Synchronous Distributed System</td>
<td>Ritam Ganguly, Anik Momtaz, Borzoo Bonakdarpour; FMSD (minor revision)</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Distributed Runtime Verification of Metric Temporal Properties for Cross-Chain Protocols</td>
<td>Ritam Ganguly, Yingjie Xue, Aaron Jonckheere, Parker Ljung, Benjamin Schornstein, Borzoo Bonakdarpour, Maurice Herlihy; ICDCS 2022</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Crash-Resilient Decentralized Synchronous Runtime Verification</td>
<td>Ritam Ganguly, Shokufeh Kazemloo, Borzoo Bonakdarpour, TDSC</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Decentralized Runtime Verification of Stream-based Partially-Synchronous Distributed System</td>
<td>Ritam Ganguly, Borzoo Bonakdarpour, EMSOFT 2023 (in review)</td>
<td></td>
</tr>
</tbody>
</table>
Overview
Overview

• A fault-proof central monitor verifying a partially-synchronous distributed system with respect to Linear Temporal Logic (LTL) specifications.
Overview

• A fault-proof central monitor verifying a partially-synchronous distributed system with respect to Linear Temporal Logic (LTL) specifications.

• Introduce two SMT-based approaches
  • Automata-based approach
  • Progression-based approach
Linear Temporal Logic

Finite Trace Example
Linear Temporal Logic

Finite Trace Example

\( \sigma_1 \models \Diamond b \)
Linear Temporal Logic

Finite Trace Example

$\neg b \rightarrow \neg b$

$\sigma_1 \models \Diamond b$
Linear Temporal Logic
Finite Trace Example

$$\neg b \implies \neg b \implies \neg b$$

$$\sigma_1 \models \Diamond b$$

?
Linear Temporal Logic

Finite Trace Example

\[ \neg b \rightarrow \neg b \rightarrow \neg b \rightarrow \neg b \]

\[ \sigma_1 \models \Diamond b \quad ? \]
Linear Temporal Logic
Finite Trace Example

$\neg b \rightarrow \neg b \rightarrow \neg b \rightarrow \neg b \rightarrow \neg b \rightarrow b$

$\sigma_1 \models \Diamond b$  $\top$
Linear Temporal Logic
Finite Trace Example

$\neg b \rightarrow \neg b \rightarrow \neg b \rightarrow \neg b \rightarrow b$

$\sigma_1 \models \Diamond b \quad \top$

$\sigma_1 \models \Box a \quad ?$
Linear Temporal Logic

Finite Trace Example

\(\neg b \rightarrow \neg b \rightarrow \neg b \rightarrow \neg b \rightarrow \neg b \rightarrow b\)

\(\sigma_1 \models \diamond b \quad \top\)

\(\sigma_1 \models \square a \quad ?\)
Linear Temporal Logic
Finite Trace Example

\[ \sigma_1 \models \Diamond b \quad \top \]

\[ \sigma_1 \models \Box a \quad ? \]

\[ \neg b \rightarrow \neg b \rightarrow \neg b \rightarrow \neg b \rightarrow \neg b \rightarrow b \]

\[ a \rightarrow a \rightarrow a \]
Linear Temporal Logic

Finite Trace Example

$\neg b \rightarrow \neg b \rightarrow \neg b \rightarrow \neg b \rightarrow \neg b \rightarrow b$

$a \rightarrow a \rightarrow a \rightarrow a \rightarrow a$

$\sigma_1 \models \lozenge b \quad \top$

$\sigma_1 \models \Box a \quad ?$
Linear Temporal Logic
Finite Trace Example

\[ \sigma_1 \models \Diamond b \quad \top \]

\[ \sigma_1 \not\models \square a \quad \bot \]

\[ \sigma_1 \models a \mathcal{U} b \quad ? \]
Linear Temporal Logic

Finite Trace Example

\( \sigma_1 \models \diamond b \quad \top \)

\( \sigma_1 \models \Box a \quad \bot \)

\( \sigma_1 \models a \mathcal{U} b \quad ? \)
Linear Temporal Logic
Finite Trace Example

$\neg b \rightarrow \neg b \rightarrow \neg b \rightarrow \neg b \rightarrow \neg b \rightarrow b$

$\sigma_1 \models \diamond b \quad \top$

$a \rightarrow a \rightarrow a \rightarrow a \rightarrow a \rightarrow \neg a$

$\sigma_1 \models \Box a \quad \bot$

$a \rightarrow a \rightarrow a$

$\sigma_1 \models a \mathcal{U} b \quad ?$
Linear Temporal Logic
Finite Trace Example

\(\sigma_1 \models \lozenge b\)  \(\top\)

\(\sigma_1 \not\models \square a\)  \(\bot\)

\(\sigma_1 \not\models a \mathcal{U} b\)  ?
Linear Temporal Logic

Finite Trace Example

\( \neg b \rightarrow \neg b \rightarrow \neg b \rightarrow \neg b \rightarrow \neg b \rightarrow b \)

\( a \rightarrow a \rightarrow a \rightarrow a \rightarrow a \rightarrow \neg a \)

\( a \rightarrow a \rightarrow a \rightarrow a \rightarrow a \rightarrow b \)

\( \sigma_1 \models \Diamond b \quad \top \)

\( \sigma_1 \models \Box a \quad \bot \)

\( \sigma_1 \models a \mathcal{U} b \quad \top \)
Proposed Solution
Automata-based RV for LTL Specifications

\[(1, p \land \neg r)\]  \[(2, \neg p \land \neg r)\]  \[(4, \neg p \land \neg r)\]

\[(1, \neg p \land \neg r)\]  \[(3, \neg p \land \neg r)\]  \[(4, \neg p \land \neg r)\]
Proposed Solution
Automata-based RV for LTL Specifications

\[ (1, p \land \neg r) \quad (2, \neg p \land \neg r) \]

\[ (1, \neg p \land \neg r) \quad 3 \quad (4, \neg p \land \neg r) \]

\[ (4, \neg p \land \neg r) \quad 6 \quad (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \quad 7 \quad (8, \neg p \land r) \]

\[ q_0 \quad \text{true} \quad q_1 \quad \neg p \land \neg r \quad q_2 \quad \neg p \land \neg r \]

\[ q_1 \quad p \vee r \quad q_2 \quad p \land r \quad \neg p \land r \]

\[ q_\top \quad \text{true} \quad q_\bot \quad \text{true} \]

\[ (\neg p \rightarrow (\neg p \varUpsilon r)) \]
Proposed Solution
Automata-based RV for LTL Specifications

- Check satisfiability for every path in the LTL₃ monitor.
- Current state consists of a set of possible states.
Proposed Solution
Automata-based RV for LTL Specifications

- Check satisfiability for every path in the LTL₃ monitor.
- Current state consists of a set of possible states.
Proposed Solution

Automata-based RV for LTL Specifications

- Check satisfiability for every path in the LTL₃ monitor.
- Current state consists of a set of possible states.
Proposed Solution
Automata-based RV for LTL Specifications

- Check satisfiability for every path in the LTL₃ monitor.
- Current state consists of a set of possible states.
Proposed Solution

Automata-based RV for LTL Specifications

- Check satisfiability for every path in the LTL$_3$ monitor.
- Current state consists of a set of possible states.
Proposed Solution
Automata-based RV for LTL Specifications

