Probabilistic Contracts for Component-based Design

Dana N. Xu  Gregor Gössler  Alain Girault

INRIA, France

ATVA 2010
Probabilistic Contracts

System designers have to cope with multiple sources of uncertainty:

- Embedded and distributed systems usually encompass **unreliable** components.
- Behaviors of (black-box) components and the environment may be uncertain.
- **Abstraction** from complex deterministic behavior (“network access is available with p=95%”).

We want to describe properties such as:
“The probability that this component fails at this point of its behavior is \( \leq 0.1\% \).”

We introduce **probabilistic contracts**, which distinguish **assumptions** on how a component is used from **guarantees** on the component behavior.
Interactive Markov Chain (IMC)

Example: client – link – server.

An IMC is an LTS with action states/transitions and probabilistic states/transitions [Hermanns 2002].

IMC used to model component behaviors:

The IMC $M_\ell$ of the Link.
A probabilistic contract is an IMC with probability intervals and a special $\top$ state:

- Action transitions leading to $\top$ are **assumed** not to be synchronized.
- Action transitions not leading to $\top$ are **guaranteed** to be offered.
- Actions not labelling any transition at a state are guaranteed not to be offered.
Essential operations:

- **refinement** and **satisfaction**;

- **parallel composition** \((C_1 || \mathcal{I} C_2)\): E.g. \(\mathcal{I} = \{a|d, b|e, c|f, g, u, v\}\)

- **conjunction** of contracts \((C_1 \land C_2)\):

Additional definitions: **bisimulation, reduction, projection**
Contract Refinement

\[ C_1 \]

\[ C_2 \]

\[ C_3 \]

\[ C_1 \leq C_3 \]

\[ C_2 \leq C_3 \]
Contract refinement for probabilistic states

[Jonsson and Larsen : LICS’91]
Contract Satisfaction

IMC $M_S$

Lifted IMC $\lfloor M_S \rfloor$
Contract Satisfaction

**Definition (Contract satisfaction)**

An IMC $M$ satisfies a contract $C$ (written $M \models C$) iff $\lfloor M \rfloor \leq C$.

That is to check: $s_0 \leq t_0$
Contract Satisfaction

**Definition (Models of contracts)**
The set of models of a contract $C$ (written $\mathcal{M}(C)$) is the set of IMCs that satisfy $C$: $\mathcal{M}(C) = \{ M \mid M \models C \}$.

**Definition (Semantical equivalence)**
Contracts $C_1$ and $C_2$ are semantically equivalent (written $C_1 \equiv C_2$) iff $\mathcal{M}(C_1) = \mathcal{M}(C_2)$.

**Lemma (Refinement and model inclusion)**
For all contracts $C_1$ and $C_2$, if $C_1 \leq C_2$, then $\mathcal{M}(C_1) \subseteq \mathcal{M}(C_2)$. 
Parallel Composition of contracts over two components

- A probabilistic transition has higher priority than an action transition.
- Interaction set $\mathcal{I}$: only transitions labeled with interactions in $\mathcal{I}$ can occur.
- Synchronize two probabilistic transitions.
- If one contract reaches $\top$, the composed contract reaches $\top$.

$$C_1 ||_\mathcal{I} C_2 \text{ where } \mathcal{I} = \{a | c, b, d\}$$
Parallel Composition of contracts over two components

- A probabilistic transition has higher priority than an action transition.
- Interaction set $\mathcal{I}$: only transitions labeled with interactions in $\mathcal{I}$ can occur.
- Synchronize two probabilistic transitions.
- If one contract reaches $\top$, the composed contract reaches $\top$.

