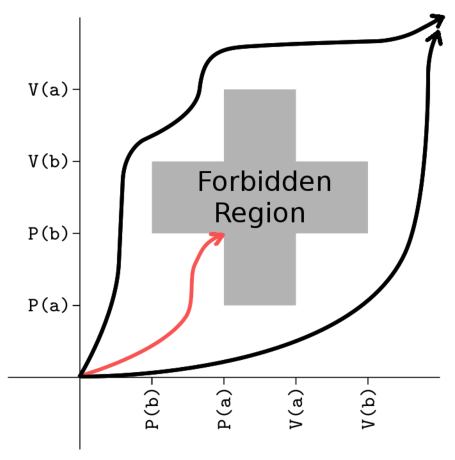
A bridge between two geometric models for concurrency

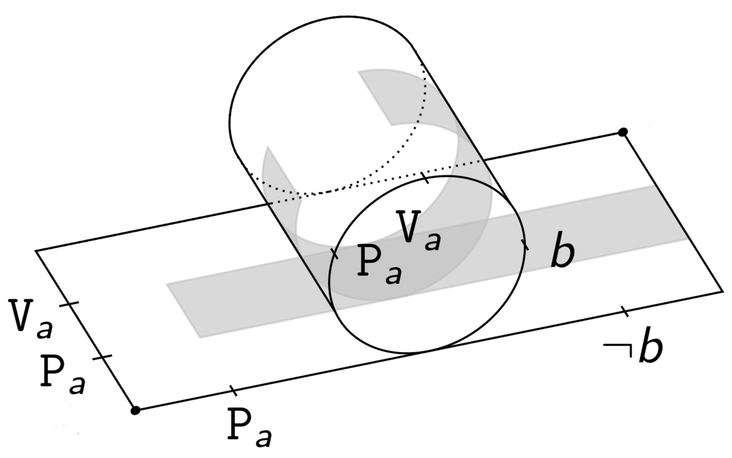
Jérémy Ledent, joint work with Éric Goubault and Samuel Mimram

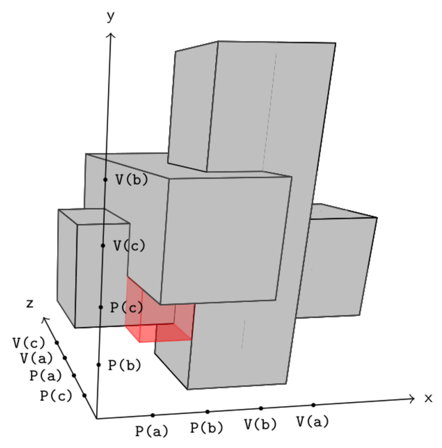
LIX, École Polytechnique

Directed algebraic topology

The directed algebraic topological approach to concurrency models programs as *directed spaces*. In this context, programs are seen as topological spaces, where each dimension corresponds to a process that can make progress independently from the others. The notion of direction models time or causality. The possible executions of the program correspond to paths in that space. This view is particularly useful for program verification, where considering these paths up to homotopy drastically reduces the number of executions that have to be taken into account.



