Behavioural Specifications and Quantitative Techniques

Emilio Tuosto

GRAN SASSO SCIENCE INSTITUTE
SCHOOL OF ADVANCED STUDIES
Scuola Universitaria Superiore

Challenges and Perspectives of Formal Methods for Trustworthy Software
PhD seminars series

IMT SCUOLA ALTI STUDI LUCCA

May 16, 2024

Plan of the talk

An overview of results

- Resource-awareness
- Probabilities
- Time

in behavioural specifications (mainly, behavioural types)

Some open problems

– Resource Awareness –

Das, Hoffmann, Pfenning: [LICS'18]
Work Analysis with Resource-Aware Session types

Static derivation of worst-case bounds on work for communication

$$S, T \ ::= \ V \ \mid \ \oplus\{l_i^{q_i} : S\} \ \mid \ \&\{l_i^{q_i} : S\} \ \mid \ S \overset{q}{\multimap} T \ \mid \ S \overset{q}{\otimes} T \ \mid \ \mathbf{1}^q$$

Das, Hoffmann, Pfenning: [LICS'18]
Work Analysis with Resource-Aware Session types

Static derivation of worst-case bounds on work for communication

$$S, T \;::=\; V \;\mid\; \oplus\{l_i^{q_i} : S\} \;\mid\; \&\{l_i^{q_i} : S\} \;\mid\; S \overset{q}{\multimap} T \;\mid\; S \overset{q}{\otimes} T \;\mid\; \mathbf{1}^{q}$$

Das, Hoffmann, Pfenning: [LICS'18] Work Analysis with Resource-Aware Session types

Static derivation of worst-case bounds on work for communication

$$S, T \;::=\; V \;\mid\; \oplus\{l_i^{q_i} : S\} \;\mid\; \&\{l_i^{q_i} : S\} \;\mid\; S \overset{q}{\multimap} T \;\mid\; S \overset{q}{\otimes} T \;\mid\; \mathbf{1}^{q}$$

Theorem (Soundness)

WF processes don't "generate" energy

Theorem (Progress)

WF processes get stuck only if they correctly terminate
(direct conseguence of progress in SILL [Toninho, Caires, Pfenning: ESOP'13])

Case studies

Das, Balzar, Hoffmann, Pfenning, Santurkar: Resource-Aware STs for digital contracts [CSF'21]

Nomos: DSL to account for requirements of digital contracts

Session-typing discipline for resource-analysis

- linearity to prevent duplication/deletion of assets
- type reconstruction to automatically infer resource bounds
- amortized resource analysis to control resource usage

Theorem

Type preservation & progress

Evaluation on case studies

Verify the system-wide QoS properties of a distributed system given the QoS of its components

Lopez Pombo, Martinez-Suñé, Tuosto: [ICTAC'23] A Dynamic Temporal Logic for QoS in Choreographic Models

Verify the system-wide QoS properties of a distributed system given the QoS of its components

QL logic

$$\begin{array}{rcl} \Phi & ::= & \top \\ & | & \psi \\ & | & \neg\Phi \\ & | & \Phi \vee \Phi \\ & | & \Phi \; \mathsf{U}^{\mathsf{G}} \Phi \end{array}$$

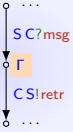
Lopez Pombo, Martinez-Suñé, Tuosto: [ICTAC'23] A Dynamic Temporal Logic for QoS in Choreographic Models

Verify the system-wide QoS properties of a distributed system given the QoS of its components

QL logic

$$\begin{array}{rcl} \Phi & ::= & \top \\ & | & \psi \\ & | & \neg\Phi \\ & | & \Phi \vee \Phi \\ & | & \Phi \; \mathsf{U}^{\mathsf{G}} \Phi \end{array}$$

QoS CFSMs

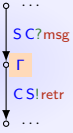


Verify the system-wide QoS properties of a distributed system given the QoS of its components

QL logic

$$\begin{array}{rcl} \Phi & ::= & \top \\ & | & \psi \\ & | & \neg\Phi \\ & | & \Phi \vee \Phi \\ & | & \Phi \; \mathsf{U}^{\mathsf{G}} \Phi \end{array}$$

