Modular Specification and Compositional Analysis of Stochastic Systems

Benoît Delahaye

IRISA - S4

PHD Defense – 8/10/2010
Outline

1. Introduction

2. Specification Theories for Stochastic Systems
   - Interval Markov Chains
   - Constraint Markov Chains
   - Probabilistic Assume/Guarantee Contracts

3. Stochastic Abstraction and Analysis of Large Heterogeneous Systems

4. Conclusion and Future Work
Introduction
Introduction

Our goal:

- Specification and analysis of large and complex heterogeneous systems
Our goal:

- Specification and analysis of large and complex heterogeneous systems

Difficulties:
Our goal:

- Specification and analysis of large and complex heterogeneous systems

Difficulties:

- Size and number of components
Our goal:

- Specification and analysis of large and complex heterogeneous systems

Difficulties:

- Size and number of components
- Independability of their implementation
Our goal:

- Specification and analysis of large and complex heterogeneous systems

Difficulties:

- Size and number of components
- Independability of their implementation

Solutions:
Our goal:
- Specification and analysis of large and complex heterogeneous systems

Difficulties:
- Size and number of components
- Independability of their implementation

Solutions:
- Abstractions
Our goal:

- Specification and analysis of large and complex heterogeneous systems

Difficulties:

- Size and number of components
- Independability of their implementation

Solutions:

- Abstractions
- Component-based design
Two directions:
Two directions:

- At specification level
Two directions:

- *At specification level*
  - Abstraction for each component
Introduction

Two directions:

- At *specification* level
  - Abstraction for each component
  - Compositional theory
Two directions:

- **At specification level**
  - Abstraction for each component
  - Compositional theory

- **At execution level**
Two directions:

- **At specification level**
  - Abstraction for each component
  - Compositional theory

- **At execution level**
  - The system is already deployed
Introduction

Two directions:

- At specification level
  - Abstraction for each component
  - Compositional theory

- At execution level
  - The system is already deployed
  - Abstraction to reason on a group of components
A major difficulty: Reasoning on environments
A major difficulty: Reasoning on environments

- Open Systems: unspecified
A major difficulty: Reasoning on environments

- Open Systems: unspecified
- Closed Systems: may be too large
A major difficulty: Reasoning on environments

- Open Systems: unspecified
- Closed Systems: may be too large

Solution: Stochastic Modeling
A major difficulty: Reasoning on environments

- Open Systems: unspecified
- Closed Systems: may be too large

Solution: Stochastic Modeling

- Models possible failures
Introduction

A major difficulty: Reasoning on environments

- Open Systems: unspecified
- Closed Systems: may be too large

Solution: Stochastic Modeling

- Models possible failures
- May appear at different levels
A major difficulty: Reasoning on environments

- Open Systems: unspecified
- Closed Systems: may be too large

Solution: Stochastic Modeling

- Models possible failures
- May appear at different levels
  - At design level
A major difficulty: Reasoning on environments

- Open Systems: unspecified
- Closed Systems: may be too large

Solution: Stochastic Modeling

- Models possible failures
- May appear at different levels
  - At design level
  - During the analysis (closed system)
Introduction

A major difficulty: Reasoning on environments

- Open Systems: unspecified
- Closed Systems: may be too large

Solution: Stochastic Modeling

- Models possible failures
- May appear at different levels
  - At design level
  - During the analysis (closed system)
- Quantifies the likelihood of failures
Outline

1. Introduction

2. Specification Theories for Stochastic Systems
   - Interval Markov Chains
   - Constraint Markov Chains
   - Probabilistic Assume/Guarantee Contracts

3. Stochastic Abstraction and Analysis of Large Heterogeneous Systems

4. Conclusion and Future Work
Specification Theories:

- A specification (interface) allows to represent the behavior of multiple components at the design level
- Allows independent reasoning
- Supports component-based design of large systems
- Reduces complexity of the design
- Existing Theories: Modal Specifications, Interface Automata...
Specification Theories:

- A specification (interface) allows to represent the behavior of multiple components at the design level.
- Allows independent reasoning.
- Supports component-based design of large systems.
- Reduces complexity of the design.
- Existing Theories: Modal Specifications, Interface Automata...