$C_1 \parallel_{\mathcal{I}} C_2$ where $\mathcal{I} = \{a|c, b, d\}$
Parallel Composition of contracts over two components

- A probabilistic transition has higher priority than an action transition.
- Interaction set $\mathcal{I}$: only transitions labeled with interactions in $\mathcal{I}$ can occur.
- Synchronize two probabilistic transitions.
- If one contract reaches $\top$, the composed contract reaches $\top$.

\[
\begin{array}{ccc}
(s_0, t_0) & \rightarrow & (s_1, t_0) \\
(s_0, t_0) & \rightarrow & (s_2, t_0) \\
\end{array}
\]

\[
\begin{array}{ccc}
s_0 & \xrightarrow{a} & s_3 \\
s_1 & \xrightarrow{b} & s_2 \\
s_3 & \xrightarrow{[0.2, 0.5]} & s_5 \\
s_5 & \xrightarrow{a} & s_6 \\
s_6 & \xrightarrow{b} & t_0 \\
t_0 & \xrightarrow{c} & t_1 \\
t_1 & \xrightarrow{[0.1, 0.3]} & t_3 \\
t_3 & \xrightarrow{c} & \top \\
\end{array}
\]

\[
C_1 \parallel_{\mathcal{I}} C_2 \quad \text{where} \quad \mathcal{I}=\{a, c, b, d\}
\]
Parallel Composition of contracts over two components

- A probabilistic transition has higher priority than an action transition.
- Interaction set $\mathcal{I}$: only transitions labeled with interactions in $\mathcal{I}$ can occur.
- Synchronize two probabilistic transitions.
- If one contract reaches $\top$, the composed contract reaches $\top$.

$$\begin{align*}
C_1 & \quad (s_0, t_0) \quad a|c \quad (s_1, t_0) \\
C_2 & \quad t_0 \quad c \rightarrow t_1 \quad (s_1, t_0) \quad a|c \quad (s_3, t_1) \\
\end{align*}$$

$$C_1 \parallel_{\mathcal{I}} C_2 \text{ where } \mathcal{I} = \{a|c, b, d\}$$
Parallel Composition of contracts over two components

- A probabilistic transition has higher priority than an action transition.
- Interaction set $\mathcal{I}$: only transitions labeled with interactions in $\mathcal{I}$ can occur.
- Synchronize two probabilistic transitions.
- If one contract reaches $\top$, the composed contract reaches $\top$.

$C_1 \parallel_{\mathcal{I}} C_2$ where $\mathcal{I} = \{ a \mid c, b, d \}$
Parallel Composition of contracts over two components

- A probabilistic transition has higher priority than an action transition.
- Interaction set $\mathcal{I}$: only transitions labeled with interactions in $\mathcal{I}$ can occur.
- Synchronize two probabilistic transitions.
- If one contract reaches $\top$, the composed contract reaches $\top$.

$C_1 || \mathcal{I} C_2$ where $\mathcal{I}=\{a|c, b, d\}$
Parallel Composition of contracts over two components

- A probabilistic transition has higher priority than an action transition.
- Interaction set $\mathcal{I}$: only transitions labeled with interactions in $\mathcal{I}$ can occur.
- Synchronize two probabilistic transitions.
- If one contract reaches $\top$, the composed contract reaches $\top$.

$C_1 \parallel_{\mathcal{I}} C_2$ where $\mathcal{I} = \{a|c, b, d\}$
Parallel Composition of contracts over two components

- A probabilistic transition has higher priority than an action transition.
- Interaction set $\mathcal{I}$: only transitions labeled with interactions in $\mathcal{I}$ can occur.
- Synchronize two probabilistic transitions.
- If one contract reaches $\top$, the composed contract reaches $\top$.

$$C_1 \parallel_\mathcal{I} C_2$$ where $\mathcal{I} = \{a|c, b, d\}$
Parallel Composition of contracts over two components

- A probabilistic transition has higher priority than an action transition.
- Interaction set $\mathcal{I}$: only transitions labeled with interactions in $\mathcal{I}$ can occur.
- Synchronize two probabilistic transitions.
- If one contract reaches $\top$, the composed contract reaches $\top$.

