On Symmetry versus Asynchronism:

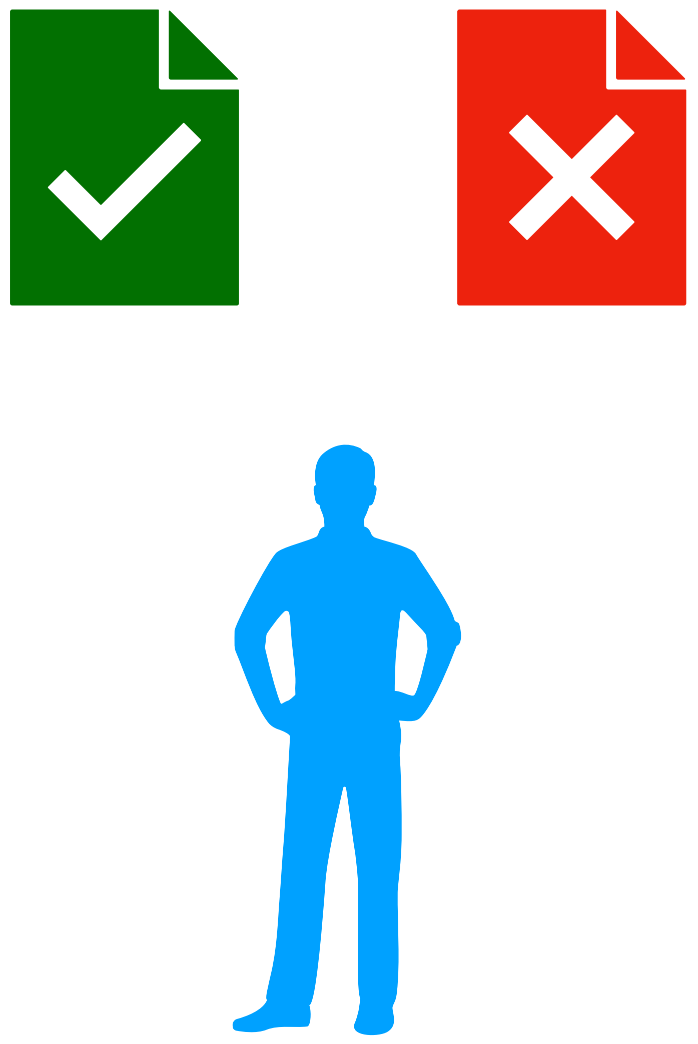
At the Edge of Universality in Automata Networks

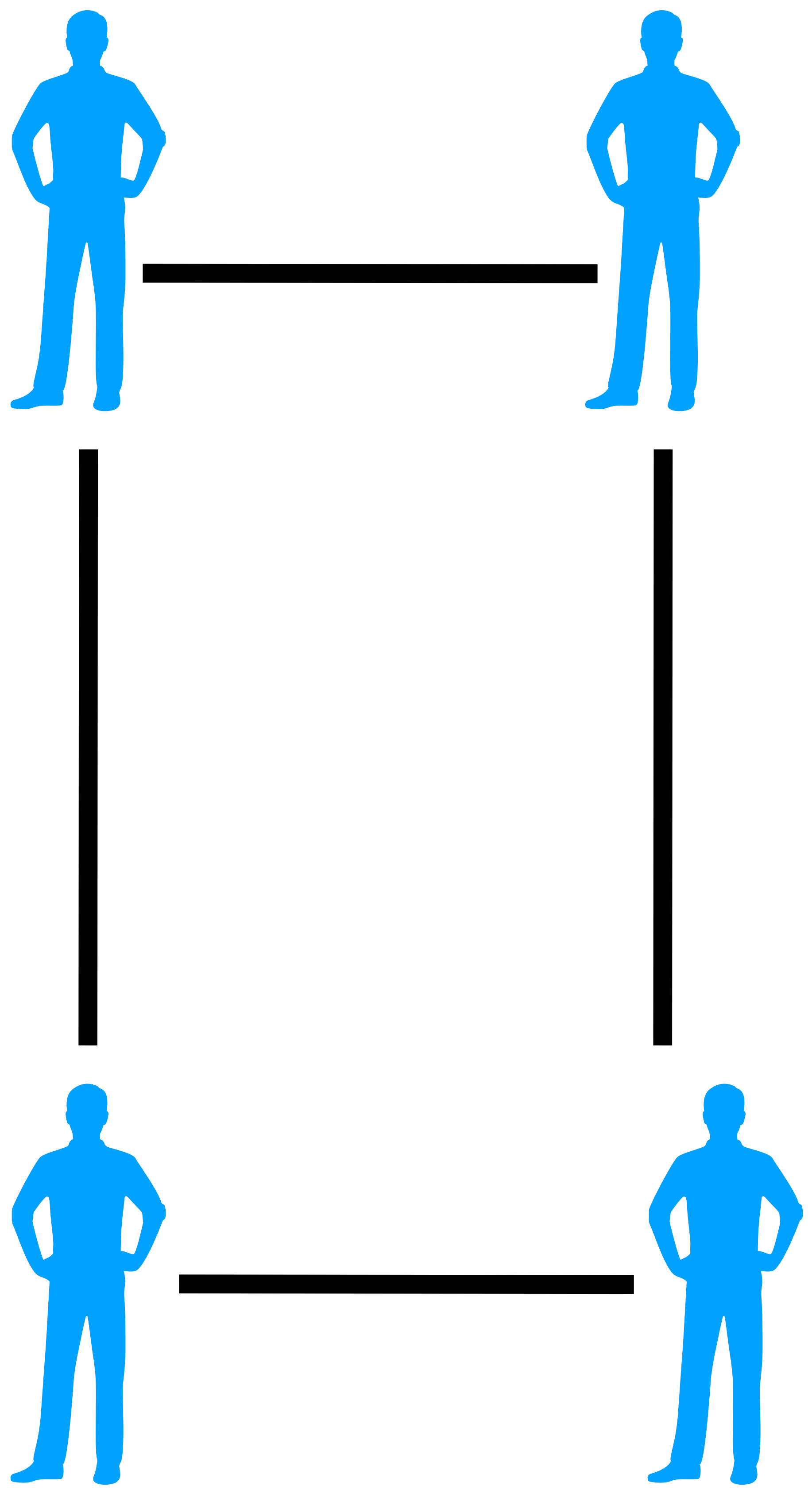
Martín Ríos Wilson and Guillaume Theyssier.