Our contribution: The first complete theory for Markov Chains.
Specification Theories:

- A specification (interface) allows to represent the behavior of multiple components at the design level
- Allows independent reasoning
- Supports component-based design of large systems
- Reduces complexity of the design
- Existing Theories: Modal Specifications, Interface Automata...

Our contribution: The first complete theory for Markov Chains

But ... What is a complete theory?
Specification Theories: Implementation/comparison (refinement)
Specification Theories:
Implementation/comparison (refinement)

Specifications
Implementations
Specification Theories: Implementation/comparison (refinement)
Specification Theories:
Implementation/comparison (refinement)
Consistency

\[ S = \emptyset \quad \text{or} \quad \bullet S \]

?
Single Component Operators

Consistency

\[ S = \emptyset \quad \text{or} \quad \bullet \cdot \quad \mathcal{S} \]

Conjunction

\[ S_1 \quad \text{or} \quad S_2 \quad \text{or} \quad S_1 \cap S_2 \]

Benoît Delahaye (IRISA - S4)
Single Component Operators

Consistency

\[ S = \emptyset \quad \text{or} \quad \bullet \cdot S \quad ? \]

Conjunction

\[ S_1 \quad \text{or} \quad S_2 \quad ? \]

Refinement

\[ S_2 \quad \text{or} \quad S_1 \quad ? \]
Multiple Component Operators

Parallel Composition

Benoît Delahaye (IRISA - S4)
Outline

1. Introduction

2. Specification Theories for Stochastic Systems
   - Interval Markov Chains
   - Constraint Markov Chains
   - Probabilistic Assume/Guarantee Contracts

3. Stochastic Abstraction and Analysis of Large Heterogeneous Systems

4. Conclusion and Future Work
Markov Chains: An Example

\[ M = (\{1, \ldots, n\}, o, M, A, V) \]
$M = (\{1, \ldots, n\}, o, M, A, V)$
Markov Chains: An Example

\[ M = (\{1, \ldots, n\}, o, M, A, V) \]
Markov Chains: An Example

\[ M = (\{1, \ldots, n\}, o, M, A, V) \]
Markov Chains: An Example

\[ M = (\{1, \ldots, n\}, o, M, A, V) \]
$S = (\{1, \ldots, n\}, o, I, A, V)$

Idea: Replace exact probabilities with intervals
Interval Markov Chains: An Example

\[ S = (\{1, \ldots, n\}, o, I, A, V) \]
$S = (\{1, \ldots, n\}, o, I, A, V)$
Interval Markov Chains: An Example

\[ S = (\{1, \ldots, n\}, o, I, A, V) \]
Interval Markov Chains: An Example

\[ S = (\{1, \ldots, n\}, o, I, A, V) \]
$S = (\{1, \ldots, n\}, o, I, A, V)$
Our results concerning IMCs

- Consistency Checking: PTIME
Our results concerning IMCs

- Consistency Checking: \textbf{PTIME}
- Common Implementation: \textbf{EXPTIME}-complete
Our results concerning IMCs

- Consistency Checking: \textsc{PTIME}
- Common Implementation: \textsc{EXPTIME-complete}
- Semantic Refinement: \textsc{EXPTIME-complete}
Our results concerning IMCs

- Consistency Checking: \textit{PTIME}
- Common Implementation: \textit{EXPTIME-complete}
- Semantic Refinement: \textit{EXPTIME-complete}
- Composition & Conjunction: \textit{Absence of closure}
Design of a coffee machine:

Requirement one (temperature):
- hot: [0, 0.5]
- cold: [0, 1]

Requirement two (choice of beverage):
- coffee: [0, 1]
- tea: [0.2, 1]
Absence of closure: An illustration (1)

Design of a coffee machine:

- Requirement one (temperature): At most 50% of the drinks delivered are hot drinks
Absence of closure: An illustration (1)

Design of a coffee machine:

- Requirement one (temperature):

  \[ S_1 \]
  
  \[
  [0, 0.5] \rightarrow 2 \text{ hot}
  \]
  
  \[
  [0, 1] \rightarrow 3 \text{ cold}
  \]
Absence of closure: An illustration (1)

Design of a coffee machine:

- Requirement one (temperature):
  - $S_1$ [0, 0.5] → 1 → 2 (hot)
  - [0, 1] → 1 → 3 (cold)

