A Geometric Approach to the Problem of Unique Decomposition of Processes

Thibaut Balabonski & Emmanuel Haucourt

PPS
Université Paris Diderot

LIST
CEA

CONCUR’10: September 3
Concurrent programming

Background

• A set of concurrent processes
• Some shared resources

Tools for peaceful interaction

• Semaphores
• Synchronisation barriers
• Communications
• …
A toy language for semaphores

Two instructions

\( Pa \) Take one occurrence of \( a \).
\( Va \) Release \( a \).

Each semaphore \( a \) is characterized by its arity.
Nontrivial independence

\[
\begin{align*}
\pi_1 &= Pa.Pc.Vc.Va \\
\| \pi_2 &= Pa.Pc.Vc.Va \\
\| \pi_3 &= Pb.Pc.Vc.Vb \\
\| \pi_4 &= Pb.Pc.Vc.Vb
\end{align*}
\]

\(a, b\) mutual exclusion

\(c\) two occurrences
Nontrivial independence

\[ \pi_1 = Pa.Pc.Vc.Va \]
\[ \parallel \pi_2 = Pa.Pc.Vc.Va \]
\[ \parallel \pi_3 = Pb.Pc.Vc.Vb \]
\[ \parallel \pi_4 = Pb.Pc.Vc.Vb \]

\[ a, b \] mutual exclusion
\[ c \] two occurrences
Nontrivial independence

\[ \begin{align*}
\pi_1 &= Pa. \quad Va \\
\| \pi_2 &= Pa. \quad Va \\
\| \pi_3 &= Pb. \quad Vb \\
\| \pi_4 &= Pb. \quad Vb
\end{align*} \]

\(a, b\) mutual exclusion

\(c\) two occurrences
Roadmap

1. A Geometric Semantics for Concurrence
2. Semantic Definition of Independence
3. A General Decomposition Theorem
4. Implementation
A Geometric Semantics for Concurrence

Interpreting programs by Euclidean spaces.

Semantic Definition of Independence
A General Decomposition Theorem
Implementation
One process, one dimension

\[ Pa \cdot Pb \cdot Vb \cdot Va \]
Two processes, two dimensions?

\[ Pa \cdot Pb \cdot Vb \cdot Va \parallel Pb \cdot Pa \cdot Va \cdot Vb \]

- Geometric Semantics
- Definition of Independence
- Decomposition Theorem
- Implementation

T. Balabonski & E. Haucourt

Geometric Decomposition of Processes — CONCUR’10: September 3 — 7/22
Two processes, two dimensions?

\[ Pa.Pb.Vb.Va \parallel Pb.Pa.Va.Vb \]
Euclidean spaces with holes

\[ Pa . Pb . Vb . Va \parallel Pb . Pa . Va . Vb \]

Diagram showing points \( Pa, Pb, Vb, Va \) in a Euclidean space with holes.
Euclidean spaces with holes

\[ Pa.Pb.Vb.Va \parallel Pb.Pa.Va.Vb \]
Euclidean spaces with holes

\[
P_a.P_b.V_b.V_a \parallel P_b.P_a.V_a.V_b
\]
Euclidean spaces with holes

\[ Pa.Pb.Vb.Va \parallel Pb.Pa.Va.Vb \]

- Geometric Semantics
- Definition of Independence
- Decomposition Theorem
- Implementation
A 3D example

\[ \pi_1 = Pa.Pc.Vc.Va \]
\[ \parallel \pi_2 = Pa.Pc.Vc.Va \]
\[ \parallel \pi^*_3 = Pc.Vc \]
Semantic Definition of Independence

Cartesian product and factorization are the key.

A General Decomposition Theorem

Implementation
Independence of programs

Geometric Semantics

Definition of Independence

Σ₁ independent from Σ₂:
Σ₂ has no effect on the execution of Σ₁.

Semantic definition

Σ₁ and Σ₂ are independent
iff

\[
\left[ \Sigma_1 \parallel \Sigma_2 \right] = \left[ \Sigma_1 \right] \times \left[ \Sigma_2 \right]
\]
Independent programs: an example

**Geometric Semantics**

\[
\begin{align*}
\text{Pa.Pc.Vc.Va} & \parallel \text{Pa.Pc.Vc.Va} \\
\text{Pa.Pc.Vc.Va} & \parallel \text{Pa.Pc.Vc.Va} \\
\text{Pa.Pc.Vc.Va} & \parallel \text{Pc.Vc} \\
\text{Pa.Pc.Vc.Va} & \parallel \text{Pa.Pc.Vc.Va} \\
\text{Pc.Vc} & \\
\end{align*}
\]

- Definition of Independence
- Decomposition Theorem
- Implementation
Algebraic notions

Irreducibles
A decomposition always exists.

Primes
Any decomposition is unique.

