# Contracts in Distributed Systems

Massimo Bartoletti

Dipartimento di Matematica e Informatica, Università degli Studi di Cagliari

(joint work with Emilio Tuosto and Roberto Zunino)

# Contracts in Computer Science

In computer science, contracts are typically used at design-time, to drive safe compositions / adaptations of software / services

- contracts are assumed to be **never violated**

$$\text{promises} \implies \text{facts}$$

- contracts dropped after composition
- contract enforcement *outside* the model: *within* the model, it is unspecified how to detect violations and punish responsibles

# Bellum omnium contra omnes

Distributed systems resemble Hobbes' "state of nature", rather than a civil society:

- agent heterogeneity,
- conflicting individual goals,
- unpredictable behaviour

Members of such systems may fail to, or even choose not to, conform to the agreed contracts.

# On the condition of mere nature

*“If a covenant be made [...] in the condition of mere nature (which is a condition of war of every man against every man) upon any reasonable suspicion, it is void. [...]*

*And therefore he which performeth first does but betray himself to his enemy [...]”*

*Thomas Hobbes, The Leviathan, ch. XIV (1651)*

# A simple transaction

Dramatis Personæ:

- **A**, a seller
- **B**, a buyer

Act I: the condition of mere nature

- **A**, I will ship, if you pay
- **B**: I pay
- **A**: Maybe I ship, maybe not

# A safe transaction

Dramatis Personæ:

- **A**, a seller
- **B**, a buyer

Act II: stuckness

- **A**, I will ship, if you pay
- **B**: I will pay, if you ship
- **A**: I do nothing
- **B**: I do nothing

# On the civil society

*“For he that performeth first has no assurance the other will perform after, because the bonds of words are too weak to bridle men’s ambition, avarice, anger, and other passions, without the fear of some coercive power. [...]*

*But in a civil estate, where there a power set up to constrain those that would otherwise violate their faith, [...] he which by the covenant is to perform first is obliged so to do.”*

*Thomas Hobbes, The Leviathan, ch. XIV (1651)*

# Towards a civil society

A calculus for contracting processes, capable of:

- abstracting from the actual contract model: the primitives of the calculus must only rely on a few general contract-related concepts
- providing a common framework where one can compare different contract models
- modelling a wide variety of things that can “go wrong” in contract-oriented computations
- finding violations and dishonest agents

# An abstract contract model (1)

Basic ingredients:

- *principals* **A**, **B**, . . .
- *atoms* a, b, . . .
- *contracts* $c, c', \ldots$

Example: "**A** will ship after **B** has paid"

- contract-as-process: **A** $says$ pay$^-$ . ship$^+$
- contract-as-formula: **A** $says$ (**B** $says$ pay) $\rightarrow$ ship

# An abstract contract model (2)

For all multiset of contracts $\mathsf{C}$, we want to:

- make $\mathsf{C}$ evolve under some actions:

$$\mathsf{C} \xrightarrow{\langle \mathbf{A}_i \ says \ \mathsf{a}_i \rangle_i} \mathsf{C}'$$

- observe some behaviour $\varphi$:

$$\mathsf{C} \vdash \varphi$$

- decide when a principal $\mathbf{A}$ has fulfilled $\mathsf{C}$:

$$\mathsf{C} \text{☺} \mathbf{A}$$

# Primitives for contracts

Basic primitives for contract-oriented computing:

- Advertise a “frozen” contract
- Find an agreement on some observable $\varphi$, and establish a multi-party session among all (and only) the involved parties
- Query a session with an observable $\varphi$
- Do some actions in a session

# The contract calculus $\mathrm{CO}_2$

$$\pi \;::=\; \tau \mid \mathbf{tell}_u \downarrow_{\mathsf{v}} \mathsf{c} \mid \mathbf{fuse}_u\, \varphi \mid \mathbf{ask}_{u,\vec{v}}\, \varphi \mid \mathbf{do}_v\, \mathsf{a}$$

| | | |
|---|---|---|
| $P \;::=$ | $\downarrow_{\mathsf{u}} \mathsf{c}$ | frozen contract |
| | $\sum_{i \in I} \pi_i . P_i$ | summation |
| | $P \mid P$ | parallel composition |
| | $(u)P$ | delimitation |
| | $X(\vec{u})$ | constant |

$$S \;::=\; \mathbf{0} \mid \mathbf{A}[P] \mid s[\mathsf{C}] \mid S \mid S \mid (u)S$$

