A Geometrical Interpretation of Asynchronous Computability

joint ongoing work with
Éric Goubault and Samuel Mimram

Christine Tasson
Christine.Tasson@pps.univ-paris-diderot.fr

Laboratoire PPS - Université Paris Diderot
A Geometrical Interpretation of Asynchronous Computability

$U_1 \ U_0 \ S_1 \ S_0 \ U_2 \ S_2$

Interleaving Trace

Simplex

Dipath

Interval Order
Distributed System:
A fix family of $n + 1$ processes communicate by Update and Scan of their local memory into a shared global memory.

Asynchronous:
- For each process, the $k$th Scan follows the $k$th Update
- Update and Scan are mutually exclusive
- no delay or order restriction

Interleaving Trace:
Each execution of a protocol is given by an interleaving trace $T \in \{U_i, S_i \mid i \in [n] = \{0 \cdots n\}\}^*$ well-bracketted.

3 processes, 2 rounds: $U_1 \ U_2 \ S_1 \ U_0 \ S_0 \ S_2 \ U_1 \ U_0 \ S_1 \ U_2 \ S_2 \ S_0$
Consider a program with \( n + 1 \) processes and \( (r_i)_{i \in [n]} \) rounds.

**State:** a pair \( s = (\ell, m) \) where

- \( \ell = (\ell_i)_{i \in [n]} \) local memories (one register by process)
- \( m = (m_i)_{i \in [n]} \) global memory (one register by process)

Initial state \( s_0: \ell_i = i \) and \( m_i = \perp \)

**Semantics:**

**Update:** \( i \) updates its local view into the global memory

\[
(\ell_0 \ldots \ell_i \ldots \ell_n, m_0 \ldots m_i \ldots m_n) \xrightarrow{U_i} (\ell_0 \ldots \ell_i \ldots \ell_n, m_0 \ldots \ell_i \ldots m_n)
\]

**Scan:** \( i \) scans the global memory into its local view

\[
(\ell_0 \ldots \ell_i \ldots \ell_n, m) \xrightarrow{S_i} (\ell_0 \ldots m \ldots \ell_n, m)
\]
Operational Semantics: Examples

2 processes, 2 rounds: \( U_0 U_1 S_1 S_0 U_0 S_0 U_1 S_1 \)
Definition:
Two interleaving traces $T$, $T'$ are operationally equivalent when
\[ s_0 \xrightarrow{T}^* s \iff s_0 \xrightarrow{T'}^* s \]

Generators:
The interleaving trace equivalence $\approx$ is the smallest congruence on well-bracketed words in $\{U_i, S_i \mid i \in [n]\}^*$ such that
\[ U_i U_j \approx U_j U_i \quad \text{and} \quad S_i S_j \approx S_j S_i \]

Proof Sketch:
Operational Equivalence

Definition:
Two interleaving traces $T$, $T'$ are operationally equivalent when

$$s_0 \xrightarrow{T}^* s \iff s_0 \xrightarrow{T}^* s$$

Generators:
The interleaving trace equivalence $\approx$ is the smallest congruence on well-bracketed words in $\{U_i, S_i \mid i \in [n]\}^*$ such that

$$U_i U_j \approx U_j U_i \quad \text{and} \quad S_i S_j \approx S_j S_i$$

Proof Sketch:
A Geometrical Interpretation of Asynchronous Computability

\[ \left[ U_1 U_0 S_1 S_0 U_2 S_2 \right] \]

Interleaving Trace/≈

Simplex

Dipath/↭

Interval Order

C. Tasson
From Interleaving Traces to Interval Order

\[ i^k \prec j^\ell \iff s^k_i < u^\ell_j \]

\[ i^k \prec j^\ell \iff s^k_i < u^\ell_j \]
Consider a program with \( n + 1 \) processes and \( (r_i)_{i \in [n]} \) rounds.