QoS CFSMs



QoS Specs

$$\begin{array}{rcl} \Sigma & = & \langle \{0,1\} \cup \mathsf{Q}, \\ & & \{+,\cdot\} \cup \{\oplus^{\mathsf{a}}\}_{\mathsf{a}\in\mathsf{Q}}, \\ & & < \\ & & \rangle \\ \Gamma & = & \Gamma_{\mathrm{RCF}} \cup \Gamma' \end{array}$$

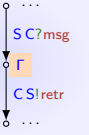
Lopez Pombo, Martinez-Suñé, Tuosto: [ICTAC'23] A Dynamic Temporal Logic for QoS in Choreographic Models

Verify the system-wide QoS properties of a distributed system given the QoS of its components

QL logic

$$\begin{array}{rcl} \Phi & ::= & \top \\ & \mid & \psi \\ & \mid & \neg\Phi \\ & \mid & \Phi \vee \Phi \\ & \mid & \Phi \; \mathsf{U}^{\mathsf{G}} \Phi \end{array}$$

QoS CFSMs



QoS Specs

$$\begin{array}{rcl} \Sigma & = & \langle \{0,1\} \cup \mathsf{Q}, \\ & & \{+,\cdot\} \cup \{\oplus^{\mathsf{a}}\}_{\mathsf{a}\in\mathsf{Q}}, \\ & & \prec \\ & & \rangle \\ \Gamma & = & \Gamma_{\mathrm{RCF}} \cup \Gamma' \end{array}$$

Bounded MC

```
1  qUntil(Φ1, G, Φ2, S, π, π′, π″):
2      if L[π″] ∈ L̂[G] and qModels(Φ2, S, π, π′π″) then
3          return true
4      else if not qModels(Φ1, S, π, π′π″) then
5          return false
6      else
7          Let ⇝ q be the transition such that π′π″ ⇝ q ∈ prf(π)
             (takes the first transition in π if π′π″ = ϵ,
             and it is not defined if π′π″ = π)
8          if π′π″ = π or L[π″ ⇝ q] ∉ L[G] then
9              return false
10         else
11             return qUntil(Φ1, G, Φ2, S, π, π′, π″ ⇝ q)
```

now implemented in **ChorGram**

– Probabilities –

$$
\begin{array}{lcll}
G & ::= & \sum_{i\in I} q \to_{\delta_i} q' : k\langle S_i\rangle.G_i & (probValues)\\
 & \mid & q \to_1 q' : k\langle T@p\rangle.G & (delegation)\\
 & \mid & \sum_{i\in I} q \to_{\delta_i} q' : k\langle l_i : G_i\rangle & (probBranching)\\
 & \mid & G, G & (parallel)\\
 & \mid & \mu\, t.G & (recursive)\\
 & \mid & t & (variable)\\
 & \mid & \mathtt{end} & (end)\\
S & ::= & \mathtt{bool} \mid \mathtt{nat}\cdots & (valuetypes)
\end{array}
$$

Theorem (Soundness)

Well-formed processes do not have "proabability errors"

In WOLLIC'22 things get simpler using results in
[Darda, Hu, Scalas, and Yoshida:ECOOP'22]

Inverso, Melgratti, Padovani, Trubiani, Tuosto: Probabilistic Analysis of Binary Sessions [Concur'20]

Reasoning about session termination ... probabilistically

$$S, T \overset{\text{co}}{::=} \circ \mid \bullet \mid ?t.S \mid !t.S \mid S \oplus^{p} T \mid S \,\&^{p}\, T$$

success

prob. choice

technical convenience

Reasoning about session termination ... probabilistically

$$S, T \overset{\text{co}}{::=} \circ \mid \bullet \mid ?t.S \mid !t.S \mid S \overset{p}{\oplus} T \mid S \overset{p}{\&} T$$

success — prob. choice — technical convenience

Session Types $\mapsto$ Discrete-Time Markov Chains

Inverso, Melgratti, Padovani, Trubiani, Tuosto: Probabilistic Analysis of Binary Sessions [Concur'20]

Reasoning about session termination ... probabilistically

$$S, T \overset{\text{co}}{::=} \circ \mid \bullet \mid ?t.S \mid !t.S \mid S \overset{p}{\oplus} T \mid S \overset{p}{\&} T$$

success — prob. choice — technical convenience

Session Types $\mapsto$ Discrete-Time Markov Chains

Theorem (Soundness)