- Check satisfiability for every path in the LTL₃ monitor.
- Current state consists of a set of possible states.
Proposed Solution
Automata-based RV for LTL Specifications

- Check satisfiability for every path in the LTL₃ monitor.
- Current state consists of a set of possible states.
Proposed Solution

Automata-based RV for LTL Specifications

- Check satisfiability for every path in the $\text{LTL}_3$ monitor.
- Current state consists of a set of possible states.
Proposed Solution

Automata-based RV for LTL Specifications

- Check satisfiability for every path in the LTL$^3$ monitor.
- Current state consists of a set of possible states.
Proposed Solution
Automata-based RV for LTL Specifications

• Check satisfiability for every path in the LTL3 monitor.
• Current state consists of a set of possible states.
Proposed Solution
Automata-based RV for LTL Specifications

• Check satisfiability for every path in the LTL₃ monitor.
• Current state consists of a set of possible states.
Proposed Solution

Automata-based RV for LTL Specifications

- Check satisﬁability for every path in the LTL3 monitor.
- Current state consists of a set of possible states.
Proposed Solution
Automata-based RV for LTL Specifications

• Check satisfiability for every path in the LTL₃ monitor.
• Current state consists of a set of possible states.
Proposed Solution
Automata-based RV for LTL Specifications

• Check satisfiability for every path in the LTL₃ monitor.
• Current state consists of a set of possible states.
Proposed Solution
Automata-based RV for LTL Specifications

- Check satisfiability for every path in the LTL$_3$ monitor.
- Current state consists of a set of possible states.
Proposed Solution
Automata-based RV for LTL Specifications

- Check satisfiability for **every path** in the LTL₃ monitor.
- Current state consists of a **set of possible states**.
Proposed Solution

Automata-based RV for LTL Specifications

• Check satisfiability for every path in the LTL₃ monitor.
• Current state consists of a set of possible states.
Proposed Solution
Automata-based RV for LTL Specifications

• Check satisfiability for every path in the LTL$_3$ monitor.
• Current state consists of a set of possible states.
Proposed Solution
Automata-based RV for LTL Specifications

- Check satisfiability for every path in the LTL₃ monitor.
- Current state consists of a set of possible states.
Proposed Solution
Automata-based RV for LTL Specifications

• Check satisfiability for every path in the LTL$_3$ monitor.
• Current state consists of a set of possible states.
Proposed Solution
Automata-based RV for LTL Specifications

- Check satisfiability for **every path** in the LTL$_3$ monitor.
- Current state consists of a **set of possible states**.
Proposed Solution
Automata-based RV for LTL Specifications

- Check satisfiability for every path in the LTL\textsubscript{3} monitor.
- Current state consists of a set of possible states.
Proposed Solution

Automata-based RV for LTL Specifications

- Check satisfiability for every path in the LTL$_3$ monitor.
- Current state consists of a set of possible states.
Proposed Solution
Automata-based RV for LTL Specifications

• Check satisfiability for every path in the LTL₃ monitor.
• Current state consists of a set of possible states.
Proposed Solution

Automata-based RV for LTL Specifications

- Check satisfiability for every path in the LTL$_3$ monitor.
- Current state consists of a set of possible states.
Proposed Solution

Automata-based RV for LTL Specifications

- Check satisfiability for every path in the LTL₃ monitor.
- Current state consists of a set of possible states.
Proposed Solution

Progression-based for LTL Specifications

\[ 
\begin{align*}
\mathcal{P}_1 & \quad e_1^0 \quad e_1^4 \quad e_1^2 \\
(1, \neg p \wedge \neg r) & \quad (2, \neg p \wedge \neg r) & \quad 4 \\
\mathcal{P}_2 & \quad e_2^0 \quad e_2^1 \quad e_2^2 \\
(1, \neg p \wedge \neg r) & \quad 3 & \quad (4, \neg p \wedge \neg r) \\
\mathcal{P}_1 & \quad e_1^4 \quad e_1^3 \quad e_1^4 \\
4 & \quad 6 & \quad (7, p \wedge \neg r) \\
\mathcal{P}_2 & \quad e_2^2 \quad e_2^3 \quad e_2^3 \\
(4, \neg p \wedge \neg r) & \quad 7 & \quad (8, \neg p \wedge r) \\
\end{align*}
\]

\[ \Box (\neg p \Rightarrow (\neg p \mathcal{U} r)) \]
Proposed Solution
Progression-based for LTL Specifications

\[ (1, p \land \neg r) \quad (2, \neg p \land \neg r) \]
\[ (1, \neg p \land \neg r) \quad (3, \neg p \land \neg r) \]
\[ (4, \neg p \land \neg r) \]

\[ \bigcirc (\neg p \rightarrow (\neg p \mathcal{U} r)) \]
Proposed Solution
Progression-based for LTL Specifications

\[
\circ \left( \neg p \rightarrow (\neg p \mathcal{U} r) \right)
\]

\[
\begin{align*}
\circ \neg p & \\
\circ \Box \neg p & \\
\circ \Diamond r & 
\end{align*}
\]
Proposed Solution
Progression-based for LTL Specifications

\( (1, p \land \neg r) \)

\( (2, \neg p \land \neg r) \)

\( (3, \neg p \land \neg r) \)

\( (4, \neg p \land \neg r) \)

\( (7, p \land \neg r) \)

\( (8, \neg p \land r) \)

\( \bigcirc (\neg p \rightarrow (\neg p \mathcal{U} r)) \)

\( \bigcirc \neg p \)

true

\( \bigcirc \square \neg p \)

true

\( \bigcirc \diamond r \)
Proposed Solution

Progression-based for LTL Specifications

\[ (1, p \land \neg r) \]
\[ (2, \neg p \land \neg r) \]
\[ (3, \neg p \land \neg r) \]
\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]
\[ (8, \neg p \land r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]
\[ (7, p \land \neg r) \]
Proposed Solution
Progression-based for LTL Specifications

\[ \bigcirc (\neg p \rightarrow (\neg p \bigcirc \neg r)) \]

\[ \bigcirc \neg p \quad \bigcirc \Box \neg p \quad \bigcirc \Diamond r \]

true \quad false \quad true \quad false \quad false
Proposed Solution
Progression-based for LTL Specifications

\[ (1, p \land \neg r) \]

\[ (2, \neg p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]

\[ (7, p \land \neg r) \]

\[ (8, \neg p \land r) \]

\[ \bigcirc (\neg p \rightarrow (\neg p \mathcal{U} r)) \]

\[ \bigcirc \neg p \quad \bigcirc \square \neg p \quad \bigcirc \diamond r \]

true false true false false

\[ \neg p \mathcal{U} r \quad \text{true} \]
Proposed Solution
Progression-based for LTL Specifications

\[
\neg p \not\forall r
\]
Proposed Solution
Progression-based for LTL Specifications

\[ (1, p \land \neg r) \]

\[ (2, \neg p \land \neg r) \]

\[ (3, \neg p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]

\[ (5, \neg p \land r) \]

\[ (6, \neg p \land \neg r) \]

\[ (7, p \land \neg r) \]

\[ (8, \neg p \land r) \]

\[ \neg p \not\in \mathcal{U} r \]

\[ \text{true} \]
Proposed Solution
Progression-based for LTL Specifications

¬p ∄ r
□ ¬p

true
◊ r
Proposed Solution
Progression-based for LTL Specifications

¬p ⊨ r
□¬p
false

d true
◊r
false
Proposed Solution
Progression-based for LTL Specifications

\[-p \not\cup r\]
\[\square \neg p\]
\[\Diamond r\]

true false true false
Proposed Solution
Progression-based for LTL Specifications

Proposed Solution
Progression-based for LTL Specifications

\( (1, \neg p \land \neg r) \)
\( (2, \neg p \land \neg r) \)
\( (4, \neg p \land \neg r) \)
\( (7, p \land \neg r) \)