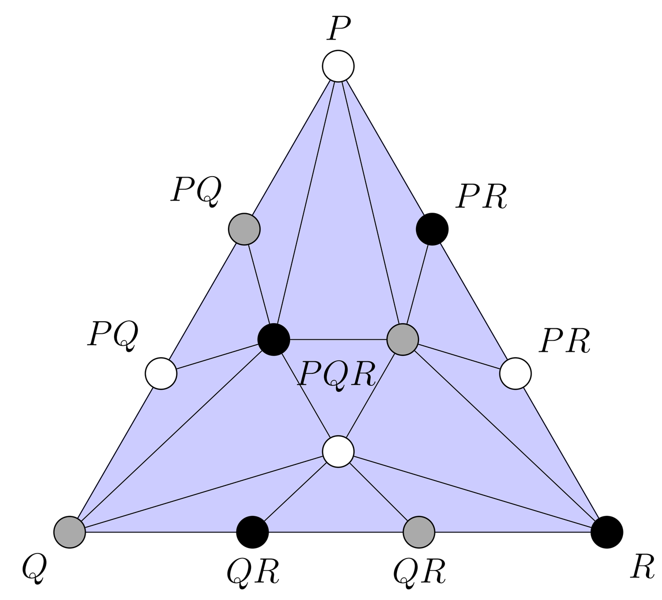


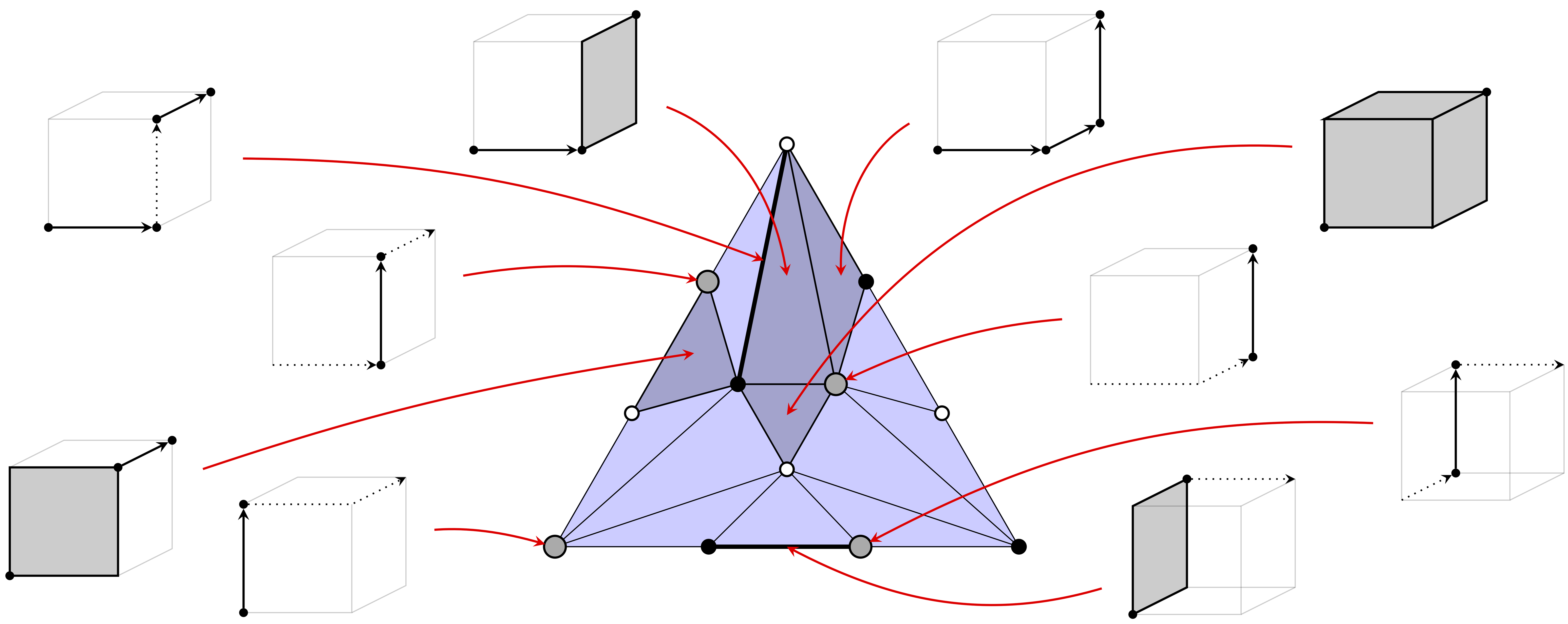


The discrete version of these spaces is called a *pre-cubical complex*, a kind of generalized graph with not only vertices and edges but also higher-dimensional cubes. The complex $\square^n$ consists of one cube of dimension n, and all its lower-dimensional faces. It represents a situation where each process is about to take one step, and every interleaving can happen.