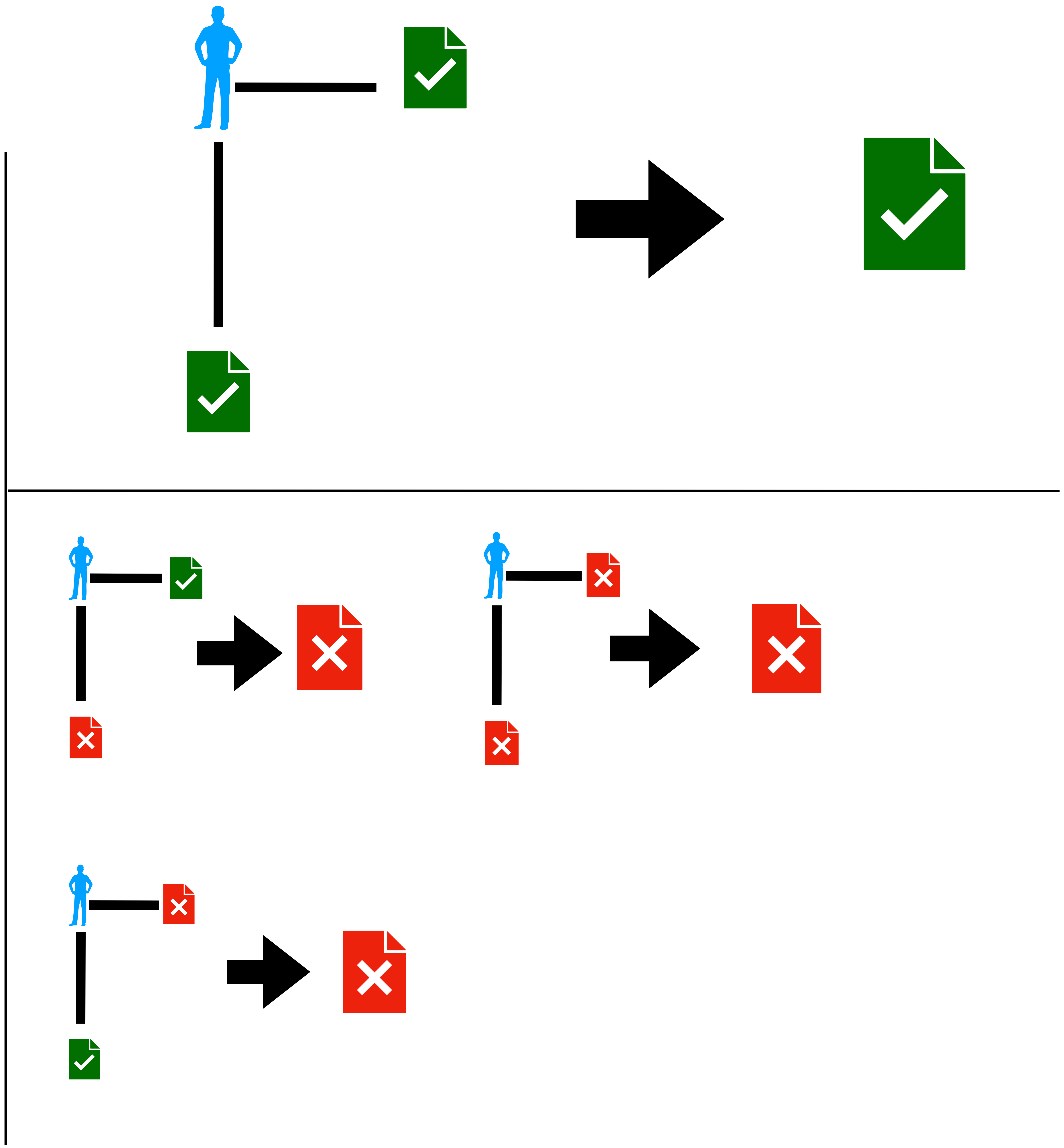
Facultad de Ingeniería y Ciencias, Universidad Adolfo Ibáñez.