\( \neg p \ \mathcal{U} \ r \)
\( \Box \neg p \)

true
false
true
false
true
false
Results
Results
Results
Results
# Contributions

|   | 1 | Monitoring Distributed System under Partial Synchrony  
|   | Ritam Ganguly, Anik Momtaz, Borzoo Bonakdarpour; OPODIS 2020 |
|   | 2 | Runtime Verification of Partially-Synchronous Distributed System  
|   | Ritam Ganguly, Anik Momtaz, Borzoo Bonakdarpour; FMSD (minor revision) |
|   | 3 | Distributed Runtime Verification of Metric Temporal Properties for Cross-Chain Protocols  
|   | Ritam Ganguly, Yingjie Xue, Aaron Jonckheere, Parker Ljung, Benjamin Schornstein, Borzoo Bonakdarpour, Maurice Herlihy; ICDCS 2022 |
|   | 4 | Crash-Resilient Decentralized Synchronous Runtime Verification  
|   | Ritam Ganguly, Shokufeh Kazemloo, Borzoo Bonakdarpour, TDSC |
|   | 5 | Decentralized Runtime Verification of Stream-based Partially-Synchronous Distributed System  
|   | Ritam Ganguly, Borzoo Bonakdarpour, EMSOFT 2023 (in review) |
Overview
Overview

• A fault-proof central monitor verifying a partially-synchronous distributed system with respect to Metric Temporal Logic (MTL) specifications.
Overview

- A fault-proof central monitor verifying a partially-synchronous distributed system with respect to Metric Temporal Logic (MTL) specifications.
- Introduce a progression-based approach with probabilistic guarantee.
Overview

- A fault-proof central monitor verifying a partially-synchronous distributed system with respect to Metric Temporal Logic (MTL) specifications.
- Introduce a progression-based approach with probabilistic guarantee.
- Implemented using SMT solvers.
Metric Temporal Logic

Finite Trace Example
Metric Temporal Logic
Finite Trace Example

\( \sigma_1 \models \Diamond_{[1,4]} b \)
Metric Temporal Logic
Finite Trace Example

\( \sigma_1 \models \Diamond_{[1,4]} b \)
Metric Temporal Logic

Finite Trace Example

\[ \sigma_1 \models \Diamond_{[1,4]} b \]
Metric Temporal Logic

Finite Trace Example

\[ \sigma_1 \models \Diamond_{[1,4]} b \quad \top \]
Metric Temporal Logic
Finite Trace Example

(1, \neg b) \rightarrow (2, \neg b) \rightarrow (3, b) \rightarrow (4, \neg b) \rightarrow (5, \neg b)

\sigma_1 \models \lozenge_{[1,4]} b \quad \top

\sigma_1 \models \square_{[2,5]} a \quad ?
Metric Temporal Logic

Finite Trace Example

$\sigma_1 \models \Diamond_{[1,4]} b \quad \top$

$\sigma_1 \models \Box_{[2,5]} a \quad ?$
Metric Temporal Logic

Finite Trace Example

\( \sigma_1 \models \Diamond_{[1,4]} b \quad \top \)

\( \sigma_1 \not\models \Box_{[2,5]} a \quad ? \)
Metric Temporal Logic
Finite Trace Example

$\sigma_1 \models \Diamond_{[1,4]} b \quad \top$

$\sigma_1 \models \Box_{[2,5]} a \quad ?$
Metric Temporal Logic
Finite Trace Example

\( \sigma_1 \models \Diamond_{[1,4]} b \quad \top \)

\( \sigma_1 \models [2,5] a \quad \top \)
Metric Temporal Logic

Finite Trace Example

\( \sigma_1 \models \Diamond_{[1,4)} b \quad \top \)

\( \sigma_1 \models \Box_{[2,5)} a \quad \top \)

\( \sigma_1 \models a \mathcal{U}_{[2,4)} b \quad ? \)
Metric Temporal Logic
Finite Trace Example

(1, ¬b) → (2, ¬b) → (3, b) → (4, ¬b) → (5, ¬b)
σ_1 ⊨ ◊_{[1,4]} b \; \top

(1, ¬a) → (2, a) → (3, a) → (4, a) → (5, ¬a)
σ_1 ⊨ □_{[2,5]} a \; \top

(1, a) → (2, a)
σ_1 ⊨ a \mathcal{U}_{[2,4]} b \; ?
Metric Temporal Logic

Finite Trace Example

\( \sigma_1 \models \Diamond_{[1,4]} b \quad \top \)

\( \sigma_1 \models \Box_{[2,5]} a \quad \top \)

\( \sigma_1 \models a \mathcal{U}_{[2,4]} b \quad \top \)
Metric Temporal Logic

Finite Trace Example

\[ \sigma_1 \models \Diamond_{(1,4)} b \quad \top \]

\[ \sigma_1 \models \Box_{(2,5)} a \quad \top \]

\[ \sigma_1 \models a \mathcal{U}_{(2,4)} b \quad \top \]
Proposed Solution
Progression-based for MTL Specifications

$\neg p \ \mathcal{U}_{[4,6]} (\Diamond_{[1,5]} r)$
Proposed Solution
Progression-based for MTL Specifications

\[ \neg p \cup_{[4,6)} (\Diamond_{[1,5]} r) \]
Proposed Solution
Progression-based for MTL Specifications

$\neg p \mathcal{U}_{[4,6)} (\Diamond_{[1,5]} r)$
Proposed Solution
Progression-based for MTL Specifications

\[\neg p \cup_{[4,6]} (\Diamond_{[1,5]} r)\]

\[\Diamond_{[4,5]} (\Diamond_{[1,5]} r)\]
Proposed Solution
Progression-based for MTL Specifications

\[ \neg p \cup_{[4,6]} (\Diamond_{[1,5]} r) \]

\[ \square_{[0,4]} \neg p \]
\[ \Diamond_{[4,5]}(\Diamond_{[1,5]} r) \]

\[ \square_{[0,5]} \neg p \]
\[ \Diamond_{[5,6]}(\Diamond_{[1,5]} r) \]
Proposed Solution
Progression-based for MTL Specifications

\(\neg p \cup [4,6) (\lozenge [1,5) r)\)

\(\Box [0,4) \neg p\)

\(\Box [0,5) \neg p\)

\(\lozenge [4,5) (\lozenge [1,5) r)\)

\(\lozenge [5,6) (\lozenge [1,5) r)\)
Proposed Solution
Progression-based for MTL Specifications

\[ \neg p \ \mathcal{U}_{[4,6)} (\Diamond_{[1,5]} r) \]
\[ \square_{[0,4]} \neg p \quad \top \]
\[ \Diamond_{[4,5]} (\Diamond_{[1,5]} r) \rightarrow \Diamond_{[0,4]} r \]
\[ \Diamond_{[5,6]} (\Diamond_{[1,5]} r) \]
Proposed Solution
Progression-based for MTL Specifications

\[ \neg p \cup_{[4,6]} (\lozenge_{[1,5]} r) \]

\[ \square_{[0,4]} \neg p \quad \top \quad \lozenge_{[4,5]}(\lozenge_{[1,5]} r) \leftarrow \lozenge_{[0,4]} r \]

\[ \square_{[0,5]} \neg p \quad \top \quad \lozenge_{[5,6]}(\lozenge_{[1,5]} r) \rightarrow \lozenge_{[1,5]} r \]
Proposed Solution

