Causal Unfolding for Disjunctive Causality
CALCO 2019

Marc de Visme
LIP, ENS Lyon
France

Glynn Winskel
University of Cambridge
United Kingdom

4 june 2019
Representing Causality

- Events

Coffee Machine

- Money
- Start
- Change
- Coffee
- Select
- Error

I. Introduction

Using Event Structure

To model concurrent systems

Game Semantics

Annotations for effects

Hiding

Event structures support hiding.
I. Introduction

Representing Causality
- Events
- Partial Order

Coffee Machine
- Money
- Select
- Start
- Change
- Coffee
- Error

Hiding
Event structures support hiding.
I. Introduction

Representing Causality
- Events
- Partial Order
- Conflict

Coffee Machine
- Money
- Select
- Start
- Change
- Coffee
- Error
I. Introduction

Representing Causality

- Events
- Partial Order
- Conflict

→ Event Structure

Coffee Machine

- Money
- Select
- Start
- Change
- Coffee
- Error

de Visme, Winskel
Causal Unfolding
4 June 2019
I. Introduction

Representing Causality
- Events
- Partial Order
- Conflict
→ Event Structure

Using Event Structure
- Model concurrent systems
- Game Semantics
- Annotations for effects

Coffee Machine
- Money
- Select
- Start
- Change
- Coffee
- Error
I. Introduction

Representing Causality
- Events
- Partial Order
- Conflict
→ Event Structure

Using Event Structure
- Model concurrent systems
- Game Semantics
- Annotations for effects

Coffee Machine
- Money
- Select
- Start
- Change
- Coffee ↘ ↘ Error

Hiding
Event structures support hiding.
I. Introduction

Representing Causality
- Events
- Partial Order
- Conflict
→ Event Structure

Using Event Structure
- Model concurrent systems
- Game Semantics
- Annotations for effects

Hiding
Event structures support hiding.

Coffee Machine

Money
↓
Change
↓
Select
↓
Coffee
adors
↓
Error
Representing Disjunction

General Event Structure

Partial Order

Enabling Relation: 

\{\text{Cash}, \text{Select}\} \vdash \text{Start}

\{\text{Card}, \text{Select}\} \vdash \text{Start}

\{\text{Start}\} \vdash \text{Change, Coffee, Error}

Coffee Machine

- Cash
- Card
- Select

OR

Start

Change

Coffee

Error

de Visme, Winskel
Causal Unfolding
4 June 2019
Representing Disjunction

General Event Structure
- Partial Order
- Enabling Relation:
  \{\text{Cash, Select}\} \vdash \text{Start}
  \{\text{Card, Select}\} \vdash \text{Start}
  \{\text{Start}\} \vdash \text{Change, Coffee, Error}

Coffee Machine

```
Cash  OR  Card  Select
   ↓    ↓      ↓      ↓
Start
   ↓    ↓      ↓      ↓
Change Coffee Error
```
Representing Disjunction

General Event Structure
- Partial Order
- Enabling Relation:
  \( \{\text{Cash, Select}\} \vdash \text{Start} \)
  \( \{\text{Card, Select}\} \vdash \text{Start} \)
  \( \{\text{Start}\} \vdash \text{Change, Coffee, Error} \)

Problem
General event structures do not support hiding.

Coffee Machine
- Cash
- Card
- Select
- Start
- Change
- Coffee
- Error
Problem of Hiding

Small Coffee Machine

Cash

Coffee

After Hiding Cash Fix Coffee, Coffee compatible with Fix does not need Cash. Information Lost: If Fix then Coffee need Cash. → Not representable with "OR". General event structures do not support hiding.
Problem of Hiding

Bugged Coffee Machine

Cash \rightarrow OR \rightarrow Bug

\downarrow \rightarrow Coffee

Coffee compatible with Fix
Coffee does not need Cash
Information Lost: If Fix then Coffee need Cash.
→ Not representable with "OR"

General event structures do not support hiding.
Problem of Hiding

Bugged Coffee Machine

Cash

Bug

OR

Fix

Coffee

Information Lost: If Fix then Coffee need Cash.

→

Not representable with "OR"

General event structures do not support hiding.
Problem of Hiding

Bugged Coffee Machine

Cash ↓ Bug ~~~~ Fix OR
Coffee

After Hiding

Cash

Coffee

Fix

- Coffee compatible with Fix
- Coffee does not need Cash

Information Lost: If Fix then Coffee need Cash.