- Requirement two (choice of beverage): At least 20% of the drinks delivered are coffee
Absence of closure: An illustration (1)

Design of a coffee machine:

- **Requirement one (temperature):**
  
  ![Diagram](image)
  
  - $S_1$
  - $[0, 0.5] ightarrow 2$ (hot)
  - $[0, 1] ightarrow 3$ (cold)

- **Requirement two (choice of beverage):**
  
  ![Diagram](image)
  
  - $S_2$
  - $[0, 1] ightarrow B$ (tea)
  - $[0.2, 1] ightarrow C$ (coffee)
Absence of closure: An illustration (2)

Conjunction of $S_1$ and $S_2$:

- Hot tea: $2, B$ (with $z_3$)
- Cold tea: $3, B$ (with $z_4$)
- Hot coffee: $2, C$ (with $z_1$)
- Cold coffee: $3, C$ (with $z_2$)
Absence of closure: An illustration (2)

Conjunction of $S_1$ and $S_2$:

$$z_1 \in [0, 0.5] \cap [0.2, 1] \Rightarrow z_1 \in [0.2, 0.5]$$
Absence of closure: An illustration (2)

Conjunction of $S_1$ and $S_2$:

- Hot tea: $(2, B)$, $z_3$, $[0.2, 0.5]$, $2, C$ (hot coffee)
- Cold tea: $(3, B)$, $z_4$, $z_2$, $3, C$ (cold coffee)
Absence of closure: An illustration (2)

Conjunction of $S_1$ and $S_2$:

$$z_2 \in [0, 1] \cap [0.2, 1] \Rightarrow z_1 \in [0.2, 1]$$
Absence of closure: An illustration (2)

Conjunction of $S_1$ and $S_2$:

- Hot tea: $2, B$ with $z_3 = [0.2, 0.5]
- Cold tea: $3, B$ with $z_4 = [0.2, 1]
- Hot coffee: $2, C$
- Cold coffee: $3, C$
Absence of closure: An illustration (2)

Conjunction of $S_1$ and $S_2$:

$$z_3 \in [0, 0.5] \cap [0, 1] \Rightarrow z_1 \in [0, 0.5]$$
Absence of closure: An illustration (2)

Conjunction of \( S_1 \) and \( S_2 \):

- Hot tea: \( 2, B \) with \([0, 0.5] \) to \( 1, A \) and \( 2, C \) with \([0.2, 0.5] \) to hot coffee.
- Cold tea: \( 3, B \) with \( z_4 \) to \( 1, A \) and \( 3, C \) with \([0.2, 1] \) to cold coffee.
Absence of closure: An illustration (2)

Conjunction of $S_1$ and $S_2$:

- Hot tea: $(2, B) \in [0, 0.5] \cap [0.2, 0.5] \Rightarrow 2, C \in \text{hot coffee}$
- Cold tea: $(3, B) \in [0.2, 1] \Rightarrow 3, C \in \text{cold coffee}$

$$z_4 \in [0, 1] \cap [0, 1] \Rightarrow z_1 \in [0, 1]$$
Absence of closure: An illustration (2)

Conjunction of $S_1$ and $S_2$:

- Hot tea: $(2, B)$ with $[0, 0.5]$ and $[0.2, 0.5]$ leading to $2, C$ for hot coffee.
- Cold tea: $(3, B)$ with $[0, 1]$ and $[0.2, 1]$ leading to $3, C$ for cold coffee.
Absence of closure: An illustration (2)

Conjunction of $S_1$ and $S_2$:

- **Hot tea**: $2, B$ with $[0, 0.5]$ and $[0.2, 0.5]$ leading to $2, C$ for hot coffee.
- **Cold tea**: $3, B$ with $[0, 1]$ and $[0.2, 1]$ leading to $3, C$ for cold coffee.