Progression-based for MTL Specifications

\( \neg p \cup [4,6) \text{ (}[1,5)r \)

\( \Box_{[0,4]} \neg p \quad \top \quad \Diamond_{[4,5)}(\Diamond_{[1,5]}r) \quad \Diamond_{[0,4]}r \)

\( \Box_{[0,5]} \neg p \quad \top \quad \Diamond_{[5,6)}(\Diamond_{[1,5]}r) \quad \Diamond_{[1,5]}r \)

\( \neg p \cup [0,2) \text{ (}[1,5)r \)

\( \neg p \cup [0,1) \text{ (}[1,5)r \)
Proposed Solution

Progression-based for MTL Specifications

$$\neg p \mathcal{U}_{[4,6)} (\Diamond_{[1,5]r})$$

$$\Box_{[0,4]} \neg p \quad \top$$  $$\Diamond_{[4,5]}(\Diamond_{[1,5]r}) \quad \Diamond_{[0,4]}r \quad 0.4375$$

$$\Box_{[0,5]} \neg p \quad \top$$  $$\Diamond_{[5,6]}(\Diamond_{[1,5]r}) \quad \Diamond_{[1,5]}r \quad 0.5625$$

$$\neg p \mathcal{U}_{[0,2)} (\Diamond_{[1,5]r}) \quad 0.5625$$

$$\neg p \mathcal{U}_{[0,1)} (\Diamond_{[1,5]r}) \quad 0.4375$$
Proposed Solution
Progression-based for MTL Specifications

$\neg p \mathcal{U}_{[0,2]} (\Diamond_{[1,5]} r)$  0.5625
$\neg p \mathcal{U}_{[0,1]} (\Diamond_{[1,5]} r)$  0.4375

$\Diamond_{[0,4]} r$  0.4375
$\Diamond_{[1,5]} r$  0.5625
Proposed Solution
Progression-based for MTL Specifications

\[ \Diamond_{[0,4]} r = 0.4375 \]
\[ \Diamond_{[1,5]} r = 0.5625 \]
\[ \neg p \mathcal{U}_{[0,2]} (\Diamond_{[1,5]} r) = 0.5625 \]
\[ \neg p \mathcal{U}_{[0,1]} (\Diamond_{[1,5]} r) = 0.4375 \]
Proposed Solution
Progression-based for MTL Specifications

\[ (1, \neg p \land \neg r) \]

\[ (2, \neg p \land \neg r) \]

\[ (3, \neg p \land \neg r) \]

\[ (4, \neg p \land \neg r) \]

\[ \Diamond_{[0,4]} r \quad 0.4375 \]

\[ \Diamond_{[1,5]} r \quad 0.5625 \]

\[ \neg p \cup_{[0,2]} (\Diamond_{[1,5]} r) \quad 0.5625 \]

\[ \neg p \cup_{[0,1]} (\Diamond_{[1,5]} r) \quad 0.4375 \]
Proposed Solution
Progression-based for MTL Specifications

\[
\begin{align*}
\Diamond_{[0,4)} r &= 0.4375 \\
\Diamond_{[1,5)} r &= 0.5625 \\
\neg p \cup_{[0,2)} (\Diamond_{[1,5)} r) &= 0.5625 \\
\neg p \cup_{[0,1)} (\Diamond_{[1,5)} r) &= 0.4375
\end{align*}
\]
Proposed Solution
Progression-based for MTL Specifications

\[ \Diamond_{[0,4]} r = 0.4375 \]
\[ \Diamond_{[1,5]} r = 0.5625 \]
\[ \neg p \cup_{[0,2]} (\Diamond_{[1,5]} r) = 0.5625 \]
\[ \neg p \cup_{[0,1]} (\Diamond_{[1,5]} r) = 0.4375 \]
Proposed Solution
Progression-based for MTL Specifications

\( \Diamond_{[0,4]}^{r} \) 0.4375
\( \Diamond_{[1,5]}^{r} \) 0.5625
\( \neg p \ \mathcal{U}_{[0,2]} (\Diamond_{[1,5]}^{r}) \) 0.5625  false 0.53125
\( \neg p \ \mathcal{U}_{[0,1]} (\Diamond_{[1,5]}^{r}) \) 0.4375
Results
Results
Results
Results
## Contributions

<table>
<thead>
<tr>
<th></th>
<th>Title</th>
<th>Authors</th>
<th>Conference/Publication</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Monitoring Distributed System under Partial Synchrony</td>
<td>Ritam Ganguly, Anik Momtaz, Borzoo Bonakdarpour</td>
<td>OPODIS 2020</td>
</tr>
<tr>
<td>2</td>
<td>Runtime Verification of Partially-Synchronous Distributed System</td>
<td>Ritam Ganguly, Anik Momtaz, Borzoo Bonakdarpour</td>
<td>FMSD (minor revision)</td>
</tr>
<tr>
<td></td>
<td>Distributed Runtime Verification of Metric Temporal Properties for Cross-Chain Protocols</td>
<td>Ritam Ganguly, Yingjie Xue, Aaron Jonckheere, Parker Ljung, Benjamin Schornstein, Borzoo Bonakdarpour, Maurice Herlihy</td>
<td>ICDCS 2022</td>
</tr>
<tr>
<td></td>
<td>Distributed Runtime Verification of Metric Temporal Properties</td>
<td>Ritam Ganguly, Yingjie Xue, Aaron Jonckheere, Parker Ljung, Benjamin Schornstein, Borzoo Bonakdarpour, Maurice Herlihy</td>
<td>JPDC (in review)</td>
</tr>
<tr>
<td>3</td>
<td>Crash-Resilient Decentralized Synchronous Runtime Verification</td>
<td>Ritam Ganguly, Shokufeh Kazemloo, Borzoo Bonakdarpour</td>
<td>TDSC</td>
</tr>
<tr>
<td>4</td>
<td>Decentralized Runtime Verification of Stream-based Partially-Synchronous Distributed System</td>
<td>Ritam Ganguly, Borzoo Bonakdarpour</td>
<td>EMSOFT 2023 (in review)</td>
</tr>
</tbody>
</table>
Overview

• Runtime verification of synchronous distributed systems, where a set of decentralized monitors only have a partial view of the system.

• The monitors are subject to crash faults.

