Strategies as Concurrent Processes

Glynn Winskel

With thanks to Marc Lasson

Once strategies are made concurrent (essentially by replacing the role of trees in games by that of event structures) they form a concurrent process language.

Happy Birthday Pierre-Louis!

Fest for Pierre-Louis Curien, Venice, September 2013
Event structures and their maps

An event structure comprises \((E, \leq, \text{Con})\), events \(E\), a partial order of causal dependency \(\leq\), and consistency a family \(\text{Con}\) of finite subsets of \(E\), s.t. ...

Its finite configurations \(\mathcal{C}(E)\) comprise those finite subsets \(x \subseteq E\) for which \(x \in \text{Con}\) and \(x\) is down-closed i.e. \(e' \leq e \in x \Rightarrow e' \in x\).

A map of event structures \(f : E \rightarrow E'\) is a partial function \(f : E \rightarrow E'\) such that for all \(x \in \mathcal{C}(E)\)

\[ fx \in \mathcal{C}(E') \text{ and } e_1, e_2 \in x \& f(e_1) = f(e_2) \Rightarrow e_1 = e_2. \]

Note that when \(f\) is total it restricts to a bijection \(x \cong fx\), for any \(x \in \mathcal{C}^\infty(E)\).

Maps preserve concurrency and locally reflect causal dependency.
Defined part of a map

A partial map

\[ f : E \to E' \]

of event structures has **partial-total factorization** as a composition

\[ E \xrightarrow{p} E \downarrow V \xrightarrow{t} E' \]

where \( V = \text{def} \{ e \in E \mid f(e) \text{ is defined} \} \) is the domain of definition of \( f \);

\( E \downarrow V = \text{def} (V, \leq_V, \text{Con}_V) \), where

\( v \leq_V v' \) iff \( v \leq v' \) \& \( v, v' \in V \) \quad \text{and} \quad X \in \text{Con}_V \) iff \( X \in \text{Con} \) \& \( X \subseteq V \);

the **partial** map \( p : E \to E \downarrow V \) acts as identity on \( V \) and is undefined otherwise;

and the **total** map \( t : E \downarrow V \to E' \), called the **defined part** of \( f \), acts as \( f \).
Pullbacks of total maps event structures

Total maps $f : A \to C$ and $g : B \to C$ have pullbacks in the category of event structures:

$$
\begin{array}{ccc}
P & \overset{\pi_1}{\searrow} & \overset{\pi_2}{\swarrow} \\
& A & \\
& f & \swarrow \\
& C & \searrow g \\
& & B \\
\end{array}
$$

Finite configurations of $P$ correspond to the composite bijections

$$
\theta : x \cong fx = gy \cong y
$$

between configurations $x \in \mathcal{C}(A)$ and $y \in \mathcal{C}(B)$ s.t. $fx = gy$ for which the transitive relation generated on $\theta$ by $(a, b) \leq (a', b')$ if $a \leq_A a'$ or $b \leq_B b'$ is a partial order.
Concurrent games and strategies

A game is represented by an event structure $A$ in which an event $a \in A$ carries a polarity $\text{pol}(a)$, $+$ (Player) and $-$ (Opponent).

A (nondeterministic concurrent) strategy in a game $A$ is represented by a total map of event structures $\sigma : S \to A$ which preserves polarities and is

**Receptive:** $\sigma(x) \preceq y$ implies there is a unique $x' \in C(S)$ such that $x \preceq x' \& \sigma(x') = y$:

\[
\begin{array}{c}
x \preceq x' \& \sigma(x') = y \\
\sigma(x) \preceq y
\end{array}
\]

A strategy should be receptive to all possible moves of opponent.

**Innocent:** if $s \rightarrow_S s' \& (\text{pol}(s) = + \text{ or } \text{pol}(s') = -)$ then $\sigma(s) \rightarrow_A \sigma(s')$.

A strategy should only adjoin immediate causal dependencies $\Theta \rightarrow \Theta$.

[The relation $\rightarrow$ stands for immediate causal dependence]
Strategies between games

Following Conway & Joyal, a **strategy from** a game $A$ to a game $B$, written $\sigma : A \rightarrow B$, is a strategy $\sigma$ in the game $A^\perp \parallel B$, based on:

(Simple) Parallel composition $A \parallel B$, by juxtaposition. Its unit is the empty game $\emptyset$.

Dual $A^\perp$, of an event structure with polarity $A$: a copy of the event structure $A$ with a reversal of polarities.