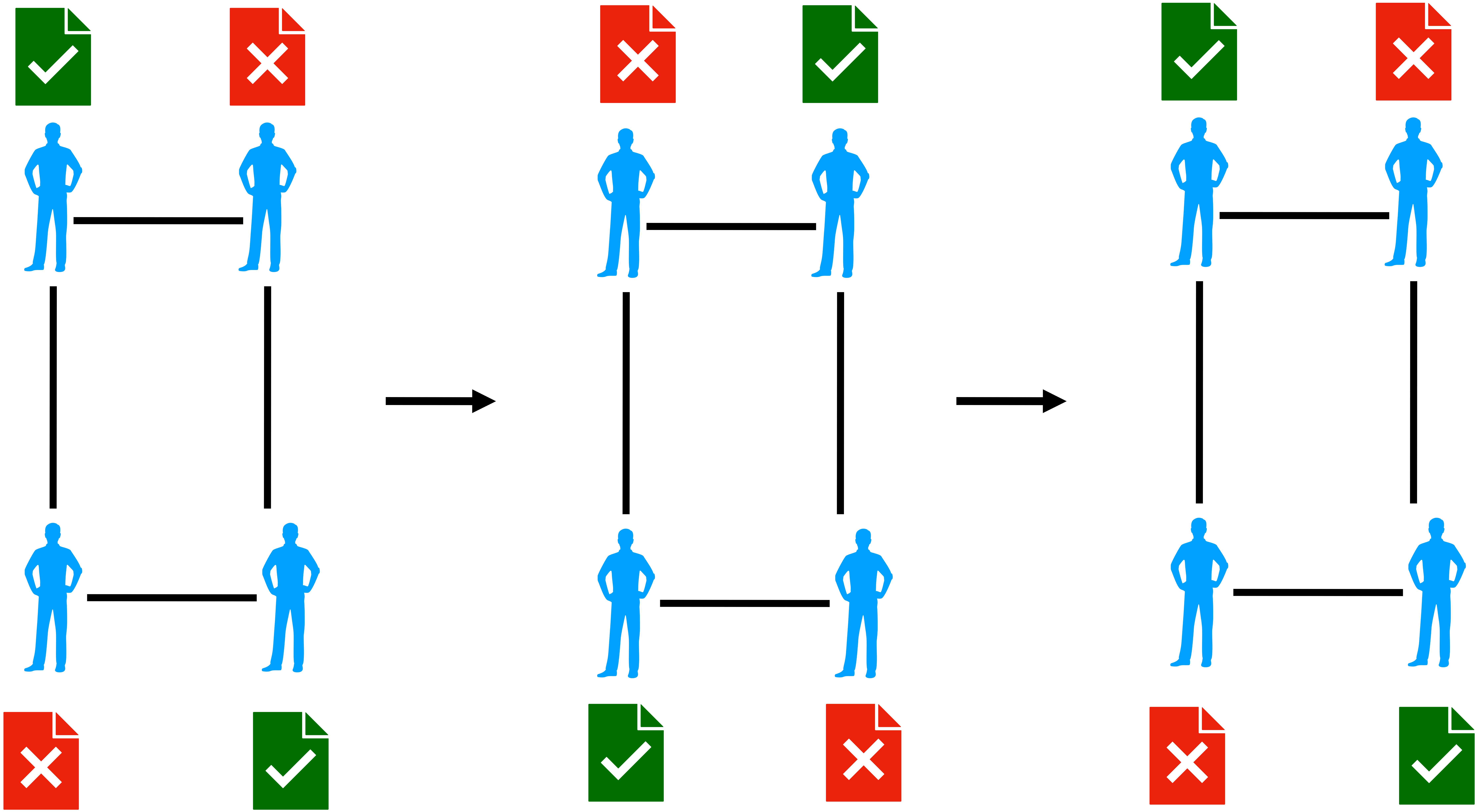
The model

Automata networks





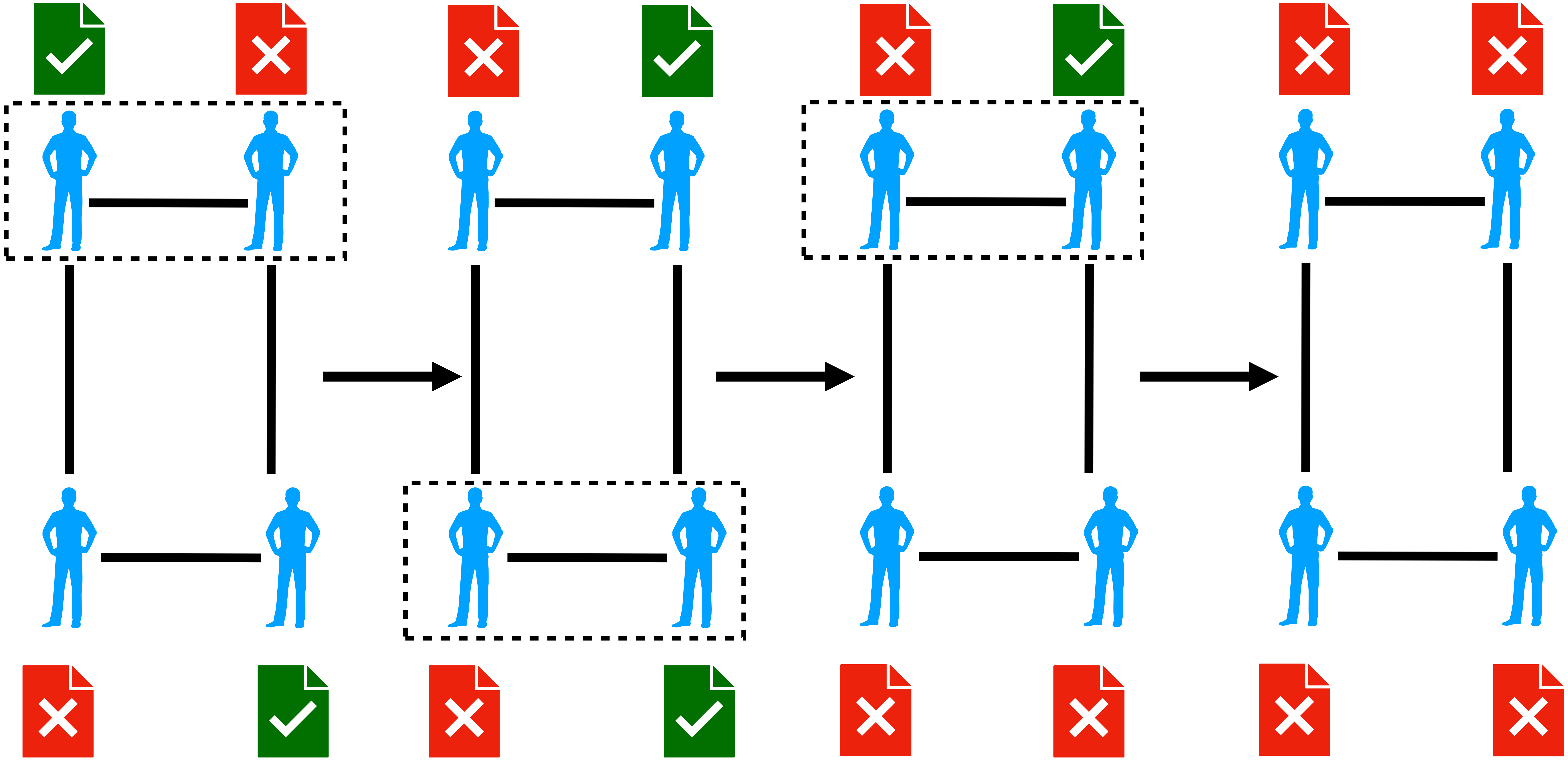




Automata networks

- Model : $G = (V, E)$ and a function $F: Q^V \mapsto Q^V$.
- (Q^V, F) is a dynamical system. Each configuration defines an **orbit.**
- Deterministic and finite: each configuration reaches an attractor (fixed point/limit cycle) in finite time.
- The dynamical behavior can change according to different factors:
  1. Update scheme.
  2. Graph structure.
  3. Function properties.