• Introduce an SMT-based monitor transformation algorithm to deal with the inconsistency in the computed automata state.
Decentralized Solution

\[ (a \land b) \]

\[ \emptyset \]

\[ a \rightarrow b \rightarrow \emptyset \rightarrow ab \]
Decentralized Solution

- Monitor 1
  - Reads: Computes:
- Monitor 2
  - Reads: Computes:
- Monitor 3
  - Reads: Computes:

\[ \diamond (a \land b) \]
Decentralized Solution

- Monitor 1
  - Reads: 
  - Computes: 

- Monitor 2
  - Reads: 
  - Computes: 

- Monitor 3
  - Reads: 
  - Computes: 

\( q_0 \)

\( q_\top \)

\( \diamond (a \land b) \)
Decentralized Solution

- Monitor 1
  - Reads: 
  - Computes:
- Monitor 2
  - Reads: 
  - Computes:
- Monitor 3
  - Reads: 
  - Computes:
Decentralized Solution

- Monitor 1
  - Reads: \{a : \text{true}, b : \top\} Computes: \{q_0, q_{\top}\}
- Monitor 2
  - Reads: \{a : \bot, b : \top\} Computes: \{q_0, q_{\top}\}
- Monitor 3
  - Reads: \{a : \bot, b : \text{false}\} Computes: \{q_0\}

\[ (a \land b) \]
Decentralized Solution

- Monitor 1
  - Reads:
  - Computes:

- Monitor 2
  - Reads:
  - Computes:

- Monitor 3
  - Reads:
  - Computes:
Decentralized Solution

- Monitor 1
  - Reads: COMPUTES: 
- Monitor 2
  - Reads: COMPUTES: 
- Monitor 3
  - Reads: COMPUTES: 

\[ \diamond (a \land b) \]
Decentralized Solution

\[ (a \land b) \]

- Monitor 1
  - Reads: \{a : false, b : \top\} Computes: \{q_0\}

- Monitor 2
  - Reads: \{a : \top, b : \top\} Computes: \{q_0, q_{\top}\}

- Monitor 3
  - Reads: \{a : \top, b : true\} Computes: \{q_0, q_{\top}\}
Decentralized Solution

• Monitor 1
  • Reads:  Computes:
• Monitor 2
  • Reads:  Computes:
• Monitor 3
  • Reads:  Computes:
Decentralized Solution

- Monitor 1
  - Reads:  
  - Computes:  
- Monitor 2
  - Reads:  
  - Computes:  
- Monitor 3
  - Reads:  
  - Computes:  

\[(a \land b)\]
Decentralized Solution

- Monitor 1
  - Reads: \( \{ a : \text{false}, b : \top \} \)
  - Computes: \( \{ q_0 \} \)

- Monitor 2
  - Reads: \( \{ a : \top, b : \top \} \)
  - Computes: \( \{ q_0, q_\top \} \)

- Monitor 3
  - Reads: \( \{ a : \top, b : \text{false} \} \)
  - Computes: \( \{ q_0 \} \)
Decentralized Solution

\[ (a \land b) \]

- Monitor 1
  - Reads: 
  - Computes:
- Monitor 2
  - Reads: 
  - Computes:
- Monitor 3
  - Reads: 
  - Computes:
Decentralized Solution

- Monitor 1
  - Reads: Computes:
- Monitor 2
  - Reads: Computes:
- Monitor 3
  - Reads: Computes:
Decentralized Solution

- Monitor 1
  - Reads: \( \{ a : \text{true}, b : \top \} \)
  - Computes: \( \{ q_0, q_\top \} \)

- Monitor 2
  - Reads: \( \{ a : \top, b : \top \} \)
  - Computes: \( \{ q_0, q_\top \} \)

- Monitor 3
  - Reads: \( \{ a : \top, b : \text{true} \} \)
  - Computes: \( \{ q_0, q_\top \} \)
Decentralized Solution

- Monitor 1
  - Reads: \( \{a : \text{true}, b : \top\} \)
  - Computes: \( \{q_0, q_{\top}\} \)

- Monitor 2
  - Reads: \( \{a : \top, b : \top\} \)
  - Computes: \( \{q_0, q_{\top}\} \)

- Monitor 3
  - Reads: \( \{a : \top, b : \text{true}\} \)
  - Computes: \( \{q_0, q_{\top}\} \)
Decentralized Solution

- Monitor 1
  - Reads: \(\{a : \text{true}, b : \neg\}\) Computes: \(\{q_0, q_\top\}\)

- Monitor 2
  - Reads: \(\{a : \neg, b : \neg\}\) Computes: \(\{q_0, q_\top\}\)

- Monitor 3
  - Reads: \(\{a : \neg, b : \text{true}\}\) Computes: \(\{q_0, q_\top\}\)

Inconsistent
Monitor Transformation

$q_0 \xrightarrow{\{a\}, \{b\}, \emptyset} q_0$

$q_0 \xrightarrow{\{a, b\}} q_T$

$\diamond (a \land b)$
Monitor Transformation

\[ q_0 \xrightarrow{(a, b)} q_T \]

\[ \{a\}, \{b\}, \emptyset \]

\[ \diamond (a \land b) \]

\[ q_0^1 \xrightarrow{} q_0^2 \]
Monitor Transformation

\[ (a \land b) \]

\[
\begin{align*}
q_0 & \rightarrow \{a\}, \{b\}, \emptyset \\
q_T & \rightarrow \{a, b\} \\
\end{align*}
\]

\[
\begin{align*}
q_0^1 & \rightarrow \{a\}, \emptyset \\
q_0^2 & \rightarrow \{b\} \\
\end{align*}
\]
Monitor Transformation

\[ \diamond (a \land b) \]
Proposed Solution

1-crash resistant

• Monitor 1
  • Reads:            Computes:
• Monitor 2
  • Reads:            Computes:
• Monitor 3
  • Reads:            Computes:
Proposed Solution

1-crash resistant

- Monitor 1
  - Reads: Computes:
- Monitor 2
  - Reads: Computes:
- Monitor 3
  - Reads: Computes:
Proposed Solution

1-crash resistant

- Monitor 1
  - Reads: $a$
  - Computes: $b \rightarrow \emptyset \rightarrow ab$

- Monitor 2
  - Reads: $a$
  - Computes: $a \rightarrow b$

- Monitor 3
  - Reads: $b$
  - Computes: $a \rightarrow a \land b$

$\diamond (a \land b)$
Proposed Solution

1-crash resistant

- Monitor 1
  - Reads: \{a : \text{true}, b : \top\} Computes: \{q_0^1, q_\top\}
- Monitor 2
  - Reads: \{a : \bot, b : \top\} Computes: \{q_0^1, q_0^2, q_\top\}
- Monitor 3
  - Reads: \{a : \bot, b : \text{false}\} Computes: \{q_0^1\}

\[ \diamondsuit (a \land b) \]
Proposed Solution

• Monitor 1
  • Reads:  
  • Computes: 

• Monitor 2
  • Reads:  
  • Computes: 

• Monitor 3
  • Reads:  
  • Computes:
Proposed Solution

1-crash resistant

- Monitor 1
  - Reads: Computes:
- Monitor 2
  - Reads: Computes:
- Monitor 3
  - Reads: Computes:

\( a \xrightarrow{} b \xrightarrow{} \emptyset \xrightarrow{} ab \)
Proposed Solution

1-crash resistant

- Monitor 1
  - Reads: \{a : \text{false}, b : \top\}
  - Computes: \{q_0^1, q_0^2\}

- Monitor 2
  - Reads: \{a : \top, b : \top\}
  - Computes: \{q_0^1, q_0^2, q_\top\}

- Monitor 3
  - Reads: \{a : \top, b : \text{true}\}
  - Computes: \{q_0^2, q_\top\}
Proposed Solution

1-crash resistant

\[ a \rightarrow b \rightarrow \emptyset \rightarrow ab \]

- **Monitor 1**
  - Reads: \( \{ a : \text{false}, b : \top \} \)
  - Computes: \( \{ q_0^1, q_0^2 \} \)