**[n]-Colored Interval Order:** \( X = \{ i^k \mid k \in [r_i], \ i \in [n] \} \) with

- a partial order \( \prec \) induced by intervals \( i^k = [u^k_i, s^k_i] \)
- restriction to any process \( i \) is a total order:

\[
\begin{align*}
0^r_0 & \rightarrow 1^r_1 \rightarrow \cdots \rightarrow n^r_n \\
0^2 & \rightarrow 1^2 \rightarrow \cdots \rightarrow n^2 \\
0^1 & \rightarrow 1^1 \rightarrow \cdots \rightarrow n^1 \\
0^0 & \rightarrow 1^0 \rightarrow \cdots \rightarrow n^0
\end{align*}
\]

\[
\begin{align*}
i^k < j^l & \iff s^k_i < u^l_j \\
u^k_i < s^k_i \\
i^k < i^{k+1}
\end{align*}
\]

**Theorem [Fishburn]:** Interval orders are exactly the \((2 + 2)\)-free posets,
From Interval Order to Interleaving Traces

\[ i^k \prec j^\ell \iff s^k_i < u^\ell_j \]

\[ i^k \prec j^\ell \Rightarrow s^k_i < u^\ell_j \quad \text{and} \quad s^\ell_j > u^k_i \]
From Interval Order to Interleaving Traces

\[ i^k \prec j^\ell \iff s_i^k < u_j^\ell \]

\[ i^k \prec j^\ell \Rightarrow s_i^k < u_j^\ell \]
\[ i^k \parallel j^\ell \Rightarrow s_i^k > u_j^\ell \quad \text{and} \quad s_j^\ell > u_i^k \]
From Interval Order to Interleaving Traces

\[ i^k \prec j^\ell \quad \text{iff} \quad s^k_i < u^\ell_j \]

\[ i^k < j^\ell \quad \Rightarrow \quad s^k_i < u^\ell_j \]

\[ i^k \parallel j^\ell \quad \Rightarrow \quad s^k_i > u^\ell_j \quad \text{and} \quad s^\ell_j > u^k_i \]
From Interval Order to Interleaving Traces

\[
i^k \prec j^\ell \iff s^k_i < u^\ell_j
\]

\[
i^k \prec j^\ell \Rightarrow s^k_i < u^\ell_j \\
i^k \parallel j^\ell \Rightarrow s^k_i > u^\ell_j \quad \text{and} \quad s^\ell_j > u^k_i
\]

Remark:
Relative position of $S$s and $U$s are fixed.

Proposition: Interval Order induces equivalent interleaving traces.
A Geometrical Interpretation of Asynchronous Computability

$U_1 U_0 S_1 S_0 U_2 S_2$

Interleaving Trace/≈

Simplex

Dipath

Interval Order

C. Tasson
Pospace: $X_n = \prod_{i \in [n]} [0, r_i] \setminus \bigcup_{i, j \in [n]} U_i^k \cap S_j^l$

Dipath: $\alpha: [0, 1] \to X_n$ continuous and non decreasing

Dihomotopy: $h: \to [0, 1] \times [0, 1] \to X_n$ continuous non decreasing
Pospace: $X_n = \prod_{i \in [n]} [0, r_i] \setminus \bigcup_{i, j \in [n]} U_i^k \cap S_j^l$

Dipath: $\alpha : [0, 1] \to X_n$ continuous and non decreasing
Pospace: $\mathbb{X}_n = \prod_{i \in [n]} [0, r_i] \setminus \bigcup_{i,j \in [n]} U_i^k \cap S_j^l$

Dipath: $\alpha : [0, 1] \to \mathbb{X}_n$ continuous and non decreasing

Dihomotopy: $h : [0, 1] \times [0, 1] \to \mathbb{X}_n$ continuous non decreasing
Intersection with Update and Scan hyperplanes:

Interval Order: Characterized by relative position of $U$ and $S$,

$U_0 < S_0 < U_1$
From Dipath to Interval Order

Intersection with Update and Scan hyperplanes:

Interval Order: Characterized by relative position of $U$ and $S$,

$$U_1^0 < S_0^1 < U_1^1$$
A Geometrical Interpretation of Asynchronous Computability

\[ U_1 U_0 S_1 S_0 U_2 S_2 \]

**Interleaving Trace/≈**

**Simplex**

**Dipath**

**Interval Order**

C. Tasson
Consider a program with \( n + 1 \) processes and \( (r_i)_{i \in [n]} \) rounds.

**Complex of executions:**

- **Vertex:** (process, local memory)
- **Maximal Simplex:** \{\((0, \ell_0), \ldots, (n, \ell_n)\)\} where \( \ell_i \) is the local view by process \( i \) of the global execution.