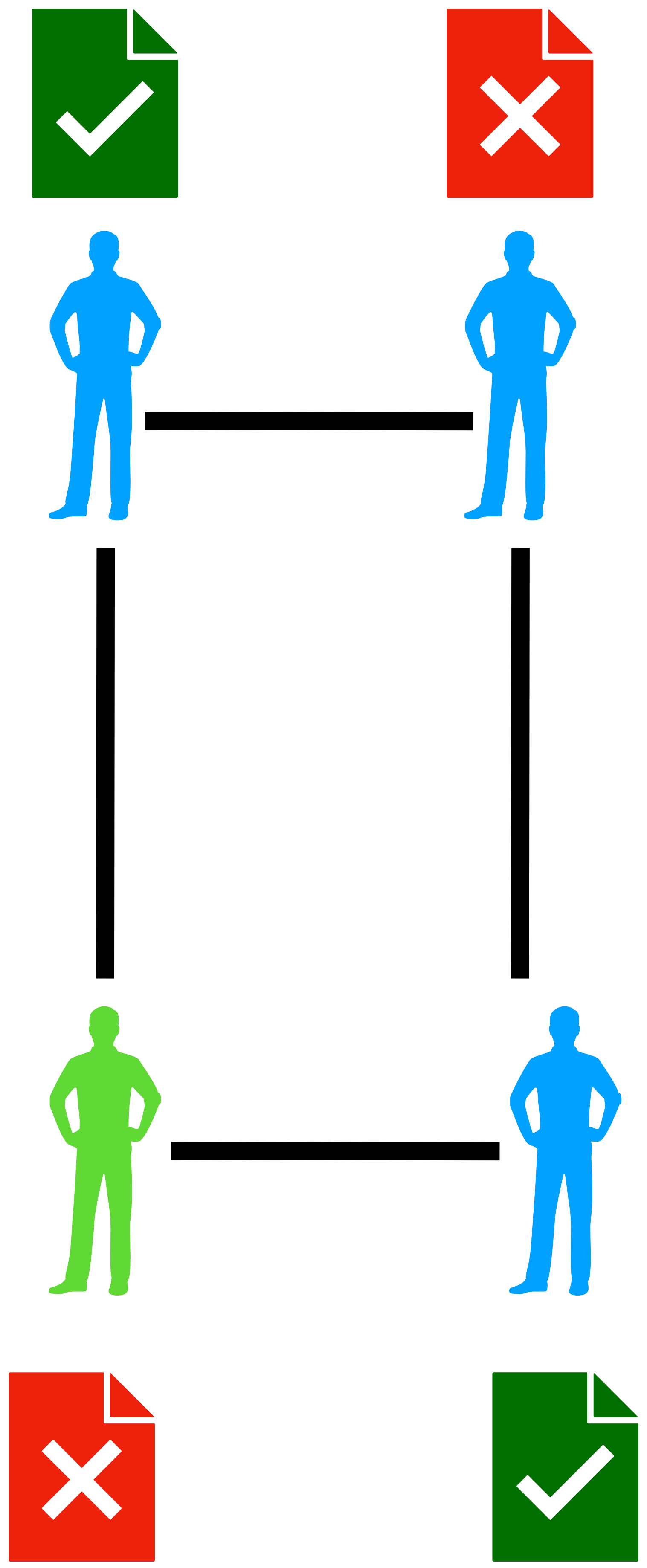
Update schemes

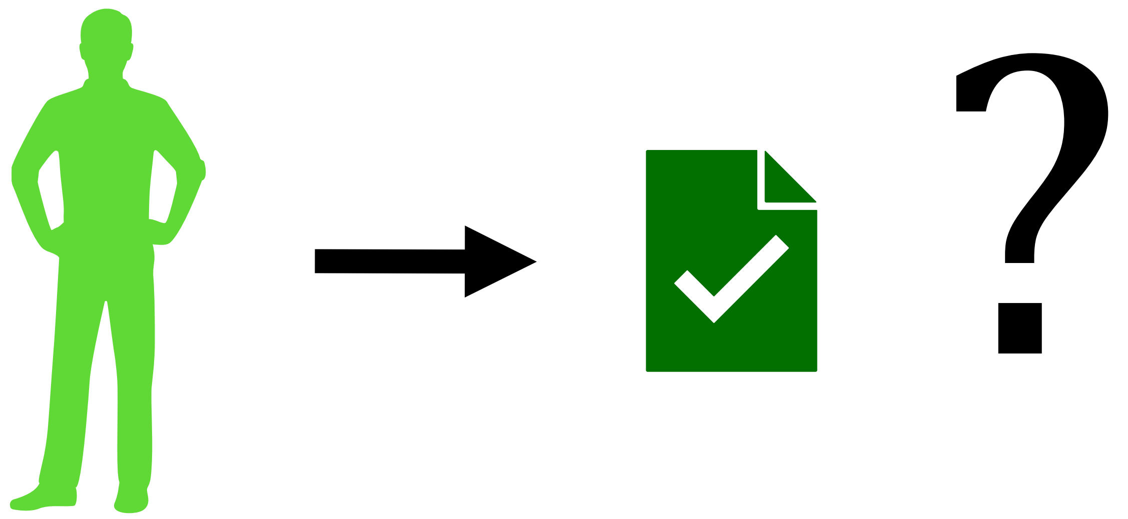


Main questions and general approach

A computational complexity approach

Prediction problem





- Predict the state of an objective node after t time steps.
- $t \in \mathbb{N}$ given in unary or binary.
- No time in the input (Prediction-Change).