→ Not representable with "OR"

General event structures do not support hiding.
Problem of Hiding

Bugged Coffee Machine

Cash  Bug  Fix
   \downarrow  \sim \sim \sim
   Coffee

OR

After Hiding

Cash  Fix
       Coffee

- Coffee compatible with Fix
- Coffee does not need Cash
- Information Lost: If Fix then Coffee need Cash.

→ Not representable with “OR”
Problem of Hiding

Bugged Coffee Machine

<table>
<thead>
<tr>
<th>Cash</th>
<th>Bug</th>
<th>Fix</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>OR</td>
</tr>
<tr>
<td>Coffee</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

After Hiding

<table>
<thead>
<tr>
<th>Cash</th>
<th>Fix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coffee</td>
<td></td>
</tr>
</tbody>
</table>

- Coffee compatible with Fix
- Coffee does not need Cash
- Information Lost: If Fix then Coffee need Cash.
  → Not representable with “OR”

General event structures do not support hiding.
Encoding GES into Event Structures

Usual Solution
- Duplicate events
- Put copies of the same event in conflict

Bugged Coffee Machine

Cash

Bug ~~~~ Fix

Coffee_1 ~~~~ Coffee_2
Encoding GES into Event Structures

**Usual Solution**
- Duplicate events
- Put copies of the same event in conflict

**Bugged Coffee Machine**

```
Cash
↓
Coffee₁  ~~~~  Coffee₂  Bug  ~~~~  Fix
```

**After Hiding**

```
Cash
↓
Coffee₁  ~~~~  Coffee₂  Fix
```
Encoding GES into Event Structures

Usual Solution
- Duplicate events
- Put copies of the same event in conflict

Bugged Coffee Machine

Cash
↓
Coffee_1 ~ Coffee_2

Bug ~ Fix

After Hiding

Cash
↓
Coffee_1 ~ Coffee_2

Fix

Event structures support hiding.
Problem with this Encoding

Coffee Machine

Cash \rightarrow Coffee_1

Card \rightarrow Coffee_2

Problems

\[ P(\text{Coffee}_1) = 1 \]
\[ P(\text{Coffee}_2) = 1 \]
\[ 1 + 1 \not\geq 1 \]
Problem with this Encoding

Problems
- Not an adjunction
- Is too restrictive with probabilities

Coffee Machine

Cash \downarrow \quad Card
\quad Coffee_1 \sim \sim \quad Coffee_2

Solution
Allow non conflicting copies.
Problem with this Encoding

Problems
- Not an adjunction
- Is too restrictive with probabilities

Polarities
- ⊓ Inputs
- ⊔ Outputs

Coffee Machine

Cash\(^-\) \quad \text{Card}\(^-\)
\quad \downarrow \quad \downarrow
\quad \text{Coffee}_1^+ \quad \sim \quad \text{Coffee}_2^+$

Probabilities (simplified)

\[ P(Coffee_1) = 1 \]
\[ P(Coffee_2) = 1 \]
\[ 1 + 1 \not\geq 1 \]

Solution
Allow non conflicting copies.
Problem with this Encoding

Problems
- Not an adjunction
- Is too restrictive with probabilities

Polarities
- ⊙ Inputs
- ⊕ Outputs

Probabilities (simplified)
- Probabilities on ⊕
- Conflict $\implies$ sum $\leq 1$

Coffee Machine

\[
\begin{align*}
\text{Cash}^- & \quad \text{Card}^- \\
\downarrow & \quad \downarrow \\
\text{Coffee}_1^+ & \sim \sim \sim \text{Coffee}_2^+
\end{align*}
\]
Problem with this Encoding

Problems
- Not an adjunction
- Is too restrictive with probabilities

Polarities
- ⊗ Inputs
- ⊕ Outputs

Probabilities (simplified)
- Probabilities on ⊕
- Conflict ⟷ sum ≤ 1

Coffee Machine

Cash⁻ \[ \rightarrow \] Card⁻

Coffee₁⁺ ~ ~ ~ Coffee₂⁺

\[ P(Coffee₁) = 1 \]
\[ P(Coffee₂) = 1 \]
\[ 1 + 1 \not\leq 1 \]
Problem with this Encoding

**Problems**
- Not an adjunction
- Is too restrictive with probabilities

**Polarities**
- ⊗ Inputs
- ⊕ Outputs

**Probabilities (simplified)**
- Probabilities on ⊕
- Conflict  \( \implies \text{sum} \leq 1 \)

**Coffee Machine**

\[
\begin{align*}
\text{Cash}^- \quad & \quad \text{Card}^- \\
\downarrow \quad & \quad \downarrow \\
\text{Coffee}^+_1 \quad & \quad \text{Coffee}^+_2
\end{align*}
\]

\[
P(\text{Coffee}_1) = 1 \\
P(\text{Coffee}_2) = 1 \\
1 + 1 \not\leq 1
\]