- *(Possibly diverging) well-typed processes respects (the probabilities in) their type*
- *Well-typed processes are deadlock-free*
- *Type preservation*

Inverso, Melgratti, Padovani, Trubiani, Tuosto: Probabilistic Analysis of Binary Sessions [Concur'20]

Reasoning about session termination ... probabilistically

$$S, T \overset{\text{co}}{::=} \circ \mid \bullet \mid ?t.S \mid !t.S \mid S \oplus^{p} T \mid S \,\&^{p}\, T$$

success — prob. choice — technical convenience

Session Types $\mapsto$ Discrete-Time Markov Chains

Theorem (Soundness)

- *(Possibly diverging) well-typed processes respects (the probabilities in) their type*
- *Well-typed processes are deadlock-free*
- *Type preservation*

[Burló, Francalanza, Scalas, Trubiani, Tuosto:SCP'21] defines a probabilistic monitoring discipline

Dal Lago, Giusti [Concur'22]
On Session Typing, Probabilistic Time, and cryptographic experiments

Binary session types to model cryptographic experiments

- Extend [Caires, Pfenning: Concur'10] with
  - probabilistic choice
  - polytime functions
- this yields a model of cryptographic protocols

Theorem (Soundness)

Subject reduction & progress holds for well typed processes

Theorem (Confluence)

Well typed processes are confluent

Dal Lago, Giusti [Concur'22]
On Session Typing, Probabilistic Time, and cryptographic experiments

Binary session types to model cryptographic experiments

- Extend [Caires, Pfenning: Concur'10] with
  - probabilistic choice
  - polytime functions
- this yields a model of cryptographic protocols

Theorem (Soundness)

Subject reduction & progress holds for well typed processes

Theorem (Confluence)

Well typed processes are confluent

Silent steps converge to a same probability distribution

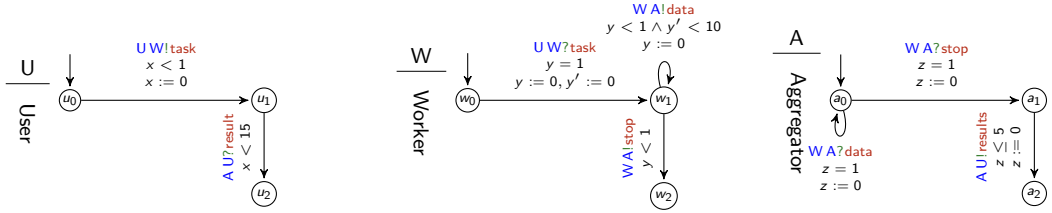
Das, Wang, Hoffman: [PoPL'23] Probabilistic Resource Aware Session Types

Resource analysis of probabilistic systems via probabilistic binary session types

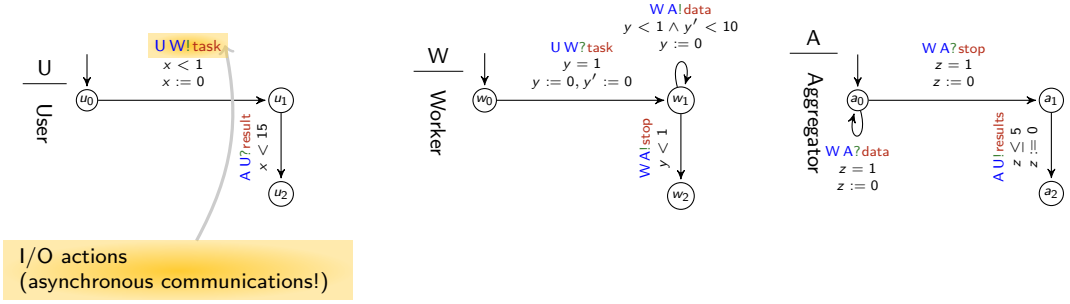
- Probabilistic binary session types to derive expected cost bounds of message-passing systems (exteding [LICS'18])
- The proposed calculus features both non-deterministic and probabilistic choices
- The typing discipline ensures type preservation, progress, and probability consistency
- NomosPro is implemented and extensively evaluated