$C_1 \mid\mid_{\mathcal{I}} C_2$ where $\mathcal{I} = \{a|c, b, d\}$
Parallel Composition of contracts over two components

- A probabilistic transition has higher priority than an action transition.
- Interaction set $\mathcal{I}$: only transitions labeled with interactions in $\mathcal{I}$ can occur.
- Synchronize two probabilistic transitions.
- If one contract reaches $\top$, the composed contract reaches $\top$.

$C_1 \parallel_{\mathcal{I}} C_2$ where $\mathcal{I}=\{a|c, b, d\}$
Parallel Composition of contracts over two components

- A probabilistic transition has higher priority than an action transition.
- Interaction set $\mathcal{I}$: only transitions labeled with interactions in $\mathcal{I}$ can occur.
- Synchronize two probabilistic transitions.
- If one contract reaches $\top$, the composed contract reaches $\top$.

$C_1 || \mathcal{I} C_2$ where $\mathcal{I} = \{a|c, b, d\}$
Parallel Composition of contracts over two components

- A probabilistic transition has higher priority than an action transition.
- Interaction set $\mathcal{I}$: only transitions labeled with interactions in $\mathcal{I}$ can occur.
- Synchronize two probabilistic transitions.
- If one contract reaches $\top$, the composed contract reaches $\top$.

$C_1 | |_\mathcal{I} C_2$ where $\mathcal{I} = \{a|c, b, d\}$
Properties for Parallel Composition

Theorem (Congruence of refinement for $\|\mathcal{I}\$)
For all contracts $C_1, C_2, C_3, C_4$ and interaction set $\mathcal{I}$, if $C_1 \leq C_2$ and $C_3 \leq C_4$, then $C_1 \|\mathcal{I} C_3 \leq C_2 \|\mathcal{I} C_4$.

Theorem (Independent implementability)
For all IMCs $M, N$, contracts $C_1, C_2$, and interaction set $\mathcal{I}$, if $M \models C_1$ and $N \models C_2$, then $M \|\mathcal{I} N \models C_1 \|\mathcal{I} C_2$. 
**Conjunction:** composition of requirements over a same component

- A probability transition has a higher priority than an action transition.
- Contracts must agree on common action transitions.
- Intersect probability intervals for two states that are similar.
- If one contract reaches $\top$, the conjunction behaves like the other contract.

$$C_1 \text{ with } A_1 = \{a, b, c\} \quad C_2 \text{ with } A_2 = \{a, b, d\}$$

$$C_1 \land C_2$$
Conjunction: composition of requirements over a same component

- A probability transition has a higher priority than an action transition.
- Contracts must agree on common action transitions.
- Intersect probability intervals for two states that are similar.
- If one contract reaches \( \top \), the conjunction behaves like the other contract.

\[
\begin{align*}
C_1 \text{ with } A_1 &= \{a, b, c\} \\
C_2 \text{ with } A_2 &= \{a, b, d\}
\end{align*}
\]
**Conjunction:** composition of requirements over a same component

- A probability transition has a higher priority than an action transition.
- Contracts must agree on common action transitions.
- Intersect probability intervals for two states that are similar.
- If one contract reaches $\top$, the conjunction behaves like the other contract.

$C_1$ with $A_1 = \{a, b, c\}$

$C_2$ with $A_2 = \{a, b, d\}$
**Conjunction:** composition of requirements over a same component

- A probability transition has a higher priority than an action transition.
- Contracts must agree on common action transitions.
- Intersect probability intervals for two states that are similar.
- If one contract reaches $\top$, the conjunction behaves like the other contract.

$$C_1 \text{ with } A_1 = \{a, b, c\} \quad C_2 \text{ with } A_2 = \{a, b, d\}$$
Conjunction: composition of requirements over a same component

- A probability transition has a higher priority than an action transition.
- Contracts must agree on common action transitions.
- Intersect probability intervals for two states that are similar.
- If one contract reaches $\top$, the conjunction behaves like the other contract.

\[ C_1 \text{ with } A_1 = \{a, b, c\} \quad C_2 \text{ with } A_2 = \{a, b, d\} \]
Conjunction: composition of requirements over a same component

- A probability transition has a higher priority than an action transition.
- Contracts must agree on common action transitions.
- Intersect probability intervals for two states that are similar.
- If one contract reaches $\top$, the conjunction behaves like the other contract.