Prime element $x$: if $x|y_1\ldots y_n$ then there is $i$ such that $x|y_i$
A General Decomposition Theorem

Homogeneous tuples of coordinates, up to permutation.

Implementation
Homogeneous sets of words

\[ N \text{-dimensional geometric objects} \]

- Point: \( N \)-uple of coordinates.
- Box: \( N \)-uple of intervals.
- **Gist**: word of length \( N \). (on a given alphabet)
Homogeneous sets of words

| $N$-dimensional geometric objects |
|----------------------------------|
| • Point: $N$-uple of coordinates. |
| • Box: $N$-uple of intervals.    |
| • Gist: word of length $N$. (on a given alphabet) |

Sets whose elements have the same length.
### Product of sets

#### Cartesian product

**Definition of Independence**

**Implementation**

**Decomposition Theorem**

- **Geometric Semantics**

**Concatenation** of all pairs.

\[
\begin{align*}
  \mathcal{S}_1 \times \mathcal{S}_2 & = \mathcal{S}_1 + \mathcal{S}_2 \\
  \dim(\mathcal{S}_1 \times \mathcal{S}_2) & = \dim(\mathcal{S}_1) + \dim(\mathcal{S}_2)
\end{align*}
\]

With homogeneous sets:
Parallel composition should be commutative, Cartesian product is not.

Permutation of coordinates (group action)

Homogeneous monoid: quotient by permutation.
## Equivalence Theorem

On the homogeneous monoid,

\[ \text{Irreducibility} = \text{Primality} \]

## Factorization Corollary

Each element of the homogeneous monoid has a unique finest decomposition.
Implementation

*The constructive side of the factorization result.*
A Galois connection

\[ \gamma(\mathcal{F}) := \bigcup \mathcal{F} \]

Families of $n$-cubes \quad Subsets of $\mathbb{R}^n$

\[ \alpha(A) := \{ \text{maximal cubes of } A \} \]
Subsets of $\mathbb{R}^n$ with finitely many maximal cubes.
The **ALCOOL** analyzer: features

- Based on the structure of **cubical areas**.
- Recognizes an **extension** of the PV language.
- **Detects** deadlocks, overflows, unreachable states...
- **Factorizes** concurrent programs.
- Modelling of loops in progress.
Algorithm: An Example

\[ \begin{align*}
\pi_1 & = Pa.Pc.Vc.Va \\
\pi_2 & = Pb.Pc.Vc.Vb \\
\pi_3 & = Pa.Pc.Vc.Va \\
\pi_4 & = Pb.Pc.Vc.Vb
\end{align*} \]
Algorithm: The State Space

| Geometric Semantics | Definition of Independence | Decomposition Theorem | Implementation |
|---------------------|---------------------------|----------------------|----------------|
|                     |                           |                      |                |
|                      |                           |                      |                |
| $[0, 1]$ $\times$ $[0, 1]$ $\times$ $[0, \infty]$ $\times$ $[0, \infty]$ | $[0, 1]$ $\times$ $[4, \infty]$ $\times$ $[0, \infty]$ $\times$ $[0, \infty]$ | $[0, 1]$ $\times$ $[0, \infty]$ $\times$ $[4, \infty]$ $\times$ $[0, \infty]$ |
| $[0, 1]$ $\times$ $[0, \infty]$ $\times$ $[0, \infty]$ $\times$ $[0, \infty]$ | $[0, \infty]$ $\times$ $[0, \infty]$ $\times$ $[0, \infty]$ $\times$ $[0, \infty]$ | $[0, \infty]$ $\times$ $[0, \infty]$ $\times$ $[0, \infty]$ $\times$ $[4, \infty]$ |
| $[0, \infty]$ $\times$ $[0, 1]$ $\times$ $[0, 1]$ $\times$ $[4, \infty]$ | $[0, \infty]$ $\times$ $[4, \infty]$ $\times$ $[0, 1]$ $\times$ $[0, \infty]$ | $[0, \infty]$ $\times$ $[4, \infty]$ $\times$ $[0, \infty]$ $\times$ $[0, \infty]$ |
| $[0, \infty]$ $\times$ $[0, 1]$ $\times$ $[4, \infty]$ $\times$ $[0, \infty]$ | $[0, \infty]$ $\times$ $[4, \infty]$ $\times$ $[4, \infty]$ $\times$ $[0, \infty]$ | $[0, \infty]$ $\times$ $[4, \infty]$ $\times$ $[0, \infty]$ $\times$ $[4, \infty]$ |
| $[0, \infty]$ $\times$ $[0, \infty]$ $\times$ $[4, \infty]$ $\times$ $[0, \infty]$ | $[0, \infty]$ $\times$ $[4, \infty]$ $\times$ $[4, \infty]$ $\times$ $[0, \infty]$ | $[0, \infty]$ $\times$ $[4, \infty]$ $\times$ $[0, \infty]$ $\times$ $[4, \infty]$ |
Algorithm: testing \{2, 3\}