- Direct simulation approach:

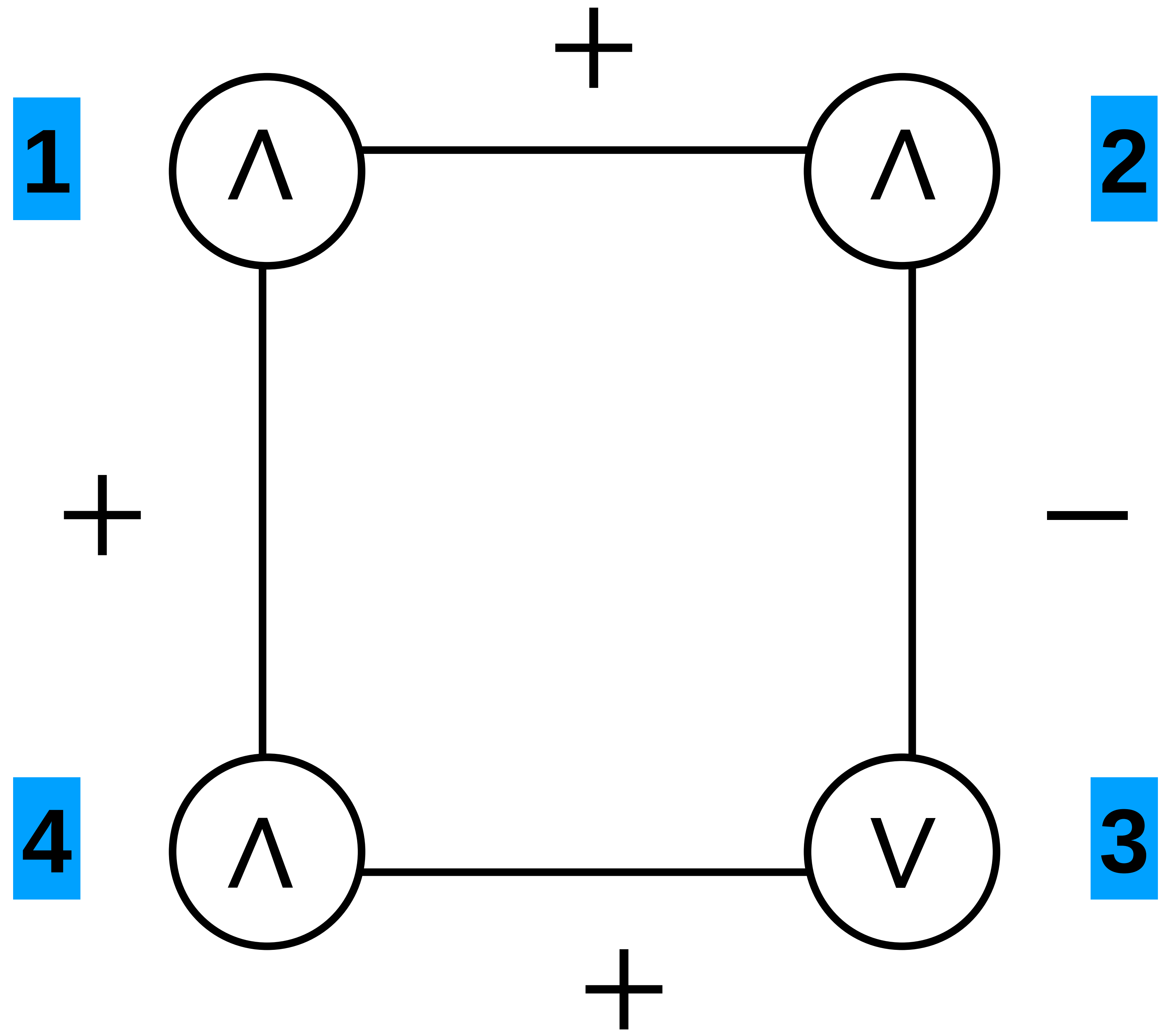
- Solvable in polynomial space.