- **Monitor 2**
  - Reads: \( \{ a : \top, b : \top \} \)
  - Computes: \( \{ q_0^1, q_0^2, q_\top \} \)

- **Monitor 3**
  - Reads: \( \{ a : \top, b : \text{true} \} \)
  - Computes: \( \{ q_0^2, q_\top \} \)
Proposed Solution

1-crash resistant

• Monitor 1
  • Reads: Computes:
• Monitor 2
  • Reads: Computes:
• Monitor 3
  • Reads: Computes:

\[ a \rightarrow b \rightarrow \emptyset \rightarrow ab \]
Proposed Solution

1-crash resistant

- Monitor 1
  - Reads:  
  - Computes: 
- Monitor 2
  - Reads:  
  - Computes: 
- Monitor 3
  - Reads:  
  - Computes:
Proposed Solution

1-crash resistant

- Monitor 1
  - Reads: \{a : false, b : \top\}
  - Computes: \{q_0^1, q_0^2\}

- Monitor 2
  - Reads: \{a : \top, b : \top\}
  - Computes: \{q_0^1, q_0^2, q_\top\}

- Monitor 3
  - Reads: \{a : \top, b : false\}
  - Computes: \{q_0^1\}
Proposed Solution

1-crash resistant

- Monitor 1
  - Reads: \( \{a : \text{false}, b : \top\} \)
  - Computes:

- Monitor 2
  - Reads: \( \{a : \top, b : \top\} \)
  - Computes:

- Monitor 3
  - Reads: \( \{a : \top, b : \text{false}\} \)
  - Computes:
Proposed Solution

1-crash resistant

• Monitor 1
  • Reads: \{a : \text{false}, b : \top\} Computes: \{q_0^1, q_0^2\}

• Monitor 2
  • Reads: \{a : \top, b : \top\} Computes: \{q_0^1\}

• Monitor 3
  • Reads: \{a : \top, b : \text{false}\} Computes: \{q_0^1\}
 Proposed Solution

1-crash resistant

• Monitor 1
  • Reads: \{a: \text{false}, b: \top\} Computes: \{q_0^1, q_0^2\}

• Monitor 2
  • Reads: \{a: \top, b: \top\} Computes: \{q_0^1\}

• Monitor 3
  • Reads: \{a: \top, b: \text{false}\} Computes: \{q_0^1\}

Crashed

\[\Diamond (a \land b)\]
Proposed Solution

1-crash resistant

• Monitor 1
  • Reads: \{a : false, b : \top\} Computes:

• Monitor 2
  • Reads: \{a : \top, b : \top\} Computes:

• Monitor 3
  • Reads: \{a : \top, b : false\} Computes: \{q_0^1\}

Crashed

\( \diamond (a \land b) \)
Proposed Solution

1-crash resistant

Monitor 1
• Reads: \( \{a: \text{false}, b: \top\} \) Computes: \( \{q_0^1\} \)

Monitor 2
• Reads: \( \{a: \top, b: \top\} \) Computes: \( \{q_0^1\} \)

Monitor 3
• Reads: \( \{a: \top, b: \text{false}\} \) Computes: \( \{q_0^1\} \)

Crashed
• Reads: \( \{a: \top, b: \text{false}\} \) Computes: \( \{q_0^1\} \)

\( (a \land b) \)
**Proposed Solution**

1-crash resistant

- Monitor 1
  - Reads: \(\{a : \text{false}, b : \top\}\)
  - Computes: \(\{q_0^1\}\)

- Monitor 2
  - Reads: \(\{a : \top, b : \top\}\)
  - Computes: \(\{q_0^1\}\)

- Monitor 3
  - Reads: \(\{a : \top, b : \text{false}\}\)
  - Computes: \(\{q_0^1\}\)

Crashed: \(\{q_1\}^0\)
Proposed Solution

1-crash resistant

- **Monitor 1**
  - Reads: 
  - Computes:

- **Monitor 2**
  - Reads: 
  - Computes:

- **Monitor 3**
  - Reads: \( \{a : \bot, b : \text{false}\} \)
  - Computes: \( \{q_0^1\} \)

\[
\begin{align*}
&\{a, \emptyset\} & \to \{b\} \\
&\{a\} & \to \{a\} & \to \emptyset & \to ab
\end{align*}
\]
Proposed Solution

1-crash resistant

- Monitor 1
  - Reads: 
  - Computes: 

- Monitor 2
  - Reads: 
  - Computes: 

- Monitor 3
  - Reads: \(\{a : \exists, b : \text{false}\}\) 
  - Computes: \(\{q_0^1\}\)

Crashed

\[\text{Crashed}\]
Proposed Solution

1-crash resistant

• Monitor 1
  • Reads: \{a : \text{true}, b : \top\} Computes: \{q_0^1, q_T\}

• Monitor 2
  • Reads: \{a : \top, b : \text{true}\} Computes: \{q_0^2, q_T\}

• Monitor 3
  • Reads: \{a : \top, b : \text{false}\} Computes: \{q_0^1\}

\[ \text{Crashed} \]
Proposed Solution

1-crash resistant

- Monitor 1
  - Reads: \{a : \text{true}, b : \text{\triangledown}\} Computes: \{q_0^1, q_T\}
- Monitor 2
  - Reads: \{a : \text{\triangledown}, b : \text{true}\} Computes: \{q_0^2, q_T\}
- Monitor 3
  - Reads: \{a : \text{\triangledown}, b : \text{false}\} Computes: \{q_0^1\}

Crashed

\diamond (a \land b)
Results
Results
Results
Results
Contributions

1. Monitoring Distributed System under Partial Synchrony
   Ritam Ganguly, Anik Momtaz, Borzoo Bonakdarpour; OPODIS 2020

2. Runtime Verification of Partially-Synchronous Distributed System
   Ritam Ganguly, Anik Momtaz, Borzoo Bonakdarpour; FMSD (minor revision)

3. Distributed Runtime Verification of Metric Temporal Properties for Cross-Chain Protocols
   Ritam Ganguly, Yingjie Xue, Aaron Jonckheere, Parker Ljung, Benjamin Schornstein, Borzoo Bonakdarpour, Maurice Herlihy; ICDCS 2022

4. Distributed Runtime Verification of Metric Temporal Properties
   Ritam Ganguly, Yingjie Xue, Aaron Jonckheere, Parker Ljung, Benjamin Schornstein, Borzoo Bonakdarpour, Maurice Herlihy; JPDC (in review)

5. Crash-Resilient Decentralized Synchronous Runtime Verification
   Ritam Ganguly, Shokufeh Kazemloo, Borzoo Bonakdarpour, TDSC

6. Decentralized Runtime Verification of Stream-based Partially-Synchronous Distributed System
   Ritam Ganguly, Borzoo Bonakdarpour, EMSOFT 2023 (in review)
Overview

- Stream-based runtime verification technique for partially-synchronous distributed system where a set of decentralized monitors only has a partial view of the system.
- Introduce partially-synchronous semantics for stream-based specification language LOLA.
Stream-based Specification
LOLA

<table>
<thead>
<tr>
<th></th>
<th>3</th>
<th>5</th>
<th>6</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>3</th>
<th>5</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>y</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>x + y</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>x &gt; y</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Stream-based Specification
LOLA

<table>
<thead>
<tr>
<th>x</th>
<th>3</th>
<th>5</th>
<th>6</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>y</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td>x + y</td>
<td>4</td>
<td>8</td>
<td>11</td>
<td>16</td>
</tr>
<tr>
<td>x &gt; y</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Stream-based Specification