**Examples:**

\[ U_1 \ U_0 \ S_1 \ S_0 \ U_2 \ S_2 \]

- \( S_0(110) \)
- \( S_1(010) \)
- \( S_2(111) \)
Consider a program with $n + 1$ processes and $(r_i)_{i \in [n]}$ rounds.

**Complex of executions:**

- **Vertex:** (process, local memory)
- **Maximal Simplex:** $\{(0, \ell_0), \ldots, (n, \ell_n)\}$ where $\ell_i$ is the local view by process $i$ of the global execution.

**Examples:**

\[
\begin{align*}
0, 0 \perp & \quad \xrightarrow{0 \rightarrow 1} \quad 1, 01 \\
0 & \quad \xrightarrow{0} \quad 1 \\
0, 01 & \quad \xrightarrow{1 \rightarrow 0} \quad 1, 1 \perp 1
\end{align*}
\]
Consider a program with $n + 1$ processes and $(r_i)_{i \in [n]}$ rounds.

**Complex of executions:**

- **Vertex:** (process, local memory)
- **Maximal Simplex:** $\{(0, \ell_0), \ldots, (n, \ell_n)\}$ where $\ell_i$ is the local view by process $i$ of the global execution.

**Examples:**

```
0, 0 ⊥ 1, 01 0, 01 1, ⊥ 1
0 1 0 1 0 1 0 1
```

Global

```
⊥ ⊥ U_0 0 ⊥ U_1 0 1 S_1 0 1 S_0 0 1
```

Local

```
0 1 0 1 0 1 0 01 01 01
```
Operational Semantics: The $i$th local memory contains all the updates preceding the last $i$th Scan.

Interval Order:

$$i^k \prec j^\ell \quad \text{iff} \quad S_i^j < U_k^\ell \quad \text{if } S_i^k > U_j^\ell \quad \text{iff} \quad i^k \parallel j^\ell \quad \text{or} \quad j^\ell \prec i^k \quad (1)$$

Asynchronous Complex: $n$ processes, $(r_i)_{i \in [n]}$ rounds

- **Vertex:** $(i^k, V_i^k)$ with $V_i^k$ interval order satisfying (1),

- **Maximal Simplex:** $\{(0^{r_0}, V_0^{r_0}), \ldots, (n^{r_n}, V_n^{r_n})\}$ if there is $X_n = \{j^\ell \mid j \in [n], \ell \in [r_i]\}$ an interval order its restriction to $i^k$ is

$$V_i^k = \{j^\ell \mid i^k \parallel j^\ell \text{ or } j^\ell \prec i^k \}$$
2 processes, 2 rounds: (no layer, no immediate snapshot)

1, ((0(01))(01))
0, ((0(01))(01))
1, ((0(01))(01))
0, ((0(01))(01))
1, ((0(01))(01))
0, ((0(01))(01))
1, ((0(01))(01))
0, ((0(01))(01))
1, ((0(01))(01))
0, ((0(01))(01))
1, ((0(01))(01))
0, ((0(01))(01))
1, ((0(01))(01))
0, ((0(01))(01))
1, ((0(01))(01))
0, ((0(01))(01))
1, ((0(01))(01))
0, ((0(01))(01))
1, ((0(01))(01))
0, ((0(01))(01))
2 processes, 2 rounds: (no layer, no immediate snapshot)
3 processes, 1 rounds: (no layer, no immediate snapshot)

Interval Order Complex Examples
**Theorem [Herlihy & al.]:** If the Protocol Complex is **contractible** then, the consensus is impossible.

**Proof sketch:**
Assume there is an algorithm $\delta$ solving the task, for any execution.

---

**Theorem [Kozlov]:**
Chromatic subdivision is collapsible, thus contractible.
An other proof of Collapsibility

Free Face: $\tau \subsetneq \sigma$ in $K$, with $\sigma$ the only such maximal simplex.