Fault-tolerant computability

The area of distributed computability studies which *tasks* can be solved in various computational models where processes may crash. A typical example is the consensus task, where the processes have to agree on one of their input values. Geometric methods have been used to obtain impossibility results, by reducing the solvability of a task to the existence of some simplicial map between simplicial complexes. This approach is based on *chromatic simplicial complexes*, where the colors of the vertices correspond to processes; and the maximal simplices (triangles) are the possible executions. Two simplices share a vertex when the corresponding process cannot distinguish between the two executions. The figure on the right shows the *standard chromatic subdivision* that arises when studying the *immediate snapshot* protocol. Each process owns a dedicated memory cell in a shared array. First, it writes its input value in its cell, then it atomically reads the whole array. For example, the white vertex labeled PQ represents process P which has seen the values of both process Q and itself, but did not see process R.





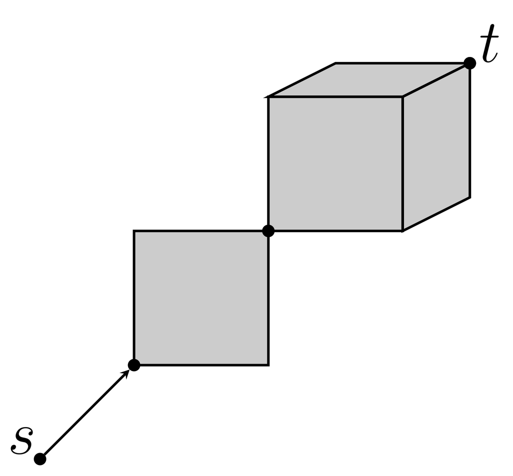
Theorem

The poset of partial cube paths in the combinatorial $(n+1)$-cube $\square^{n+1}$ is order-isomorphic to the face poset of the n-dimensional standard chromatic subdivision $\chi(\Delta^n)$:

$$(p\mathcal{CP}(\square^{n+1}), \preccurlyeq) \;\simeq\; (\chi(\Delta^n), \subseteq)$$

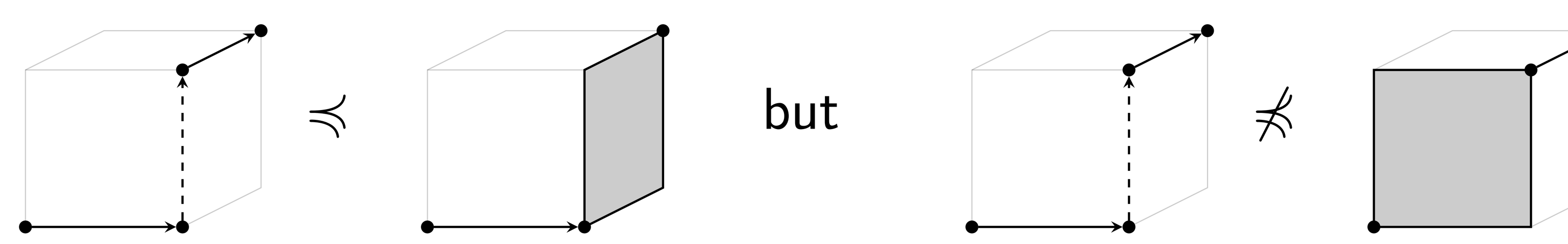
Cube paths

In a pre-cubical complex (such as $\square^{n+1}$), we can define a notion of cube path. Remember that our cubes are directed: a cube c has a source vertex $\partial^-(c)$ (drawn at the bottom-left on the pictures) and a target vertex $\partial^+(c)$ (top-right on the pictures). A *cube path* from s to t is a sequence of cubes $(c_0, \ldots, c_k)$ such that $\partial^-(c_0) = s$, $\partial^+(c_k) = t$ and $\partial^+(c_i) = \partial^-(c_{i+1})$. In the Theorem, (total) cube paths correspond to simplices of maximal dimension, i.e., triangles on the picture.



Partial cube paths

To get the full structure of $\chi(\Delta^n)$, we need to interpret lower-dimensional simplices. They correspond to *partial cube paths*, whose set is written $p\mathcal{CP}$, which are cube paths with holes: we only require that there exists a path from $\partial^+(c_i)$ to $\partial^-(c_{i+1})$. A partial path p can be *extended* to p', written $p \preccurlyeq p'$, by either filling the holes with new cubes, or replacing an existing cube with a greater-dimensional one of which it is a forward face.



Jérémy Ledent – École Polytechnique
jeremy.ledent@lix.polytechnique.fr