The conditions on strategies exactly ensure that copy-cat is identity w.r.t. composition.
Example of a strategy: copy-cat strategy from $A$ to $A$

$$\gamma_A : \text{CC}_A \to A^\perp \parallel A$$

\begin{center}
\begin{tikzpicture}
\node (a1) at (0,0) {$\overline{a}_1$};
\node (a2) at (1,0) {$\overline{a}_2$};
\node (b1) at (0,1) {$\overline{a}_1$};
\node (b2) at (1,1) {$\overline{a}_2$};
\node (c1) at (0,2) {$A^\perp$};
\node (c2) at (1,2) {$A$};
\node (d1) at (2,2) {$A$};
\node (d2) at (2,1) {$A^\perp$};
\draw (a1) -- (b1); \draw (a2) -- (b2);
\draw (a1) -- (c1); \draw (a2) -- (c2);
\draw (b1) -- (c1); \draw (b2) -- (c2);
\end{tikzpicture}
\end{center}
Composition of strategies $\sigma : S \to A^\perp \parallel B$, $\tau : T \to B^\perp \parallel C$

Via pullback. Ignoring polarities, the composite partial map

\[
\begin{array}{c}
\text{has defined part, which yields } \tau \circ \sigma : T \circ S \to A^\perp \parallel C \text{ once reinstate polarities.}
\end{array}
\]

(Later we shall use composition without hiding $\tau \otimes \sigma$)
An alternative characterization of strategies

Defining a partial order — the Scott order — on configurations of $A$

$$x \sqsubseteq_A y \iff x \sqsubseteq \cdot \subseteq \cdot \sqsubseteq \cdot \subseteq \cdot \sqsubseteq \cdot \subseteq^+ y.$$ 

we obtain a factorization system $((\mathcal{C}(A), \sqsubseteq_A), \sqsupseteq, \sqsubseteq^+)$, i.e. $\exists! z. x \sqsubseteq z$.

Theorem Strategies $\sigma : S \to A$ correspond to discrete fibrations

$$\sigma : (\mathcal{C}(S), \sqsubseteq_S) \to (\mathcal{C}(A), \sqsubseteq_A), \text{ i.e. } \exists! x'. x' \sqsubseteq_S x \quad \sigma \quad \exists \sigma''(x),$$

preserving $\sqsubseteq$, $\subseteq^+$ and $\emptyset$.

$\leadsto$ A lax functor from strategies to profunctors ...
Research directions

Concurrent strategies between games form a bicategory rich in structure. Concurrent strategies are at the crossroads of several areas.

They extend to games with winning conditions or payoff, imperfect information, probabilistic and quantum games.

But concurrent strategies are linear. One reason to introduce symmetry in games and strategies. Then can support (co)monads up to symmetry for copying, to break linearity. There are relations with homotopy.

Here: concurrent strategies from a concurrent-process point of view. In fact, many of the operations on strategies I’ll describe have been useful in the proofs of determinacy and value theorems.
Operations on strategies

Write e.g. $\sigma : A, B$ to mean $\sigma$ is a strategy in game $A \parallel B$.

**Composition** $\sigma \odot \tau : A, C$, if $\sigma : A, B$ and $\tau : B^\perp, C$.

**Simple parallel composition** $\sigma \parallel \tau : A, B$, if $\sigma : A$ and $\tau : B$.
(Special case of synchronized composition, writing $\sigma : A^\perp \rightarrow \emptyset$, $\tau : \emptyset \rightarrow B$)

**Sums of strategies** $\bigsqcup_{i \in I} \sigma_i : A$, for $\sigma_i : A$ where $i \in I$.
The operation makes the $+$-events of different components conflict and identifies initial $-$-ve events of the components over a common events in $A$. 
Operations on strategies cont.1

Relabelling $f_! \sigma : B$, if $\sigma : A$ and $f : A \to B$, possibly partial, is receptive and innocent. The composition of maps $f \sigma : S \xrightarrow{\sigma} A \xrightarrow{f} B$ is receptive and innocent. Its defined part, $f_! \sigma : B$, is given by the partial-total factorization

$$
S \xrightarrow{f_\sigma} S_0 \xrightarrow{f_! \sigma} B,
$$

is a strategy over $B$. Relabelling generalizes to (partial) functions $f$ which are just innocent—it is easy to make $f_! \sigma$ receptive.
Operations on strategies cont.2

**Pullback** $f^*\sigma : A$, if $\sigma : B$ and $f : A \to B$ is a map of event structures, possibly partial. The strategy $f^*\sigma$ is got by the pullback

$$
\begin{array}{ccc}
S' & \longrightarrow & S \\
\downarrow f^*\sigma & & \downarrow \sigma \\
A & \longrightarrow & B.
\end{array}
$$

Pullback along $f : A \to B$ may introduce events and causal links, present in $A$ but not in $B$. The operation subsumes prefixing by an event.