# $CO_2$ by examples

Dramatis Personæ:

- **A**, a fraudulent seller
- **B**, a naïve buyer

Act IV: fraud

- **A**, I will ship, if you pay
- **B**: I will pay
- **A**: I initiate a session with **B**
- **B**: I do pay
- **A**: I choose wheter swindling **B** or not

# $CO_2$ by examples (1)

$$
\begin{array}{ll}
\mathbf{A}[(x,z) & \mathbf{tell}_{\mathbf{A}} \downarrow_x ((z\ says\ \mathsf{pay}) \rightarrow \mathsf{ship}). \\
 & \mathbf{fuse}_x\ (\mathbf{A}\ says\ \mathsf{ship}). \\
 & \mathbf{ask}_x\ (\mathbf{B}\ says\ !\mathsf{pay}). \\
 & \mathbf{do}_x\ \mathsf{ship} + \mathbf{do}_x\ \mathsf{sheep}\ ] \\
 & \\
| \ \mathbf{B}[(y) & \mathbf{tell}_{\mathbf{A}} \downarrow_y \mathsf{pay}. \\
 & \mathbf{do}_y\ \mathsf{pay}\ ]
\end{array}
$$

# $CO_2$ by examples (2)

$$\begin{aligned}
&\mathbf{A}[(x,z) \quad \downarrow_x \mathbf{A}\ says\ ((z\ says\ \mathsf{pay}) \rightarrow \mathsf{ship})\ |\\
&\qquad\qquad \mathbf{fuse}_x\ (\mathbf{A}\ says\ \mathsf{ship}).\\
&\qquad\qquad \mathbf{ask}_x\ (\mathbf{B}\ says\ !\mathsf{pay}).\\
&\qquad\qquad \mathbf{do}_x\ \mathsf{ship} + \mathbf{do}_x\ \mathsf{sheep}\ ]\\
&|\ \mathbf{B}[(y) \qquad \mathbf{tell}_{\mathbf{A}} \downarrow_y \mathsf{pay}.\\
&\qquad\qquad \mathbf{do}_y\ \mathsf{pay}\ ]
\end{aligned}$$

# $CO_2$ by examples (3)

$$(x, y, z) \quad \mathbf{A}[\ \downarrow_x \mathbf{A}\ says\ ((z\ says\ \mathsf{pay}) \rightarrow \mathsf{ship})\ |$$
$$\downarrow_y \mathbf{B}\ says\ \mathsf{pay}\ |$$
$$\mathbf{fuse}_x\ (\mathbf{A}\ says\ \mathsf{ship}).$$
$$\mathbf{ask}_x\ (\mathbf{B}\ says\ !\mathsf{pay}).$$
$$\mathbf{do}_x\ \mathsf{ship} + \mathbf{do}_x\ \mathsf{sheep}\ ]$$

$$|\ \mathbf{B}[\ \mathbf{do}_y\ \mathsf{pay}\ ]$$

# $CO_2$ by examples (4)

$(s)$ **A**$[$ **ask**$_s$ $($**B** $says$ !pay$)$.
**do**$_s$ ship $+$ **do**$_s$ sheep $]$

$|$ **B**$[$ **do**$_s$ pay $]$

$|$ $s[$ **A** $says$ $(($**B** $says$ pay$) \rightarrow$ ship$)$ $|$
**B** $says$ pay $]$

