A Distributed Treatment of Exceptions in Multiagent Contracts

Özgür Kafalı and Pınar Yolum

Department of Computer Engineering - Boğaziçi University - Turkey

Declarative Agent Languages and Technologies (DALT)

May 3, 2011
Exceptions

- Commitments are widely used for specifying agent interactions and regulating contracts.
- However, in open MAS, there are no guarantees on how agents will carry out their interactions.
- Exceptions might occur even if the agent carries out its specification correctly, are often identical to commitment violation.
- An agent should be able to
  - monitor its interactions,
  - step in if there is a problem.
To Expect or Not To Expect?
To Expect or Not To Expect?
To Expect or Not To Expect?

pay

deliver in 3 days
To Expect or Not To Expect?

Monday  Tuesday  Wednesday  Thursday  Friday
To Expect or Not To Expect?

Case I: Typical violation

Monday       Tuesday       Wednesday       Thursday       Friday

pay

deliver in 3 days

deadline
To Expect or Not To Expect?

Case I: Typical violation

Monday       Tuesday       Wednesday       Thursday       Friday

pay

deliver in 3 days

deadline
To Expect or Not To Expect?

pay

deliver in 3 days

Monday     Tuesday     Wednesday     Thursday     Friday

Case I: Typical violation
To Expect or Not To Expect?

Case I: Typical violation

Monday  Tuesday  Wednesday  Thursday  Friday

deadline

pay
deliver in 3 days
To Expect or Not To Expect?

Case I: Typical violation

Monday, Tuesday, Wednesday, Thursday, Friday
To Expect or Not To Expect?

Case II: Expect late delivery
To Expect or Not To Expect?

Case II: Expect late delivery
To Expect or Not To Expect?

Case II: Expect late delivery
To Expect or Not To Expect?

Case II: Expect late delivery

Monday | Tuesday | Wednesday | Thursday | Friday

Ozgur Kafali and Pinar Yolum

A Distributed Treatment of Exceptions

DALT 2011
To Expect or Not To Expect?

Case II: Expect late delivery
To Expect or Not To Expect?

Case III: Expect early delivery

Monday      Tuesday      Wednesday      Thursday      Friday

Ozgür Kafalı and Pınar Yolum

A Distributed Treatment of Exceptions

DALT 2011
Introduction

To Expect or Not To Expect?

Case III: Expect early delivery

Monday  Tuesday  Wednesday  Thursday  Friday

deadline

deliver in 3 days

pay

Monday  Tuesday  Wednesday  Thursday  Friday

Case III: Expect early delivery
To Expect or Not To Expect?

Case III: Expect early delivery

Monday  Tuesday  Wednesday  Thursday  Friday

Ozgur Kafali and Pinar Yolum

A Distributed Treatment of Exceptions

DALT 2011
To Expect or Not To Expect?

Case III: Expect early delivery
Motivation

- Do not only look for commitment violations
- Take into account agent’s projections about the future
- Only signal an exception if those projections are not satisfied
Distributed Multigent Architecture

Agent$_i$

Reasoner

Agent$_j$

Reasoner

Agent$_k$

Reasoner

Ozgur Kafali and Pinar Yolum

A Distributed Treatment of Exceptions

DALT 2011
Distributed Multigent Architecture

\[ \text{Protocol}_{ij} \]

\[ \text{pay}(i,j) \rightarrow \text{paid} \]
Distributed Multigent Architecture

\[ \text{Agent}_i \xrightarrow{\text{pay}(i,j) \rightarrow \text{paid}} \xrightarrow{\text{Protocol}_{ij}} \text{Agent}_j \]

\[ \text{Protocol}_{ij} \xrightarrow{\text{active} \rightarrow \text{fulfilled}} \]

\[ \text{Protocol}_{ij} \xrightarrow{\text{enacts}} \text{Agent}_i \xrightarrow{\text{enacts}} \text{Agent}_j \]