– Time –

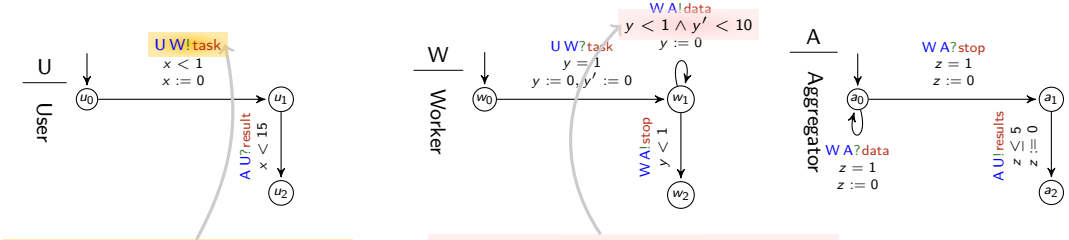
An example borrowed from [Bocchi, Lange, Yoshida:Concur'15]:



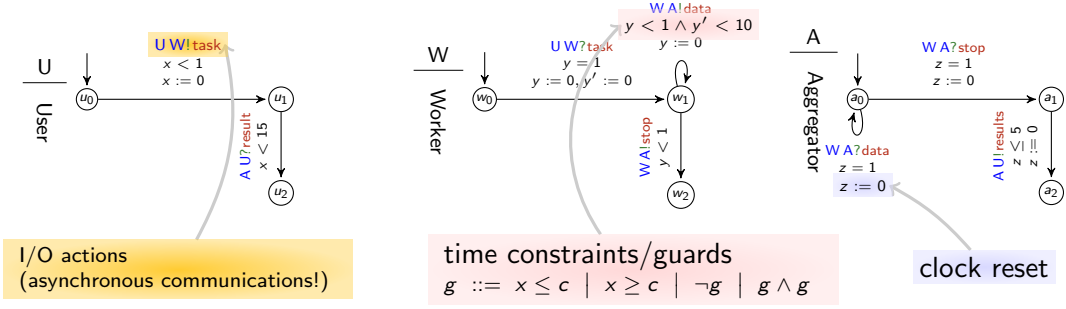
An example borrowed from [Bocchi, Lange, Yoshida:Concur'15]:



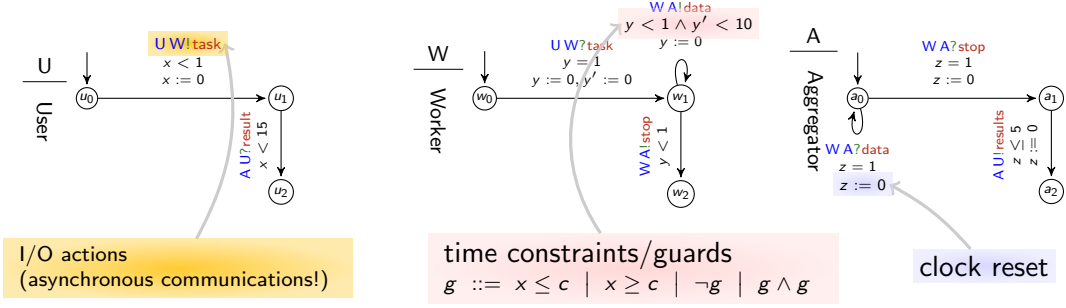
An example borrowed from [Bocchi, Lange, Yoshida:Concur'15]:



An example borrowed from [Bocchi, Lange, Yoshida:Concur'15]:



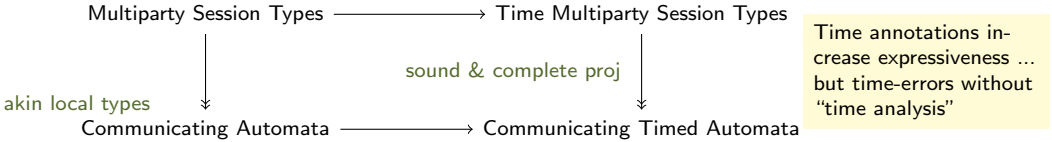
An example borrowed from [Bocchi, Lange, Yoshida:Concur'15]:



Evaluations

$\nu : x \mapsto r$ where $r \in \Re^{\geq 0}$

Bocchi, Yang, Yoshida [Concur'14]
Timed Multiparty Session Types



Theorem (Soundness)