$C_1$ with $A_1 = \{a, b, c\}$  
$C_2$ with $A_2 = \{a, b, d\}$

$C_1 \land C_2$
Conjunction: composition of requirements over a same component

- A probability transition has a higher priority than an action transition.
- Contracts must agree on common action transitions.
- Intersect probability intervals for two states that are similar.
- If one contract reaches \( \top \), the conjunction behaves like the other contract.

\( C_1 \) with \( A_1 = \{a, b, c\} \)  
\( C_2 \) with \( A_2 = \{a, b, d\} \)

\[ C_1 \land C_2 \]
Conjunction: composition of requirements over a same component

- A probability transition has a higher priority than an action transition.
- Contracts must agree on common action transitions.
- Intersect probability intervals for two states that are similar.
- If one contract reaches $\top$, the conjunction behaves like the other contract.

$C_1$ with $A_1 = \{a, b, c\}$  
$C_2$ with $A_2 = \{a, b, d\}$

$C_1 \land C_2$
Conjunction: composition of requirements over a same component

- A probability transition has a higher priority than an action transition.
- Contracts must agree on common action transitions.
- Intersect probability intervals for two states that are similar.
- If one contract reaches ⊤, the conjunction behaves like the other contract.

\begin{align*}
C_1 \text{ with } A_1 &= \{a, b, c\} \\
C_2 \text{ with } A_2 &= \{a, b, d\}
\end{align*}
Conjunction: composition of requirements over a same component

- A probability transition has a higher priority than an action transition.
- Contracts must agree on common action transitions.
- Intersect probability intervals for two states that are similar.
- If one contract reaches $\top$, the conjunction behaves like the other contract.

$C_1$ with $A_1 = \{a, b, c\}$  $C_2$ with $A_2 = \{a, b, d\}$

$C_1 \land C_2$
Unambiguous Contracts

For conjunction, we require the contracts to be **unambiguous**.

Unambiguous Contract

Ambiguous Contract

Unambiguous Contract
Properties of Conjunction

Theorem (Soundness of conjunction)

For all unambiguous contracts $C_1$ and $C_2$ with alphabets $A$ such that:

$$C_1 \land C_2 \leq C_i \text{ for } i = 1, 2$$
Case Study

Requirment $C_s$ on the server

Contract $C_P$ of a processor

Contract $C_T$ of a re-execution scheduler

$I = \{success, comp, fail, exe|exe', ok|ok', nok|nok'\}$
Case Study

\begin{align*}
\mathcal{C}_s &= \mathbb{I} \cdot \mathbb{P} \\
\mathcal{C}_T \| P &= \mathcal{C}_T \| I \mathcal{C}_P
\end{align*}

Shortcuts: \( \texttt{exe} = \texttt{exe} | \texttt{exe}' \quad \texttt{ok} = \texttt{ok} | \texttt{ok}' \quad \texttt{nok} = \texttt{nok} | \texttt{nok}' \)
Case study: Refinement to Guarantee Reliability

- Collapse probabilistic transitions:

\[ C_\pi = \pi_B(C_T \| P) \]

\[ \mathcal{B} = \{ \text{success}, \text{comp}, \text{fail} \} \]

- Refinement \( \tilde{C}_\pi \leq C_S \) of reliability contract \( C_S \) gives constraint on \( p \): \((1 - p)^2 \leq 0.001\), that is, \( p \geq 0.969 \).
Conclusion

- Developed a probabilistic contract framework for component-based design.
- Provide operations for bottom-up and top-down design: refinement, parallel composition, and conjunction.
- Proved the desired properties of these operations.
- Small case study to show its usefulness.

Future work directions:
- Implement the framework in a tool, e.g. CADP model-checker
- Work on larger case studies.
- Study blaming (statically and at run-time).