LOLA

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>3</td>
<td>5</td>
<td>6</td>
<td>9</td>
</tr>
<tr>
<td>y</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>7</td>
</tr>
</tbody>
</table>

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>x + y</td>
<td>4</td>
<td>8</td>
<td>11</td>
<td>16</td>
</tr>
</tbody>
</table>

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>x &gt; y</td>
<td>true</td>
<td>true</td>
<td>true</td>
<td>true</td>
</tr>
</tbody>
</table>
Stream-based Specification

LOLA

<table>
<thead>
<tr>
<th>x</th>
<th>3</th>
<th>5</th>
<th>6</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>y</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td>x + y</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>x &gt; y</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Stream-based Specification

LOLA

\[
x \quad 3 \quad \epsilon \quad 5 \quad \epsilon \quad 6 \quad \epsilon \quad 9
\]

\[
y \quad 1 \quad \epsilon \quad 3 \quad \epsilon \quad 5 \quad \epsilon \quad 7
\]

\[
x + y \quad \epsilon \quad \epsilon \quad \epsilon \quad \epsilon \quad \epsilon \quad \epsilon \quad \epsilon
\]

\[
x > y \quad \epsilon \quad \epsilon \quad \epsilon \quad \epsilon \quad \epsilon \quad \epsilon \quad \epsilon
\]
Stream-based Specification

LOLA

\[ x \quad 3 \quad 5 \quad 6 \quad 9 \]

\[ y \quad 1 \quad 3 \quad 5 \quad 7 \]

\[ x + y \]

\[ x > y \]
Stream-based Specification

LOLA

\[
\begin{align*}
x & \quad 3 & \quad 5 & \quad 6 & \quad 9 \\
y & \quad 1 & \quad 3 & \quad 5 & \quad 7 \\
x + y & \quad \{4\} & \quad \{4,6,8\} & \quad \{8\} & \quad \{8,9,10,11\} & \quad \{11\} & \quad \{16\} & \quad \{11,13,14,16\} \\
x > y & \quad \quad & \quad \quad & \quad \quad & \quad \quad & \quad \quad & \quad \quad & \quad \quad
\end{align*}
\]
**Stream-based Specification**

**LOLA**

<table>
<thead>
<tr>
<th>x</th>
<th>y</th>
<th>x + y</th>
<th>x &gt; y</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>1</td>
<td>{4}</td>
<td>true</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>{4,6,8}</td>
<td>true</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
<td>{8}</td>
<td>true</td>
</tr>
<tr>
<td>9</td>
<td>7</td>
<td>{8,9,10,11}</td>
<td>true</td>
</tr>
<tr>
<td>1</td>
<td>11</td>
<td>{11}</td>
<td>true</td>
</tr>
<tr>
<td>3</td>
<td>13</td>
<td>{11,13,14,16}</td>
<td>true</td>
</tr>
<tr>
<td>5</td>
<td>14</td>
<td>{16}</td>
<td>true</td>
</tr>
<tr>
<td>6</td>
<td>16</td>
<td>{11}</td>
<td>true</td>
</tr>
<tr>
<td>9</td>
<td>16</td>
<td>{true}</td>
<td>true</td>
</tr>
</tbody>
</table>

{true, false}
Proposed Solution

Decentralized monitoring of stream-based specification

- Monitor 1
  - Read: 37

- Monitor 2
  - Read:
Proposed Solution
Decentralized monitoring of stream-based specification

$c := \text{ite}(a[−1,0] \leq b[1,0], a[1,0], b[−1,0])$

- Monitor 1
  - Read:

- Monitor 2
  - Read:
Proposed Solution

Decentralized monitoring of stream-based specification

\[ c := \text{ite}(a[1,0] \leq b[1,0], a[1,0], b[1,0]) \]

- Monitor 1
  - Read:

- Monitor 2
  - Read:
Proposed Solution

Decentralized monitoring of stream-based specification

\[ c := \text{ite}(a[-1,0] \leq b[1,0], a[1,0], b[-1,0]) \]

- Monitor 1
  - Read: \{a, (1,1), (3,5)\}, \{b, (2,5), (3,9)\}

- Monitor 2
  - Read: \{a, (1,1), (2,7)\}, \{b, (1,3), (3,9)\}
Proposed Solution
Decentralized monitoring of stream-based specification

\[ c := \text{ite}(a[-1,0] \leq b[1,0], a[1,0], b[-1,0]) \]

- Monitor 1
  - Read: \{a, (1,1), (3,5), b, (2,5), (3,9)\}
  
  \[ \text{LS}_1^1 = \{c(1) = a(2), c(2) = 5, c(3) = \text{ite}(a(2) \leq b(4), a(4), 5)\} \]

- Monitor 2
  - Read: \{a, (1,1), (2,7), b, (1,3), (3,9)\}
  
  \[ \text{LS}_1^2 = \{c(1) = \text{ite}(0 \leq b(2), 7, 0), c(2) = a(3), c(3) = \text{ite}(7 \leq b(4), a(4), b(2))\} \]
Proposed Solution

Decentralized monitoring of stream-based specification

\[ c := \text{ite}(a[-1,0] \leq b[1,0], a[1,0], b[-1,0]) \]

- **Monitor 1**
  - **Read:** \( \{a, (1,1), (3,5)\}, \{b, (2,5), (3,9)\} \)
  
  \[ \text{LS}_1^1 = \{ \text{c}(1) = \text{a}(2), \text{c}(2) = 5, \text{c}(3) = \text{ite}(a(2) \leq b(4), a(4), 5) \} \]

  \[ \Pi_1^1 = \{ \text{c}(1) = 7, \text{c}(2) = 5, \text{c}(3) = \text{ite}(7 \leq b(4), a(4), 5) \} \]

- **Monitor 2**
  - **Read:** \( \{a, (1,1), (2,7)\}, \{b, (1,3), (3,9)\} \)
  
  \[ \text{LS}_1^2 = \{ \text{c}(1) = \text{ite}(0 \leq b(2), 7, 0), \text{c}(2) = \text{a}(3), \text{c}(3) = \text{ite}(7 \leq b(4), a(4), b(2)) \} \]

\[ \Pi_1^2 = \{ \text{c}(1) = 7, \text{c}(2) = 5, \text{c}(3) = \text{ite}(7 \leq b(4), a(4), 5) \} \]
Proposed Solution

Decentralized monitoring of stream-based specification

- Monitor 1
  - Read:

- Monitor 2
  - Read:

\[
P_1 = \{ c(1) = 7, c(2) = 5, c(3) = \text{ite} (7 \leq b(4), a(4), 5) \}\]
Proposed Solution

Decentralized monitoring of stream-based specification

$c := \text{ite}(a[-1,0] \leq b[1,0], a[1,0], b[-1,0])$

• Monitor 1
  • Read:

  \[ \Pi_1^1 = \{c(1) = 7, c(2) = 5, c(3) = \text{ite}(7 \leq b(4), a(4), 5)\} \]

• Monitor 2
  • Read:
Proposed Solution

Decentralized monitoring of stream-based specification

\[ c := \text{ite}(a[-1,0] \leq b[1,0], a[1,0], b[-1,0]) \]