*For innocent partial $f$ get $f! \dashv f^*$. When does $f^*$ have a right adjoint?*
Operations on strategies cont.3

**Sum types** If $A_i, i \in I$, is a countable family of games, we can form their sum, the game $\sum_{i \in I} A_i$ as the coproduct of event structures.

**Injection** If $\sigma : A_j$, for $j \in I$, we can create the strategy $j \sigma : \sum_{i \in I} A_i$ in which $\sigma$ is extended with those initial -ve events needed to maintain receptivity.

**Projection** A strategy of sum type $\sigma : \sum_{i \in I} A_i$ projects to a strategy $(\sigma)_j : A_j$ where $j \in I$.

Projection and injection are special cases of pullback and relabelling.

**Abstraction** $\lambda x : A. \sigma : A \rightarrow B$. Because strategies form a monoidal-closed bicategory, with tensor $A \| B$ and function space $A \rightarrow B = \text{def } A^\perp \| B$, they support a (linear) $\lambda$-calculus, so process-passing.

**Recursive** types and processes, along standard lines.
Issues

The operations form the basis of a higher-order process language.

But,

- what is its operational semantics?

- what are suitable equivalences? (E.g. w.r.t. “may” and “must” testing)

- its expressivity?

These require we examine the effects of synchronization and the internal, neutral events it produces, more carefully.
Deadlocks in composition

Composition of strategies can introduce deadlock which is presently undetected:

**Example 1** Deadlock may arise in a composition \( \tau \odot \sigma \) through \( \sigma : A \rightarrow B \) and \( \tau : B \rightarrow C \) imposing incompatible causal dependencies between events in \( B \).

**Example 2** For games \( B = \oplus \| \oplus \) and \( C = \ominus \),

strategy \( \sigma_1 : \emptyset \rightarrow B \) nondeterministically chooses the right or left move in \( B \),
strategy \( \sigma_2 : \emptyset \rightarrow B \) chooses just the right move in \( B \),
strategy \( \tau : B \rightarrow C \) yields output in \( C \) if gets the right event of \( B \) as input.

The two strategy compositions \( \tau \odot \sigma_1 \) and \( \tau \odot \sigma_2 \) are indistinguishable.
Partial strategies

A partial strategy in a game $A$ (in which all events have +ve or −ve polarity) comprises a (partial) map $\sigma : S \rightarrow A$ of event structures with polarity (in which $S$ may also have neutral events) which

(i) is receptive;
(ii) has domain of definition the non-neutral events of $S$;
(iii) partial-total factorization in which the defined part $\sigma_0$ is a strategy:

$$
\begin{array}{c}
S \\
\sigma
\end{array}
\xrightarrow{\sigma_0}
\begin{array}{c}
S_0 \\
A
\end{array}
$$

The operations on strategies extend to partial strategies—though the use of $pb$ is replaced by a stricter variant of $pb$ when working with partial maps.
Operational semantics

A partial strategy $\sigma : A$, denoting $\sigma : S \to A$, is associated with transitions

\[(i) \ A : \sigma \xrightarrow{a} \sigma' : A/a \quad \text{and} \quad (ii) \ A : \sigma \xrightarrow{\ast} \sigma' : A,\]

meaning

(i) $\sigma$ can play the initial move $a$ and resume as the partial strategy $\sigma'$ in the sub-game $A/a$, after playing $a$,

(ii) $\sigma$ can make an initial neutral move, invisible in the game $A$, and resume as the partial strategy $\sigma'$.

Notation

Use $\alpha, \beta, \gamma$ for an initial move $a,b,c$ of games $A, B, C$ or invisible move $\ast$. Take the undefined of partial functions to be $\ast$. 