\[ \text{Commitment Theory} \]
Distributed Multigent Architecture

Protocol$_{ij}$

Agent$_i$

Reasoner

delivery on Thursday

pay(i,j) → paid

→

Agent$_j$

Reasoner

Protocol$_{ij}$
enacts

Agent$_k$

Reasoner

Commitment Theory

active → fulfilled

uses

enacts

— — — —
Distributed Multigent Architecture

\[ \text{Agent}_i \] \quad \text{Reasoner} \quad \text{enacts} \quad \text{pay}(i,j) \to \text{paid} \quad \text{enacts} \quad \text{Agent}_j \] \quad \text{Reasoner} \\
\text{delivery on Thursday} \quad \text{exception?} \\

\[ \text{Agent}_k \] \quad \text{Reasoner} \quad \text{uses} \quad \text{active} \to \text{fulfilled} \\
\text{Commitment Theory}
States

- World: $S_G^T = \langle \Phi_G, C_G \rangle$
  - $\Phi_G$ is a finite set of atomic propositions that hold at $T$
  - $C_G$ is a finite set of commitments that exist at $T$

- State: $S_A^T = \langle \Phi, C \rangle \subset S_G^T$

- Projected state: $PS_A^T = \langle \Phi_P, C_P \rangle$
Commitments

- Commitment: $C_{A_i, A_j}^{St}(\text{Ant, Con})$

(a) Conditional
- $C^c(Q, P)$
  - conditional
- $C^a(Q, P), Q$
  - active
- $C^v(Q, P), Q$
  - violated
- $C^f(Q, P), P$
  - fulfilled

(b) Base-level
- $C^v(\top, P)$
  - violated
- $C^a(\top, P)$
  - active
- $C^f(\top, P), P$
  - fulfilled
- $C^f(\top, P), P$
  - fulfilled
Fundamentals

- Compare two states and identify if one satisfies the other

\[ X \models Y : "Y \text{ is satisfiable by } X" \]

- \( Y \) represents the minimum satisfactory condition

- \( X \) may include more than the necessary propositions or more beneficial commitments than \( Y \)
Satisfiability Network

\[
paid \land delivered
\]

paid

\[
paid \land delivered
\]

delivered
Satisfiability Network

\[ C^a(\top, \text{paid}) \quad C^f(\top, \text{paid}) \quad C^f(\top, \text{delivered}) \quad C^a(\top, \text{delivered}) \]

\[ C^v(\top, \text{paid}) \quad C^v(\top, \text{delivered}) \]
Satisfiability Network

$C^v(paid, delivered)$

$C^a(paid, delivered)$

$C^f(paid, delivered)$

$\text{paid}$

$\text{paid} \land \text{delivered}$

$\text{delivered}$

$C^a(\top, paid)$

$C^f(\top, paid)$

$C^f(\top, delivered)$

$C^a(\top, delivered)$

$C^v(\top, paid)$

$C^c(paid, delivered)$

$C^v(\top, delivered)$
Satisfiability

Satisfiability Network

\[ C^v(paid, delivered) \]

\[ C^a(paid, delivered) \]

\[ C^f(paid, delivered) \]

\[ paid \]

\[ paid \land delivered \]

\[ delivered \]

\[ C^a(\top, paid) \]

\[ C^f(\top, paid) \]

\[ C^f(\top, delivered) \]

\[ C^a(\top, delivered) \]

\[ C^v(\top, paid) \]

\[ C^c(paid, delivered) \]

\[ C^v(\top, delivered) \]

\[ \text{Ozg"ur Kafali and P"inar Yolum} \]

A Distributed Treatment of Exceptions

DALT 2011
Satisfiability Network

\[ C^\vee(paid, delivered) \]

\[ C^a(paid, delivered) \]

\[ C^f(paid, delivered) \]

\[ \text{paid} \]

\[ \text{paid} \land \text{delivered} \]

\[ \text{delivered} \]

\[ C^a(\top, paid) \]

\[ C^f(\top, paid) \]

\[ C^f(\top, delivered) \]

\[ C^a(\top, delivered) \]

\[ C^v(\top, paid) \]

\[ C^c(paid, delivered) \]

\[ C^v(\top, delivered) \]
Satisfiability Network

\( C^v(paid, delivered) \)

\( C^a(paid, delivered) \)

\( C^f(paid, delivered) \)

\( paid \)