$(z_1, z_2, z_3, z_4) = (0.3, 0.2, 0.3, 0.2)$ is an implementation, but prob. for reaching “hot”, $z_1 + z_3 = 0.6$ violates $S_1$.
Absence of closure: An illustration (3)

IMCs are not closed under compositional operations

- Conjunction
- Parallel Composition

Solution: Constraint Markov Chains
Outline

1. Introduction

2. Specification Theories for Stochastic Systems
   - Interval Markov Chains
   - Constraint Markov Chains
   - Probabilistic Assume/Guarantee Contracts

3. Stochastic Abstraction and Analysis of Large Heterogeneous Systems

4. Conclusion and Future Work
Markov Chains (recap.)

$$(\{1, \ldots, n\}, o, M, A, V)$$

- states $\{1, \ldots, n\}$, $o$ initial state,
- $A$ is a set of atomic propositions, $V: \{1, \ldots, n\} \rightarrow 2^A$,
- $M \in [0, 1]^{n \times n}$ is a probability transition matrix: $\sum_{j=1}^{n} M_{ij} = 1$ for $i = 1, \ldots, n$. 

\[ V(1) \subseteq A \quad V(3) \subseteq A \]
\[ V(2) \subseteq A \quad V(4) \subseteq A \]
Constraint Markov Chains

$$(\{1, \ldots, k\}, o, \varphi, A, V)$$

- states $\{1, \ldots, k\}$, $o$ initial state
- $A$ is a set of atomic propositions, $V:\{1, \ldots, k\} \rightarrow 2^A$
- $\varphi:\{1, \ldots, k\} \rightarrow [0, 1]^k \rightarrow \{0, 1\}$

$V(1) \subseteq 2^A$
$V(3) \subseteq 2^A$
$V(2) \subseteq 2^A$
$V(4) \subseteq 2^A$
Constraint Markov Chain: An Example

\[(\{1, \ldots, k\}, 0, \varphi, A, V)\]

\[\varphi_1(1)(x_1, x_2, x_3) \equiv (x_1 = 0) \land (x_2 \geq 0.7) \land (x_2 + x_3 = 1)\]
Satisfaction Relation: An example

\[ \varphi_1(1)(x_1, x_2, x_3) \equiv (x_1 = 0) \land (x_2 \geq 0.7) \land (x_2 + x_3 = 1) \]
Satisfaction Relation: An example

\[ \varphi_1(1)(x_1, x_2, x_3) \equiv (x_1 = 0) \land (x_2 \geq 0.7) \land (x_2 + x_3 = 1) \]
Satisfaction Relation: An example

\[ \phi_1(1)(x_1, x_2, x_3) \equiv (x_1 = 0) \land (x_2 \geq 0.7) \land (x_2 + x_3 = 1) \]
Satisfaction Relation: An example

\[ \varphi_1(1)(x_1, x_2, x_3) \equiv (x_1 = 0) \land (x_2 \geq 0.7) \land (x_2 + x_3 = 1) \]

\[ \Delta = \begin{bmatrix}
0 & 0 & 0 \\
0 & 1 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1 \\
\end{bmatrix} \]
Satisfaction Relation: An example

\[ \varphi_1(1)(x_1, x_2, x_3) \equiv (x_1 = 0) \land (x_2 \geq 0.7) \land (x_2 + x_3 = 1) \]

\[ (0, 0.7, 0.1, 0.2) \times \Delta = (0, 0.8, 0.2) \]
Satisfaction Relation: An example

\[ \varphi_1(1)(x_1, x_2, x_3) \equiv (x_1 = 0) \wedge (x_2 \geq 0.7) \wedge (x_2 + x_3 = 1) \]

\[ (0, 0.7, 0.1, 0.2) \times \Delta = (0, 0.8, 0.2) \]

\[ \Rightarrow \varphi_1(1)((0, 0.7, 0.1, 0.2) \times \Delta) \text{ holds} \]
Weak Refinement

\[
\Delta = \begin{bmatrix}
0 & 0 & 0 \\
0 & 1 & 0 \\
0 & \gamma & 1-\gamma \\
0 & 0 & 1 \\
0 & 0 & 1
\end{bmatrix}
\]

\[
\gamma = \frac{0.7-z_{2,2}}{z_{2,3}} \text{ if } z_{2,2} \leq 0.7 \text{ and } \gamma = \frac{0.8-z_{2,2}}{z_{2,3}} \text{ otherwise}
\]
Conjunction: An Example

\[\varphi_1(1)(x) \equiv (x_1 = 0) \land (x_2 \leq 0.5) \land (x_2 + x_3 = 1)\]

\[\varphi_2(A)(y) \equiv (y_A = 0) \land (y_C \geq 0.2) \land (y_B + y_C = 1)\]
Conjunction: An Example

\[ \varphi_1(x) \equiv (x_1 = 0) \land (x_2 \leq 0.5) \land (x_2 + x_3 = 1) \]

\[ \varphi_2(A)(y) \equiv (y_A = 0) \land (y_C \geq 0.2) \land (y_B + y_C = 1) \]