Well-typed processes respect timing

Theorem (Progress)

$$\textit{Feasibility} + \textit{Wait-Freedom} \implies \textit{time progress of typed processes}$$

where

Feasibility	=	*Partial timed executions can be completed*
Wait-Freedom	=	*if senders respect their timing then receivers do not have to wait*

CFSM/timed automata $\longrightarrow$ Communicating Timed Automata

Sound membership decision procedure
- safety = deadlock freedom + no orphan messages
- eventual reception
- progress
- no-zenoness

Theorem (Soundness)

If S is a multiparty-compatible system then

- *S is safe and*
- *S is timed bisimilar to its projection*

Toolchain for timed interactions

- define timed protocols in **Scribble**
- check for feasibility and wait-freedom
- projections
- derive monitors to check timing constraints

The binary (synchronous) case is interesting too!

Duality does not yield compatibility (i.e., deadlock-freedom)

- is compatibility decidable in timed session types?
- can compliant timed counterpart be found?

The binary (synchronous) case is interesting too!
Duality does not yield compatibility (i.e., deadlock-freedom)

- is compatibility decidable in timed session types?
- can compliant timed counterpart be found?

Theorem (Reduction)

Timed compliance can be (algorithmically) reduced to model-check deadlock-freedom in timed automata ($\Longrightarrow$ decidability of timed compliance)

Theorem (Characterisation of evaluations)

The set of evaluations of a timed session type admitting a compliant timed session type is effectively computable
($\Longrightarrow$ canonical compliant timed session types are computable)

Theorem (Decidability of Subtyping)

Subtyping is decidable in timed session types

Synchronous compliance $\not\Longrightarrow$ Asynchronous compliance
The implication holds if firable inputs are not delayed
This result generated interesting new stuff:

- Refinements of Communicating Timed Automata (aking to timed session types) that do not introduce deadlocks/livelocks are simulated by abstract systems [Bartoletti, Bocchi, Murgia:Concur'18]
- Generalisation of duality & compliance based on urgent semantics [Bocchi, Murgia, Vasconcelos, Yoshida:ESOP'19]

Binary asynchronous session types with LTL-like modalities

- $\circ A$ inhabited by process of the form delay;P (P behaves as A after a time unit!)
- $\Box A$ = "always ready to do A"
- $\Diamond A$ = "eventually ready to do A"

This give a generic framework for cost analysis

Well-typed processes enjoy progress & type preservation

The cost model can be parameterised

Extend mixed sessions [Vasconcelos et al.: ESOP'20] to allow mixed choice in timed session types

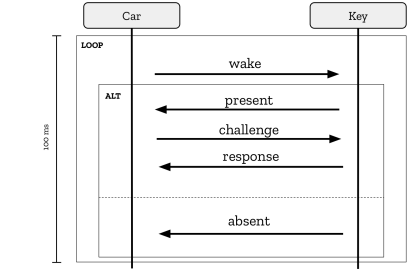
Theorem (Progress)

Well-typed systems have progress

This framework assume urgent input [Murgia:ICE'18,JLAMP'19]

Irachi, Chuang, Hu, Ziarek [OOPSLA'23]
Validating IoT Devices with Range-based session types

Grant Iraci, Cheng-En Chuang, Raymond Hu, and Lukasz Ziarek



- Synchronous binary session types to reason about rates
- Simple extension: periodic recursion

Mind the rates!

$\omega_n t.!m.t \not\sim \omega_n t.!m.!m.t$

Theorem (Soundness)

Well-typed systems do not have rate-errors

– Open Problems –

Resource	Probabilities	Time
• Cost analysis for QoS • Data Dependent QoS	• Relative Expressiveness • Rates	• (Inter)action duration • challenges in CPS • asynchronous subtyping • relativity
tooling	from binary to multiparty	unification

Resource	Probabilities	Time
• Cost analysis for QoS • Data Dependent QoS	• Relative Expressiveness • Rates	• (Inter)action duration • challenges in CPS • asynchronous subtyping • relativity
tooling	from binary to multiparty	unification

Can we use [Bocchi, Honda, Yoshida, Tuosto:Concur'10]?

– Thank you! –

[emilio.tuosto@gssi.it]