# $CO_2$ by examples (5)

$$
\begin{array}{lll}
(s) & \mathbf{A}[ & \mathbf{ask}_s\,(\mathbf{B}\ says\ !\mathsf{pay}).\\
 & & \mathbf{do}_s\,\mathsf{ship} + \mathbf{do}_s\,\mathsf{sheep}\ ]\\
 & & \\
 & \mid\ \mathbf{B}[ & \mathbf{0}\ ]\\
 & & \\
 & \mid\ s[ & \mathbf{A}\ says\ ((\mathbf{B}\ says\ \mathsf{pay}) \rightarrow \mathsf{ship})\ \mid\\
 & & \mathbf{B}\ says\ \mathsf{pay}\ \mid\\
 & & \mathbf{B}\ says\ !\mathsf{pay}\ ]
\end{array}
$$

# $CO_2$ by examples (6)

$(s)$ **A**$[$ $\quad$ $\mathbf{do}_s$ ship $+$ $\mathbf{do}_s$ sheep $]$

$\mid$ **B**$[$ $\quad$ $\mathbf{0}$ $]$

$\mid$ $s[$ $\quad$ **A** $says$ $(($**B** $says$ pay$) \rightarrow$ ship$)$ $\mid$

**B** $says$ pay

**B** $says$ !pay $]$

# $CO_2$ by examples (7)

$(s)$ **A**$[\ \mathbf{0}\ ]$

$\mid$ **B**$[\ \mathbf{0}\ ]$

$\mid$ $s[$ **A** $says$ $(($**B** $says$ pay$)$ $\rightarrow$ ship$)$ $\mid$

**B** $says$ pay $\mid$

**B** $says$ !pay $\mid$

**A** $says$ !sheep $]$

**A** is **not** honest: $s[$C$]$ with C $\not\odot$ **A**

# On agreements

A **fuse**$_x$ $\varphi$ chooses variables to be fused with $x$.

Is this choice totally unconstrained?

Let $\varphi$ be the observable "**A** will ship", and:

- $c_1$ = **A** $says$ (**b** $says$ pay) $\rightarrow$ ship
- $c_2$ = **B** $says$ pay
- $c_3$ = **C** $says$ kiss a frog

Shall **C** be forced to kiss a frog ?

# On agreements

A **fuse**$_x$ $\varphi$ chooses variables to be fused with $x$.

Is this choice totally unconstrained?

Let $\varphi$ be the observable "**A** will ship", and:

- $c_1$ = **A** $says$ (**b** $says$ pay) $\rightarrow$ ship
- $c_2$ = **B** $says$ pay
- $c_3$ = **C** $says$ kiss a frog

Shall **C** be forced to kiss a frog ?

No, since **fuse** picks a **minimal** agreement.

# On honesty (and sanctity)

An agent **A**$[P]$ is **honest** in a system $S$ iff:

$\forall$ maximal fair traces $(S_i)_{i \in \mathcal{I}}$ of **A**$[P] \mid S$:

$\forall$ sessions $s$:

$\exists j \in \mathcal{I}$ :

$\forall i \geq j,\ \vec{n},\ \mathsf{C},\ S'$ :

$$S_i \equiv (\vec{n})(s[\mathsf{C}] \mid S') \implies \mathsf{C} \heartsuit \mathbf{A}$$

An agent **A**$[P]$ is a **saint** if honest in **all** $S$.

# On things that may go wrong

$CO_2$ is capable of modelling a wide variety of misbehaviour:

- contract violations by unfulfilled promises
- contract violations by unfair schedulers
- circular contract violations
- goals not reached because contract too weak
- fraudulent contract brokers

# Conclusions

Facts:

- an abstract contract model, 2 instantiations
- encoding (fragments) contracts-as-formulae into contracts-as-processes
- an expressive calculus for contracting systems

Promises:

- static analysis for sanctity
- bridge between contracts-as-processes and contracts-as-formulae

# Thanks!

Questions?

# A guaranteed transaction

Dramatis Personæ:

- **A**, a seller
- **B**, a buyer
- **K**, a broker
- **J**, a judge

All these characters are agents in our system.

# A guaranteed transaction

Act III: justice

- **A**, I tell K: I will ship, if B pays
- **B**: I tell K: I will pay, if A ships
- **K**: A and B have agreed on their contracts. They can now start their transaction.
- **B**: I pay A
- (time passes. . . )
- **J**: A has promised to ship, but he didn't. So, he is culpable and will be punished.