\((z_1, A, z_1, B, \ldots, z_3, C)\) is valid iff
Conjunction: An Example

\[ \varphi_1(x) \equiv (x_1 = 0) \land (x_2 \leq 0.5) \land (x_2 + x_3 = 1) \]
\[ \varphi_2(y) \equiv (y_A = 0) \land (y_C \geq 0.2) \land (y_B + y_C = 1) \]

\((z_{1,A}, z_{1,B}, \ldots, z_{3,C})\) is valid iff
\[ z_{1,A} + z_{1,B} + z_{1,C} = 0 \]
Conjunction: An Example

\[ \varphi_1(1)(x) \equiv (x_1 = 0) \land (x_2 \leq 0.5) \land (x_2 + x_3 = 1) \]
\[ \varphi_2(A)(y) \equiv (y_A = 0) \land (y_C \geq 0.2) \land (y_B + y_C = 1) \]

\[(z_1,A, z_1,B, \ldots, z_3,C) \text{ is valid iff} \]
\[ z_1,A + z_1,B + z_1,C = 0 \]
\[ z_1,A + z_2,A + z_3,A = 0 \]
Conjunction: An Example

\[ \varphi_1(x) \equiv (x_1 = 0) \land (x_2 \leq 0.5) \land (x_2 + x_3 = 1) \]

\[ \varphi_2(y) \equiv (y_A = 0) \land (y_C \geq 0.2) \land (y_B + y_C = 1) \]

\( (z_1, z_1, B, \ldots, z_3, C) \) is valid iff

\[ z_1 + z_1, B + z_1, C = 0 \]
\[ z_1, A + z_2, A + z_3, A = 0 \]
\[ z_2, A + z_2, B + z_2, C \leq 0.5 \]
Conjunction: An Example

\[ \varphi_1(1)(x) \equiv (x_1 = 0) \land (x_2 \leq 0.5) \land (x_2 + x_3 = 1) \]

\[ \varphi_2(A)(y) \equiv (y_A = 0) \land (y_C \geq 0.2) \land (y_B + y_C = 1) \]

\((z_1,A, z_1,B, \ldots, z_3,C)\) is valid iff

\[
\begin{align*}
z_1,A + z_1,B + z_1,C & = 0 \\
z_1,A + z_2,A + z_3,A & = 0 \\
z_2,A + z_2,B + z_2,C & \leq 0.5 \\
z_1,C + z_2,C + z_3,C & \geq 0.2
\end{align*}
\]
Conjunction: An Example

\[ \varphi_1(1)(x) \equiv (x_1 = 0) \land (x_2 \leq 0.5) \land (x_2 + x_3 = 1) \]
\[ \varphi_2(A)(y) \equiv (y_A = 0) \land (y_C \geq 0.2) \land (y_B + y_C = 1) \]

\((z_1,A, z_1,B, \ldots, z_3,C)\) is valid iff

\[ z_1,A + z_1,B + z_1,C = 0 \]
\[ z_1,A + z_2,A + z_3,A = 0 \]
\[ z_2,A + z_2,B + z_2,C \leq 0.5 \]
\[ z_1,C + z_2,C + z_3,C \geq 0.2 \]
\[ \sum_{i,j} z_{i,j} = 1 \]
Conjunction

Conjunction is equal to intersection of sets of implementations

$$\varphi((u, v))(x_{1, 1}, x_{1, 2}, \ldots, x_{2, 1}, \ldots, x_{k_1, k_2}) \equiv \varphi_1(u)(\sum_{j=1}^{k_2} x_{1,j}, \ldots, \sum_{j=1}^{k_2} x_{k_1,j}) \land \varphi_2(v)(\sum_{i=1}^{k_1} x_{i,1}, \ldots, \sum_{i=1}^{k_1} x_{i,k_2})$$

Obtain linear constraints, even when composing linear (including interval) constraints
Results on CMCs [QEST’10]