**Solution**
Allow non conflicting copies.
The Solution (CSL 2017)

Event Structure with Equivalence
- Duplicate events
- $\equiv$ relates copies of the same event

Bugged Coffee Machine

Cash \quad Bug \rightsquigarrow \quad Fix

Coffee_1 \equiv Coffee_2
Event Structure with Equivalence

- Duplicate events
- \( \equiv \) relates copies of the same event

**Bugged Coffee Machine**

Cash

\[ \Downarrow \]

Bug \( \sim \sim \sim \) Fix

\[ \Downarrow \]

Coffee_1 \( \equiv \equiv \) Coffee_2

**After Hiding**

Cash

\[ \Downarrow \]

Coffee_1 \( \equiv \equiv \) Coffee_2

Fix
The Solution (CSL 2017)

**Event Structure with Equivalence**
- Duplicate events
- $\equiv$ relates copies of the same event

**Bugged Coffee Machine**

<table>
<thead>
<tr>
<th>Cash</th>
<th>Bug</th>
<th>Fix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coffee$_1$</td>
<td>$\equiv$</td>
<td>Coffee$_2$</td>
</tr>
</tbody>
</table>

**After Hiding**

<table>
<thead>
<tr>
<th>Cash</th>
<th>Fix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coffee$_1$</td>
<td>$\equiv$</td>
</tr>
</tbody>
</table>

Event structures with equivalence support hiding.
II. Translation from GES to ESE
Motivating Example

Small Conference

Ana $\rightarrow$ OR $\rightarrow$ Welcome $\leftarrow$ Markus

Welcome $\downarrow$ AND $\downarrow$ Start
Motivating Example

Small Conference
- Ana
  - OR
  - Welcome
  - AND
  - Start

Markus

After Causal Unfolding
- Ana
  - Welcome
  - Start

Markus
- Welcome
  - Start
Motivating Example

Small Conference

Ana OR Markus
Welcome

Ana AND Markus
Start

Prime Extremal Realisations

Ana
Welcome
Start

Markus
Welcome
Start
Realisations

- partial order
- respecting the enabling
- and conflict-free

Realisation

\[ (R, \leq) \preceq (S, \leq) \] if:

\[ f : R \rightarrow S \text{ bijective} \]

\[ r \leq R \Rightarrow f(r) \leq S f(r') \]
Extremal Realisations

**Realisation**
- partial order
- respecting the enabling
- and conflict-free

**Constraint Order**
\[(R, \leq_R) \preceq (S, \leq_S) \text{ if:}
\]
- \( f : R \to S \) bijective
- \( r \leq_R r' \implies f(r) \leq_S f(r') \)
Extremal Realisations

Realisation
- partial order
- respecting the enabling
- and conflict-free

Constraint Order
\((R, \leq_R) \preceq (S, \leq_S)\) if:
- \(f : R \rightarrow S\) bijective
- \(r \leq_R r' \implies f(r) \leq_S f(r')\)

Extremal Realisation
Minimal for \(\preceq\).
Prime Extremal Realisations

Definition

Extremal realisation with a maximum element.
Prime Extremal Realisations

Definition
Extremal realisation with a maximum element.

Small Conference

Prime Extremal Realisations

Ana

Welcome

Start

Markus

Welcome

Start
Sum up of the Translation

Prime Extremal Realisation = Causal History of an Event

III. CATEGORIES
Maps of Event Structures

Labelled Transition System

$f$ functional simulation if:

\[ s \xrightarrow{\ell} s' \implies f(s) \xrightarrow{f(\ell)} f(s') \]
Maps of Event Structures

**Labelled Transition System**

$f$ functional simulation if:

\[ s \rightarrow_{\ell} s' \implies f(s) \rightarrow_{f(\ell)} f(s') \]

**Event Structures** \((E, \leq_E, \sim_E)\)

Configurations \(x \in C(E)\):

- \(a \leq_E b \in x \implies a \in x\)
- \(x\) is conflict free

Extension \(x \leftarrow a \subset x \cup \{a\}\)
Maps of Event Structures

Maps of Event Structures

$f$ map of event structures if:

\[ x \subset y \implies f(x) \subset f(y) \]

Event Structures \((E, \leq_E, \sim_E)\)