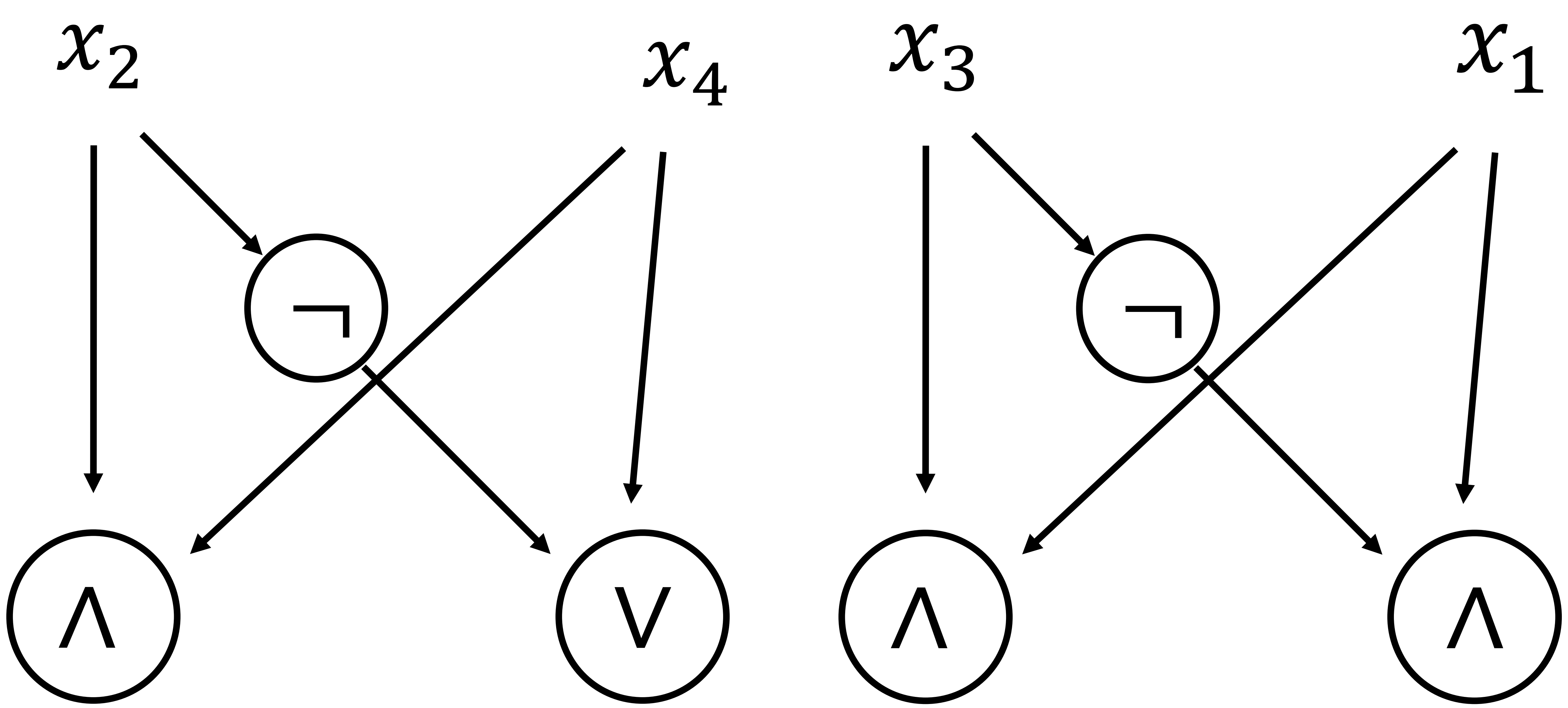
- Polynomial time for Unary PRED.

Representations

Concrete symmetric automata networks (CSAN).

$$F(x) = (x_2 \wedge x_4, x_1 \wedge \overline{x}_3, x_4 \vee \overline{x}_2, x_1 \wedge x_3)$$