- Satisfaction: OK
- Consistency: OK
- Refinement: OK
  - Syntactic notions
  - Complete for deterministic systems
- Conjunction: OK
- Composition: OK
  - Independent parallel composition
  - Synchronization using conjunction
- Polynomial CMCs: Closed under all operations
Outline

1. Introduction

2. Specification Theories for Stochastic Systems
   - Interval Markov Chains
   - Constraint Markov Chains
   - Probabilistic Assume/Guarantee Contracts

3. Stochastic Abstraction and Analysis of Large Heterogeneous Systems

4. Conclusion and Future Work
Beyond automata-based specification

Previously: Satisfaction relation = verification that a component satisfies a property expressed by an automaton

Question: What can we do with a logic?

Solution: Probabilistic Contracts
We develop a Contract-based compositional theory:

- Abstract representation of system behaviors
- Explicit separation between hypotheses on a component (Guarantees) and hypotheses on the environment (Assumptions)

⇒ Assume / Guarantee formalism
Qualitative / Quantitative notions for
- Satisfaction – Reliability VS Availability
- Refinement

Compositionality results for
- Parallel Composition
- Conjunction

Effective and Symbolic representations
Outline

1 Introduction

2 Specification Theories for Stochastic Systems
   - Interval Markov Chains
   - Constraint Markov Chains
   - Probabilistic Assume/Guarantee Contracts

3 Stochastic Abstraction and Analysis of Large Heterogeneous Systems

4 Conclusion and Future Work
Case Study: Accuracy of clock Synchronization

Challenges:

- Heterogeneous System over an Ethernet backbone
  - Distributed application
  - 280 communicating components

- Local clocks synchronized using the Precision Time Protocol

- Requirement: Verify that the difference between any 2 clocks is lower than a given bound
Case Study: Accuracy of clock Synchronization

Challenges:

- Heterogeneous System over an Ethernet backbone
  - Distributed application
  - 280 communicating components

- Local clocks synchronized using the Precision Time Protocol

- Requirement: Verify that the difference between any 2 clocks is lower than a given bound

- Our goal: Compute the best bound to satisfy this requirement without analyzing the whole architecture
find message delay characteristics by simulation of the overall model
find message delay characteristics by simulation of the overall model

apply statistical model-checking on the reduced model
First step

1. Learn the Probability distributions
First step

1. Learn the Probability distributions

![Graph showing device to server delay for PTP DELAY-REQUEST frames]
2. Use the distributions to study PTP
2. Use the distributions to study PTP
What are the questions?

- Qualitative question: Does $S$ satisfy $\Phi$ with a probability greater than $\theta$?
- Quantitative question: What is the probability for $S$ to satisfy $\Phi$?

Principle:

- Reason on a finite set of executions and answer the question
- We may make mistakes, but we can control precision
Model/Abstraction:

- PTP and HCS modeled using BIP
- Distributions of delays: 2000 measures

Statistical Model Checking:

- Quantitative question: precision $10^{-2}$, confidence $10^{-2}$: 100000 simulations
- Qualitative question: precision $10^{-3}$, confidence $10^{-10}$: 300000 simulations
Some Results 1/2

The property is not satisfied for the given bound!

Probability of bounded accuracy

Probability of satisfying Bounded Accuracy for a bound of 50µs
Some Results 1/2

Probability of satisfying Bounded Accuracy for a bound of 50µs

- The property is not satisfied for the given bound!
Probability of satisfying Bounded Accuracy as a function of the bound

The best bound for which B.A. is satisfied with probability 1 is $10^5 \mu s$.
The best bound for which B.A. is satisfied with probability 1 is 105µs.
Results for stochastic abstraction

- Abstraction and verification method

- Applied to 2 case studies:
  - HCS case study [FORTE’10]
  - AFDX network [RV’10]
Outline

1. Introduction

2. Specification Theories for Stochastic Systems
   - Interval Markov Chains
   - Constraint Markov Chains
   - Probabilistic Assume/Guarantee Contracts

3. Stochastic Abstraction and Analysis of Large Heterogeneous Systems

4. Conclusion and Future Work
Contributions

- Specification of stochastic systems
Contributions

- Specification of stochastic systems
  - Algorithms and complexity results for IMCs