Configurations \(x \in C(E)\):

- \(a \leq_E b \in x \implies a \in x\)
- \(x\) is conflict free

Extension \(x \subset x \cup \{a\}\)

\[ \text{Cash} \quad \text{Coffee} \sim \sim \text{Chocolate} \quad \text{Cup} \]

\[ \text{Cash} \quad \text{Coffee} \quad \text{Cup} \]

\[ \text{Cash} \quad \text{Chocolate} \quad \text{Tea} \]
Maps of Event Structures

Maps of Event Structures

\[ f \text{ map of event structures if:} \]

\[ x \triangleleft \subset y \implies f(x) \triangleleft \subset f(y) \]

Event Structures \((E, \leq_E, \sim_E)\)

Configurations \(x \in C(E)\):

- \(a \leq_E b \in x \implies a \in x\)
- \(x\) is conflict free

Extension \(x \triangleleft \subset x \cup \{a\}\)

Maps of Event Structures

\[ f \text{ map of event structures if:} \]

- \(x \in C(E) \implies f(x) \in C(E')\)
- \(\forall x \in C(E), f\) injective on \(x\)
The Adjunction

Maps of Event Structures

- $f$ map of event structures if:
  - $x \in C(E) \implies f(x) \in C(E')$
  - $\forall x \in C(E), \forall a, b \in x,$
    - $f(a) = f(b) \implies a = b$
# The Adjunction

## Maps of Event Structures

A map of event structures $f$ if:

- $x \in C(E) \implies f(x) \in C(E')$
- $\forall x \in C(E), \forall a, b \in x$, $f(a) = f(b) \implies a = b$

## Maps of GES

A map of GES $f$ if:

- $x \in C(E) \implies f(x) \in C(E')$
- $\forall x \in C(E), \forall a, b \in x$, $f(a) = f(b) \implies a = b$
# The Adjunction

## Maps of ESE

### $f$ map of ESE if:

- $x \in C(E) \implies f(x) \in C(E')$
- $\forall x \in C(E), \forall a, b \in x, f(a) \equiv f(b) \implies a \equiv b$
- $a \equiv b \implies f(a) \equiv f(b)$

## Maps of GES

### $f$ map of GES if:

- $x \in C(E) \implies f(x) \in C(E')$
- $\forall x \in C(E), \forall a, b \in x, f(a) = f(b) \implies a = b$
## The Adjunction

### Maps of ESE

- \( f \) map of ESE if:
  - \( x \in C(E) \implies f(x) \in C(E') \)
  - \( \forall x \in C(E), \forall a, b \in x, \quad f(a) \equiv f(b) \implies a \equiv b \)
  - \( a \equiv b \implies f(a) \equiv f(b) \)

### Maps of GES

- \( f \) map of GES if:
  - \( x \in C(E) \implies f(x) \in C(E') \)
  - \( \forall x \in C(E), \forall a, b \in x, \quad f(a) = f(b) \implies a = b \)

### A 2-category

\[
  f \equiv g \iff \forall e, f(e) \equiv g(e)
\]
The Adjunction

Maps of ESE

\[ f \text{ map of ESE if:} \]
- \( x \in C(E) \implies f(x) \in C(E') \)
- \( \forall x \in C(E), \forall a, b \in x, \quad f(a) \equiv f(b) \implies a \equiv b \)
- \( a \equiv b \implies f(a) \equiv f(b) \)

Maps of GES

\[ f \text{ map of GES if:} \]
- \( x \in C(E) \implies f(x) \in C(E') \)
- \( \forall x \in C(E), \forall a, b \in x, \quad f(a) = f(b) \implies a = b \)

A 2-category

\[ f \equiv g \iff \forall e, f(e) \equiv g(e) \]

Pseudo-adjunction

\[
\begin{align*}
\text{ESE} & \cong \text{collapse} \quad \text{PER} \quad \text{GES} \\
\text{Hom}(\text{collapse}(A), B) & \cong \text{Hom}(A, \text{PER}(B))/\equiv
\end{align*}
\]
Conclusion

Partial Orders

- Hiding
- Game Semantics
- Event Structures
- No proper disjunction
- ESE
- Strategies with Parallel Causes (CSL17)

Enabling Relations

- GES
- Disjunction
- No Hiding

Pseudo-adjunction

\[ ESE \equiv \begin{array}{c}
\text{collapse} \\
\perp & \text{PER} \\
\end{array} \Rightarrow GES \]

Extensions

Causal unfolding of other models with disjunctive causality via extremal realisations.

Thank you.