Collapse: Take off a free face $\tau \subset \sigma$ and in between simplexes. $K$ is collapsible if there is a collapse sequence to the point.

\[ \bullet \quad \bullet \quad \bullet \quad \bullet \]
An other proof of Collapsibility

**Free Face:** \( \tau \subsetneq \sigma \) in \( K \), with \( \sigma \) the **only** such maximal simplex.

**Collapse:** Take off a free face \( \tau \subset \sigma \) and in between simplexes.

\( K \) is **collapsible** if there is a collapse sequence to the point.

\[ \bullet - \bullet - \bullet - \bullet \quad \Rightarrow \quad \circ - \bullet - \bullet - \bullet - \circ \]
An other proof of Collapsibility

**Free Face:** $\tau \subsetneq \sigma$ in $K$, with $\sigma$ the **only** such maximal simplex.

**Collapse:** Take off a free face $\tau \subset \sigma$ and in between simplexes. $K$ is **collapsible** if there is a collapse sequence to the point.

\[
\begin{array}{c}
\bullet \quad \bullet \quad \bullet \quad \bullet \\
\Rightarrow \\
\circ \quad \cdots \quad \bullet \quad \bullet \quad \cdots \quad \circ 
\end{array}
\]
Free Face: $\tau \subsetneq \sigma$ in $K$, with $\sigma$ the only such maximal simplex.

Collapse: Take off a free face $\tau \subset \sigma$ and in between simplexes. $K$ is collapsible if there is a collapse sequence to the point.
An other proof of Collapsibility

**Free Face:** \( \tau \varsubsetneq \sigma \) in \( K \), with \( \sigma \) the **only** such maximal simplex.

**Collapse:** Take off a free face \( \tau \subset \sigma \) and in between simplexes. 

\( K \) is **collapsible** if there is a collapse sequence to the point.

\[
\begin{align*}
\bullet \quad \bullet \quad \bullet \quad \bullet & \quad \Rightarrow \quad \circ \quad \cdots \cdots \quad \bullet \quad \bullet \quad \cdots \cdots \quad \circ \\
\bullet \quad \bullet \quad \bullet \quad \bullet & \quad \Rightarrow \quad \bullet \quad \circ \quad \circ \quad \bullet
\end{align*}
\]
An other proof of Collapsibility

**Free Face:** $\tau \subsetneq \sigma$ in $K$, with $\sigma$ the **only** such maximal simplex.

**Collapse:** Take off a free face $\tau \subset \sigma$ and in between simplexes.

$K$ is **collapsible** if there is a collapse sequence to the point.

\[ \bullet \longrightarrow \bullet \longrightarrow \bullet \longrightarrow \bullet \quad \Rightarrow \quad \circ \ldots \ldots \bullet \longrightarrow \bullet \longrightarrow \circ \]

\[ \bullet \longrightarrow \bullet \longrightarrow \bullet \longrightarrow \bullet \quad \Rightarrow \quad \bullet \longrightarrow \circ \quad \circ \longrightarrow \circ \bullet \]
An other proof of Collapsibility

**Free Face:** \( \tau \subsetneq \sigma \) in \( K \), with \( \sigma \) the only such maximal simplex.

**Collapse:** Take off a free face \( \tau \subset \sigma \) and in between simplexes.

\( K \) is **collapsible** if there is a collapse sequence to the point.
An other proof of Collapsibility

**Free Face:** \( \tau \not\subset \sigma \) in \( K \), with \( \sigma \) the **only** such maximal simplex.

**Collapse:** Take off a free face \( \tau \subset \sigma \) and in between simplexes.

\( K \) is **collapsible** if there is a collapse sequence to the point.

\[
\bullet \quad \bullet \quad \bullet \quad \bullet \quad \Rightarrow \quad \bullet \quad \bullet \quad \bullet \quad \bullet \quad \bullet
\]

\[
\bullet \quad \bullet \quad \bullet \quad \bullet \quad \Rightarrow \quad \bullet \quad \bullet \quad \bullet \quad \bullet
\]

\[
\bullet \quad \bullet \quad \bullet \quad \bullet \quad \Rightarrow \quad \bullet \quad \bullet \quad \bullet \quad \bullet \quad \bullet
\]

\[
\bullet \quad \bullet \quad \bullet \quad \bullet \quad \Rightarrow \quad \bullet \quad \bullet \quad \bullet \quad \bullet \quad \bullet
\]
An other proof of Collapsibility

**Free Face:** $\tau \subsetneq \sigma$ in $K$, with $\sigma$ the **only** such maximal simplex.

**Collapse:** Take off a free face $\tau \subset \sigma$ and in between simplexes. $K$ is **collapsible** if there is a collapse sequence to the point.