Interesting result: Refinement complete for deterministic CMCs

Quantitative analysis: probabilistic contracts

New simulation-based technique for the verification of huge systems

Verification of Bounded Accuracy on HCS

Verification of Latency on an AFDX network
Contributions

- Specification of stochastic systems
  - Algorithms and complexity results for IMCs
  - First compositional specification theory for Markov Chains: CMCs [QEST’10]
    Interesting result: Refinement complete for deterministic CMCs

- Quantitative analysis: probabilistic contracts [ACSD’10]
- New simulation-based technique for the verification of huge systems [FORTE’10]
- Verification of Bounded Accuracy on HCS [RV’10]
- Verification of Latency on an AFDX network [RV’10]
Contributions

- Specification of stochastic systems
  - Algorithms and complexity results for IMCs
  - First compositional specification theory for Markov Chains: CMCs [QEST’10]
    - Interesting result: Refinement complete for deterministic CMCs
  - Quantitative analysis: probabilistic contracts [ACSD’10]

- New simulation-based technique for the verification of huge systems
  - Verification of Bounded Accuracy on HCS [FORTE’10]
  - Verification of Latency on an AFDX network [RV’10]
Contributions

- Specification of stochastic systems
  - Algorithms and complexity results for IMCs
  - First compositional specification theory for Markov Chains: CMCs [QEST’10]
    - Interesting result: Refinement complete for deterministic CMCs
  - Quantitative analysis: probabilistic contracts [ACSD’10]

- New simulation-based technique for the verification of huge systems
Contributions

- Specification of stochastic systems
  - Algorithms and complexity results for IMCs
  - First compositional specification theory for Markov Chains: CMCs [QEST’10]
    - Interesting result: Refinement complete for deterministic CMCs
  - Quantitative analysis: probabilistic contracts [ACSD’10]

- New simulation-based technique for the verification of huge systems
  - Verification of Bounded Accuracy on HCS [FORTE’10]
Contributions

- Specification of stochastic systems
  - Algorithms and complexity results for IMCs
  - First compositional specification theory for Markov Chains: CMCs [QEST’10]
    - Interesting result: Refinement complete for deterministic CMCs
  - Quantitative analysis: probabilistic contracts [ACSD’10]

- New simulation-based technique for the verification of huge systems
  - Verification of Bounded Accuracy on HCS [FORTE’10]
  - Verification of Latency on an AFDX network [RV’10]
Future Work

- Continuous-time & Non-determinism
Future Work

- Continuous-time & Non-determinism
  - Abstract Probabilistic Automata [VMCAI’11]
Future Work

- Continuous-time & Non-determinism
  - Abstract Probabilistic Automata [VMCAI’11]
  - Continuous-time CMCs
Future Work

- Continuous-time & Non-determinism
  - Abstract Probabilistic Automata [VMCAI’11]
  - Continuous-time CMCs

- Implementation
Future Work

- Continuous-time & Non-determinism
  - Abstract Probabilistic Automata [VMCAI’11]
  - Continuous-time CMCs

- Implementation
  - Tool for CMCs
Future Work

- Continuous-time & Non-determinism
  - Abstract Probabilistic Automata [VMCAI’11]
  - Continuous-time CMCs

- Implementation
  - Tool for CMCs
  - Tool for probabilistic contracts
Future Work

- Continuous-time & Non-determinism
  - Abstract Probabilistic Automata [VMCAI'11]
  - Continuous-time CMCs

- Implementation
  - Tool for CMCs
  - Tool for probabilistic contracts

- Rare events
Future Work

- Continuous-time & Non-determinism
  - Abstract Probabilistic Automata [VMCAI’11]
  - Continuous-time CMCs

- Implementation
  - Tool for CMCs
  - Tool for probabilistic contracts

- Rare events
  - Use this theory to “guide” simulations for better precision
Publications

- [FORTE’10] Statistical Abstraction and Model-Checking of Large Heterogeneous Systems. *FMOODS/FORTE 2010*
- [QEST’10] Compositional Design Methodology with Constraint Markov Chains. *QEST 2010*
- [RV’10] Verification of an AFDX Architecture Using Simulations and Probabilities. *RV 2010*
- [RV’10 (2)] Statistical Model Checking: Present and Future. *RV 2010*
Thank you for your attention