\[
\begin{align*}
[0, 1] \
[0, 1] \
[4, \infty] \
[4, \infty] & \times [0, \infty] \times [0, \infty] \times [0, 1] \
[0, \infty] \
[0, \infty] \
[4, \infty] \
[4, \infty] \\
& \times [0, \infty] \times [0, 1] \times [0, \infty] \\
[0, \infty] \
[0, \infty] \
[4, \infty] \
[4, \infty] & \times [0, 1] \times [0, \infty] \times [0, \infty] \\
[0, 1] \
[4, \infty] \
[0, 1] \
[4, \infty] & \times [0, 1] \times [4, \infty] \times [0, \infty] \\
[4, \infty] \
[4, \infty] \
[4, \infty] \
[4, \infty] & \times [4, \infty] \times [0, \infty] \times [0, \infty] \\
[0, \infty] \
[0, \infty] \
[0, 1] \
[4, \infty] & \times [0, 1] \times [4, \infty] \times [4, \infty] \\
[0, \infty] \
[0, \infty] \
[4, \infty] \
[4, \infty] & \times [4, \infty] \times [4, \infty] \times [0, \infty]
\end{align*}
\]
Algorithm: testing \( \{2, 4\} \)

\[
\begin{array}{c}
[0, 1] \\
[4, \infty] \\
[0, \infty] \\
[0, \infty] \\
\times \quad [0, 1] \\
\times \quad [0, \infty] \\
\times \quad [0, 1] \\
\times \quad [4, \infty] \\
\end{array}
\]

\[
\begin{array}{c}
[0, 1] \\
[4, \infty] \\
[0, \infty] \\
[0, \infty] \\
\times \quad [4, \infty] \\
\times \quad [0, \infty] \\
\times \quad [0, 1] \\
\times \quad [4, \infty] \\
\end{array}
\]

\[
\begin{array}{c}
[0, 1] \\
[4, \infty] \\
[0, \infty] \\
[0, \infty] \\
\times \quad [0, \infty] \\
\times \quad [0, \infty] \\
\times \quad [0, 1] \\
\times \quad [4, \infty] \\
\end{array}
\]

\[
\begin{array}{c}
[0, 1] \\
[4, \infty] \\
[0, \infty] \\
[0, \infty] \\
\times \quad [0, \infty] \\
\times \quad [0, \infty] \\
\times \quad [4, \infty] \\
\end{array}
\]

T. Balabonski & E. Haucourt
Some tests

| Example               | Time | Decompos. |
|-----------------------|------|-----------|
| 6 philosophers        | 0.2  | No        |
| 7 philosophers        | 0.7  | No        |
| 8 philosophers        | 3.5  | No        |
| 9 philosophers        | 21   | No        |
| 10 philosophers       | 152  | No        |

Time is given in seconds.

| Example | Time | Decomposition | Example | Time | Decompos. |
|---------|------|---------------|---------|------|-----------|
| $\Sigma_{2,2}$ | 0.1  | $\{1, 3\} \{2, 4\}$ | $\Sigma'_{2,2}$ | 0.1  | No        |
| $\Sigma_{2,2,2}$ | 0.1  | $\{1, 4\} \{2, 5\} \{3, 6\}$ | $\Sigma'_{2,2,2}$ | 0.3  | No        |
| $\Sigma_{3,3}$ | 0.13 | $\{1, 3, 5\} \{2, 4, 6\}$ | $\Sigma'_{3,3}$ | 0.52 | No        |
| $\Sigma_{2,2,2,2}$ | 0.13 | $\{1, 5\} \{2, 6\} \{3, 7\} \{4, 8\}$ | $\Sigma'_{2,2,2,2}$ | 7.1  | No        |
| $\Sigma_{4,4}$ | 1    | $\{1, 3, 5, 7\} \{2, 4, 6, 8\}$ | $\Sigma'_{4,4}$ | 33   | No        |
| $\Sigma_{3,3,3}$ | 1.5  | $\{1, 4, 7\} \{2, 5, 8\} \{3, 6, 9\}$ | $\Sigma'_{3,3,3}$ | 293  | No        |
| $\Sigma_{4,5}$ | 6.1  | $\{1, 3, 5, 7\} \{2, 4, 6, 8, 9\}$ | $\Sigma'_{4,5}$ | 327  | No        |
| $\Sigma_{5,5}$ | 50   | $\{1, 3, 5, 7, 9\} \{2, 4, 6, 8, 10\}$ | $\Sigma'_{5,5}$ | 2875 | No        |
Conclusion

- Geometry is cool.
- Cubical areas can model several mainstream concurrent primitives.
- “Torical” areas give support for loops.
- Relation to algebraic topology and directed homotopy.