\( paid \land delivered \)

\( delivered \)

\( C^a(\top, paid) \)

\( C^f(\top, paid) \)

\( C^f(\top, delivered) \)

\( C^a(\top, delivered) \)

\( C^v(\top, paid) \)

\( C^c(paid, delivered) \)

\( C^v(\top, delivered) \)

\( \text{Özgür Kafalı and Pınar Yolum} \)

A Distributed Treatment of Exceptions

DALT 2011
Satisfiability Network

\[ C^v(paid, delivered) \]

\[ C^a(paid, delivered) \rightarrow C^v(paid, delivered) \rightarrow C^f(paid, delivered) \]

\[ C^a(\top, paid) \]

\[ C^f(\top, paid) \]

\[ C^f(\top, delivered) \]

\[ C^a(\top, delivered) \]

\[ C^v(\top, paid) \]

\[ C^c(paid, delivered) \]

\[ C^v(\top, delivered) \]

\[ paid \]

\[ paid \land delivered \]

\[ delivered \]

\[ \Box \]

\[ \Diamond \]
Properties

- \( \alpha \models \alpha \) (reflexive)

- not necessarily true that \( \beta \models \alpha \) or \( \beta \not\models \alpha \) if \( \alpha \models \beta \) (non-symmetric)

- \( \alpha \models \gamma \) if \( \alpha \models \beta \) and \( \beta \models \gamma \) (transitive)
When to React?

- State satisfiability based on term satisfiability

$$S^t_{\text{customer}}$$
- paid
- delivered
- \(C^f_{\text{mer,cus}}(\text{paid, delivered})\)

$$PS^t_{\text{customer}}$$
- paid
- delivered

An exception occurs for agent A if a projected state of A is not satisfiable.
Exceptions

Exception: typical violation

$S_{\text{customer}}^{\text{fri}}$

- paid
- $C_{\text{mer,cus}}^{\upsilon}(\text{paid, delivered})$

$\mathcal{P}S_{\text{customer}}^{\text{fri}}$

- paid
- delivered

deadline

Monday Tuesday Wednesday Thursday Friday
No exception: customer does not expect delivery

\[ S_{\text{customer}}^{\text{fri}} \]

- paid
- \( C_{\text{mer,cus}}^{\vee}(\text{paid, delivered}) \)

\[ \mathcal{P}S_{\text{customer}}^{\text{fri}} \]

- paid
- \( C_{\text{mer,cus}}^{\vee}(\top, \text{delivered}) \)

Monday  Tuesday  Wednesday  Thursday  Friday

Exceptions

Özgür Kafalı and Pınar Yolum

A Distributed Treatment of Exceptions

DALT 2011 13 / 16
Exception: customer expects early delivery

\[ S_{\text{customer}}^{\text{thu}} \]
- paid
- \( C_{\text{mer}, \text{cus}}^{\text{a}}(\text{paid, delivered}) \)

\[ \mathcal{P} S_{\text{customer}}^{\text{thu}} \]
- paid
- delivered

Monday  Tuesday  Wednesday  Thursday  Friday

deadline

Özgür Kafalı and Pınar Yolum
A Distributed Treatment of Exceptions

DALT 2011
Customer and merchant’s projections do not match

\[ S_{\text{customer}}^{\text{thu}} \]

- paid
- \( C_{\text{mer,cus}}^a(paid, delivered) \)

\[ \mathcal{P} S_{\text{customer}}^{\text{thu}} \]

- paid
- delivered

\[ S_{\text{merchant}}^{\text{thu}} \]

- paid
- \( C_{\text{mer,cus}}^a(paid, delivered) \)

\[ \mathcal{P} S_{\text{merchant}}^{\text{thu}} \]

- paid
- \( C_{\text{cou,mer}}^a(\top, delivered) \)
Considered exceptions from the agent’s side:
- Proposed a state-oriented approach to describe the agent’s world
- Capture agent’s expectations through projected states
- Built a satisfiability relation to compare projections to actual execution

Reachability relation:
- In the case of an exception, identify if it is recoverable or not
- If the commitment is still active, recovery is possible

Implement the framework with $REC$