- Monitor 1
  - Read: \{a, (4,4), (5,4)\}, \{b, (4,3), (6,1)\}

- Monitor 2
  - Read: \{a, (5,4), (6,7)\}, \{b, (4,3), (5,5)\}

\[ \Pi_1 = \{c(1) = 7, c(2) = 5, c(3) = \text{ite}(7 \leq b(4), a(4), 5)\} \]
Proposed Solution

Decentralized monitoring of stream-based specification

• Monitor 1
  • Read: \{a, (4,4), (5,4)\}, \{b, (4,3), (6,1)\}
  \(LS_1^1 = \{c(4) = 9, c(5) = 3, c(6) = b(5)\}\)

• Monitor 2
  • Read: \{a, (5,4), (6,7)\}, \{b, (4,3), (5,5)\}
  \(LS_2^2 = \{c(4) = \text{ite}(a(3) \leq 5, 4, 9), c(5) = \text{ite}(a(4) \leq b(6), 7, 3), c(6) = 5\}\)

\[
\begin{align*}
  \Pi_1^1 &= \{c(1) = 7, c(2) = 5, c(3) = \text{ite}(7 \leq b(4), a(4), 5)\} \\
  c &= \text{ite}(a[-1,0] \leq b[1,0], a[1,0], b[-1,0])
\end{align*}
\]
Proposed Solution
Decentralized monitoring of stream-based specification

- Monitor 1
  - Read: \( \{a, (4,4), (5,4)\}, \{b, (4,3), (6,1)\} \)
  \[
  \text{LS}_2^1 = \{ c(4) = 9, c(5) = 3, c(6) = b(5) \}
  \]

- Monitor 2
  - Read: \( \{a, (5,4), (6,7)\}, \{b, (4,3), (5,5)\} \)
  \[
  \text{LS}_2^2 = \{ c(4) = \text{ite}(a(3) \leq 5, 4, 9)), c(5) = \text{ite}(a(4) \leq b(6), 7, 3), c(6) = 5 \}
  \]

\[
c := \text{ite}(a[-1,0] \leq b[1,0], a[1,0], b[-1,0])
\]
Proposed Solution
Decentralized monitoring of stream-based specification

- Monitor 1
  - Read: \{a, (4,4), (5,4), b, (4,3), (6,1)\}
  
  $\text{LS}_1^1 = \{ c(4) = 9, c(5) = 3, c(6) = b(5) \}$

- Monitor 2
  - Read: \{a, (5,4), (6,7), b, (4,3), (5,5)\}
  
  $\text{LS}_2^2 = \{ c(4) = \text{ite}(a(3) \leq 5, 4, 9), c(5) = \text{ite}(a(4) \leq b(6), 7, 3), c(6) = 5 \}$

\[ \text{Π}_2^1 = \{ c(3) = 5, c(4) = 9, c(5) = 3, c(6) = 5 \} \]
Results
Results
Results
Results
Outline of the talk

1. Motivation and Preliminaries
2. Contributions
   1. Centralized runtime verification w.r.t. LTL specifications
   2. Centralized runtime verification w.r.t. MTL specifications
   3. Decentralized runtime verification w.r.t. LTL specifications
   4. Decentralized runtime verification w.r.t. stream-based specifications
3. Conclusion and Future Work
Summary

• Introduced an automata-based and a progression-based RV technique for LTL specifications.
Summary

• Introduced an automata-based and a progression-based RV technique for LTL specifications.

• Introduced a progression-based RV technique for MTL specifications with probabilistic guarantee.
Summary

- Introduced an automata-based and a progression-based RV technique for LTL specifications.
- Introduced a progression-based RV technique for MTL specifications with probabilistic guarantee.
- Introduced a fault-tolerant decentralized RV technique for LTL specifications where the monitors only have a partial view.
Summary

• Introduced an automata-based and a progression-based RV technique for LTL specifications.

• Introduced a progression-based RV technique for MTL specifications with probabilistic guarantee.

• Introduced a fault-tolerant decentralized RV technique for LTL specifications where the monitors only have a partial view.

• Introduced a decentralized partially-synchronous stream-based RV technique where the monitors only have a partial view.
Summary

• Introduced an automata-based and a progression-based RV technique for LTL specifications.
• Introduced a progression-based RV technique for MTL specifications with probabilistic guarantee.
• Introduced a fault-tolerant decentralized RV technique for LTL specifications where the monitors only have a partial view.
• Introduced a decentralized partially-synchronous stream-based RV technique where the monitors only have a partial view.
• Extensively studied all the approaches using both synthetic and real-life case studies.
Summary

• Introduced an automata-based and a progression-based RV technique for LTL specifications.
• Introduced a progression-based RV technique for MTL specifications with probabilistic guarantee.
• Introduced a fault-tolerant decentralized RV technique for LTL specifications where the monitors only have a partial view.
• Introduced a decentralized partially-synchronous stream-based RV technique where the monitors only have a partial view.
• Extensively studied all the approaches using both synthetic and real-life case studies.
Summary

• Introduced an automata-based and a progression-based RV technique for LTL specifications.

• Introduced a progression-based RV technique for MTL specifications with probabilistic guarantee.

• Introduced a fault-tolerant decentralized RV technique for LTL specifications where the monitors only have a partial view.

• Introduced a decentralized partially-synchronous stream-based RV technique where the monitors only have a partial view.

• Extensively studied all the approaches using both synthetic and real-life case studies.

Runtime Verification of a partially-synchronous distributed system in real-time is feasible.
Summary

• Introduced an automata-based and a progression-based RV technique for LTL specifications.
• Introduced a progression-based RV technique for MTL specifications with probabilistic guarantees.
• Introduced a fault-tolerant decentralized RV technique for LTL specifications where the monitors only have a partial view.
• Introduced a decentralized partially-synchronous stream-based RV technique where the monitors only have a partial view.
• Extensively studied all the approaches using both synthetic and real-life case studies.

Runtime Verification of a partially-synchronous distributed system in real-time is feasible.
Future Work
Future Work

• Distributed System
  • Byzantine faulty monitors
  • Predictive runtime verification
  • Runtime enforcement
Future Work

• Distributed System
  • Byzantine faulty monitors
  • Predictive runtime verification
  • Runtime enforcement
• AI-Safety
  • Monitoring of multi-agent systems
  • Enforcing properties
Thank you!
Thank you!
Thank you!

CCF 2118356
CCF 2102106
Thank you!
Thank you!
Thank you!
Future Work

- Distributed System
- Byzantine faulty monitors
- Predictive runtime verification
- Runtime enforcement
- AI-Safety
  - Monitoring of multi-agent systems
  - Enforcing properties
Proposed Solution
Decentralized monitoring of stream-based specification
Proposed Solution
Decentralized monitoring of stream-based specification

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>1</td>
<td>7</td>
<td>5</td>
</tr>
<tr>
<td>b</td>
<td>3</td>
<td>5</td>
<td>9</td>
</tr>
</tbody>
</table>

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>4</td>
<td>4</td>
<td>7</td>
</tr>
<tr>
<td>b</td>
<td>3</td>
<td>5</td>
<td>1</td>
</tr>
</tbody>
</table>
Proposed Solution
Decentralized monitoring of stream-based specification
Proposed Solution

Decentralized monitoring of stream-based specification
Proposed Solution
Decentralized monitoring of stream-based specification
Proposed Solution
Decentralized monitoring of stream-based specification
Proposed Solution

Decentralized monitoring of stream-based specification