![Diagram of collapsible complex]

$\Rightarrow$

![Diagram of simplified complex]
An other proof of Collapsibility

**Free Face:** $\tau \subsetneq \sigma$ in $K$, with $\sigma$ the **only** such maximal simplex.

**Collapse:** Take off a free face $\tau \subset \sigma$ and in between simplexes. $K$ is **collapsible** if there is a collapse sequence to the point.
An other proof of Collapsibility

**Free Face:** \( \tau \subsetneq \sigma \) in \( K \), with \( \sigma \) the only such maximal simplex.

**Collapse:** Take off a free face \( \tau \subset \sigma \) and in between simplexes.

\( K \) is **collapsible** if there is a collapse sequence to the point.
An other proof of Collapsibility

**Free Face:** $\tau \subsetneq \sigma$ in $K$, with $\sigma$ the **only** such maximal simplex.

**Collapse:** Take off a free face $\tau \subset \sigma$ and in between simplexes.

$K$ is **collapsible** if there is a collapse sequence to the point.

\[
\bullet \quad \bullet \quad \bullet \quad \bullet \quad \Rightarrow \quad \circ \quad \cdots \quad \bullet \quad \circ \quad \cdots \quad \circ
\]

\[
\bullet \quad \bullet \quad \bullet \quad \bullet \quad \Rightarrow \quad \bullet \quad \cdots \quad \circ \quad \circ \quad \cdots \quad \bullet
\]
An other proof of Collapsibility

**Free Face:** $\tau \subsetneq \sigma$ in $K$, with $\sigma$ the only such maximal simplex.

**Collapse:** Take off a free face $\tau \subset \sigma$ and in between simplexes. $K$ is collapsible if there is a collapse sequence to the point.
An other proof of Collapsibility

**Free Face:** $\tau \subsetneq \sigma$ in $K$, with $\sigma$ the only such maximal simplex.

**Collapse:** Take off a free face $\tau \subset \sigma$ and in between simplexes. $K$ is **collapsible** if there is a collapse sequence to the point.
Collapses and (Colored) Joins

Join:

\[ \star \]

Collapses:

\[ \chi(\Delta^I) \star \Delta^J \Rightarrow \partial \chi(\Delta^I) \star \Delta^J \]

\[ \chi(\Delta^I) \star \Delta^J \Rightarrow \chi(\Delta^I) \star \partial \Delta^J \]
Collapsibility of Iterated Protocol Complex

0, 0

02, 2
02, 0
012, 1
012, 0
012, 2
01, 0
01, 1
12, 1
12, 2
2, 2
1, 1
Collapsibility of Iterated Protocol Complex
Collapsibility of Iterated Protocol Complex

\[
\begin{align*}
012, 2 & \quad \Rightarrow \quad 012, 1 \\
& \quad \downarrow \\
\quad & \quad \downarrow \\
& \quad 012, 0
\end{align*}
\]
Collapsibility of Iterated Protocol Complex

\[ \Rightarrow \]

C. Tasson

Asynchronous  Interval Order  Dipath  Simplex  Perspective
Equivalent presentations of Asynchronous Computations:

\[ U_1 \ U_0 \ S_1 \ S_0 \ U_2 \ S_2 \]

Interleaving Trace \( \approx \)

\[ \begin{array}{c}
0 \\
2 \\
1 \\
\end{array} \]

Interval Order

Simplex

Dipath

Collapsing path of iterated protocol complex: a procedure

What’s next:

- Such equivalence for other models of communication
- Compare collapsing procedure with Kozlov procedure
- Translate collapsing path into pospace (link with Trace Space [Raussen]).