17
Operational semantics cont.1

Composition (without hiding)

\[
\frac{A, B : \sigma \xrightarrow{\alpha} \sigma' : A/\alpha, B}{A, C : \tau \otimes \sigma \xrightarrow{\alpha} \tau' \otimes \sigma' : A/\alpha, C}
\]

\[
\frac{B^{\perp}, C : \tau \xrightarrow{\gamma} \tau' : B, C/\gamma}{A, C : \tau \otimes \sigma \xrightarrow{\gamma} \tau' \otimes \sigma' : A, C/\gamma}
\]

\[
\frac{A, B : \sigma^{(2,b)} \xrightarrow{\alpha} \sigma' : A, B/b}{A, C : \tau \otimes \sigma ^* \xrightarrow{\gamma} \tau' \otimes \sigma' : A, C}
\]

Without typing,

\[
\frac{\sigma \xrightarrow{\alpha} \sigma'}{\tau \otimes \sigma \xrightarrow{\alpha} \tau' \otimes \sigma'}
\]

\[
\frac{\tau \otimes \sigma \xrightarrow{\gamma} \tau' \otimes \sigma'}{\tau \otimes \sigma \xrightarrow{\gamma} \tau' \otimes \sigma'}
\]
Operational semantics cont.2

**Relabelling**, w.r.t. receptive and innocent (partial) \( f : A \to B \),

\[
\frac{A : \sigma \xrightarrow{\alpha} \sigma' : A/\alpha}{B : f!\sigma \xrightarrow{f\alpha} (f/\alpha)!\sigma' : B/f\alpha}
\]

Without typing,

\[
\frac{\sigma \xrightarrow{\alpha} \sigma'}{f!\sigma \xrightarrow{f\alpha} (f/\alpha)!\sigma'}
\]
Pullback w.r.t. (partial) \( f : A \to B \), and \( a \) an initial move of \( A \),

\[
\begin{align*}
B : \sigma \xrightarrow{f^a} \sigma' : B/f^a \\
A : f^* \sigma \xrightarrow{a} (f/a)^* \sigma' : A/a
\end{align*}
\]

\( f(a) \) is defd

\[
\begin{align*}
A : f^* \sigma \xrightarrow{a} (f/a)^* \sigma : A/a
\end{align*}
\]

\( f(a) \) is undefd

Without typing,

\[
\begin{align*}
\sigma \xrightarrow{f^a} \sigma' \\
f^* \sigma \xrightarrow{a} (f/a)^* \sigma' \\
f^* \sigma \xrightarrow{a} (f/a)^* \sigma
\end{align*}
\]

\( f(a) \) is defd

\[
\begin{align*}
f^* \sigma \xrightarrow{a} (f/a)^* \sigma
\end{align*}
\]

\( f(a) \) is undefd

\[
\sigma \xrightarrow{*} \sigma'
\]

\[
\begin{align*}
f^* \sigma \xrightarrow{*} f^* \sigma'
\end{align*}
\]
Operational semantics cont.4

**Sum of strategies**, without typing,

\[
\begin{align*}
\sigma_i \xrightarrow{a} \sigma'_i, & \quad i \in I & \text{a is } -ve \\
\bigsqcap_{i \in I} \sigma_i \rightarrow \bigsqcap_{i \in I} \sigma'_i & \quad j \in I \\
\sigma_j \xrightarrow{*} \sigma'_j & \\
\bigsqcap_{i \in I} \sigma_i \rightarrow (\bigsqcap_{i \in I} \sigma_i)[\sigma'_j/j] & \\
\sigma_j \xrightarrow{a} \sigma'_j & \quad j \in I \& \text{ a is } +ve \\
\bigsqcap_{i \in I} \sigma_i \rightarrow \sigma'_j &
\end{align*}
\]
Operational semantics cont.5

Assume certain primitive strategies $\sigma_0 : A$, so as a map $\sigma_0 : S \rightarrow A$, for which we assume a rule

$$\begin{array}{c}
A : \sigma_0 \xrightarrow{\alpha} \sigma'_0 : A/\alpha \\
\hline
s \text{ is initial in } S \& \sigma_0(s) = \alpha
\end{array}$$

Then, derivations in the operational semantics of

$$A : \sigma \xrightarrow{\alpha} \sigma' : A/\alpha,$$

in which $\sigma$ denotes the partial strategy $\sigma : S \rightarrow A$, are in 1-1 correspondence with initial events $s$ in $S$ such that $\sigma(s) = \alpha$. 

22
Blocking configurations

For ‘may’ and ‘must’ equivalence it is not necessary to use partial strategies; it’s sufficient to carry with a strategy the extra structure of ‘blocking’ configurations (＝ images of +/0-maximal configurations in a partial strategy). Composition on strategies extends to composition on strategies with blocking configurations.

Let $\sigma : S \to A$ be a partial strategy. Its defined part is a strategy $\sigma_0 : S \sigma_0 \to S_0$.

Define the (possibly) blocking configurations in $C^\infty(S_0)$ to be

$\text{Block}(\sigma) = \{ p \cdot x \mid x \in C^\infty(S) \text{ is } +/0\text{-maximal} \}$.

This operation preserves composition.