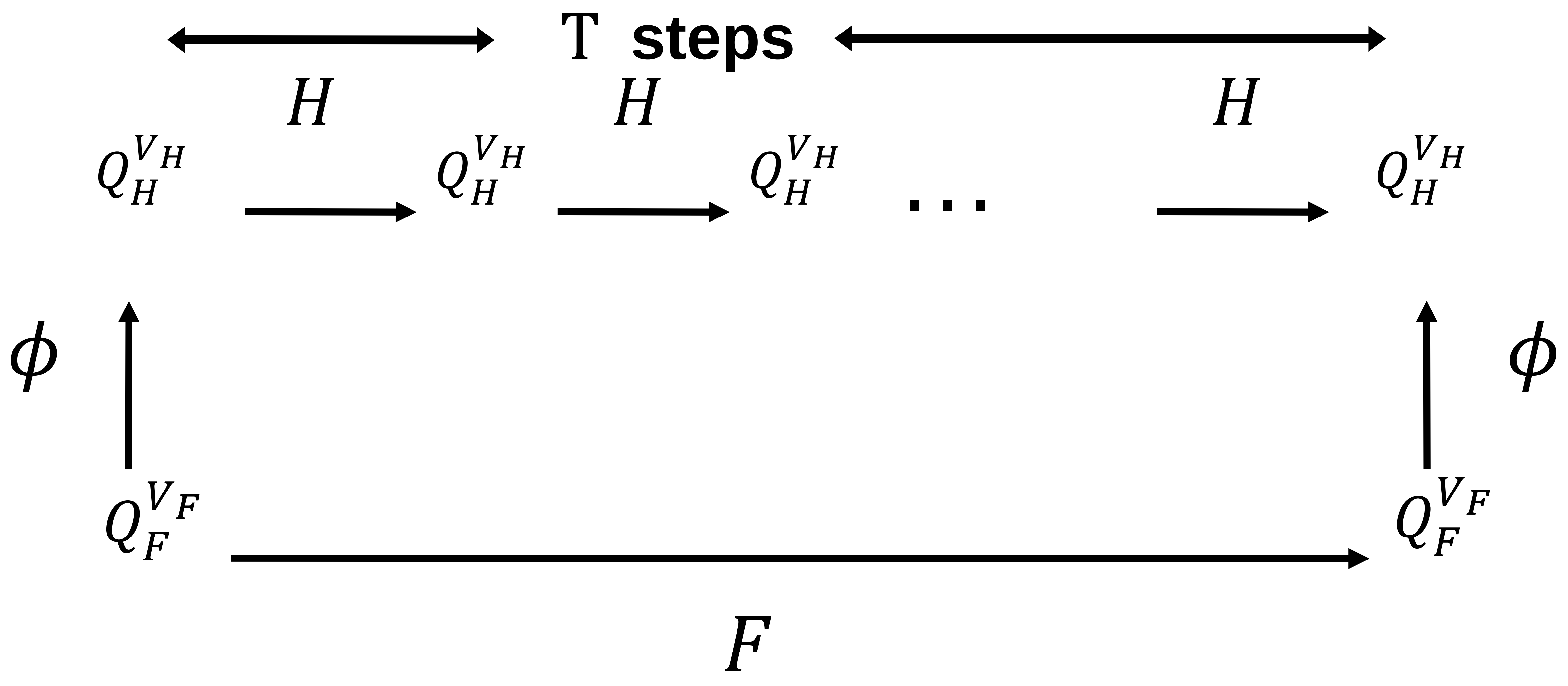


Labeled graph Boolean circuit

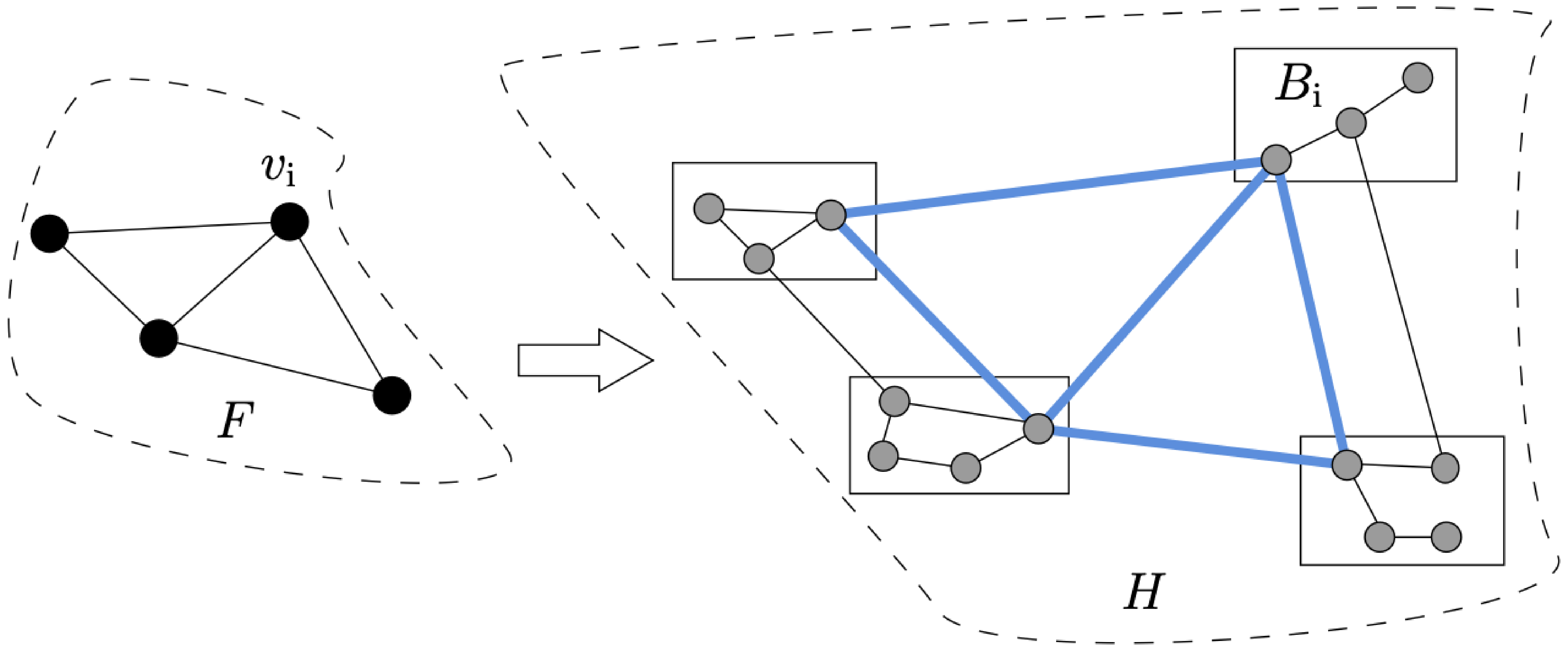
Labeled graph ? Boolean circuit

Simulation between automata networks

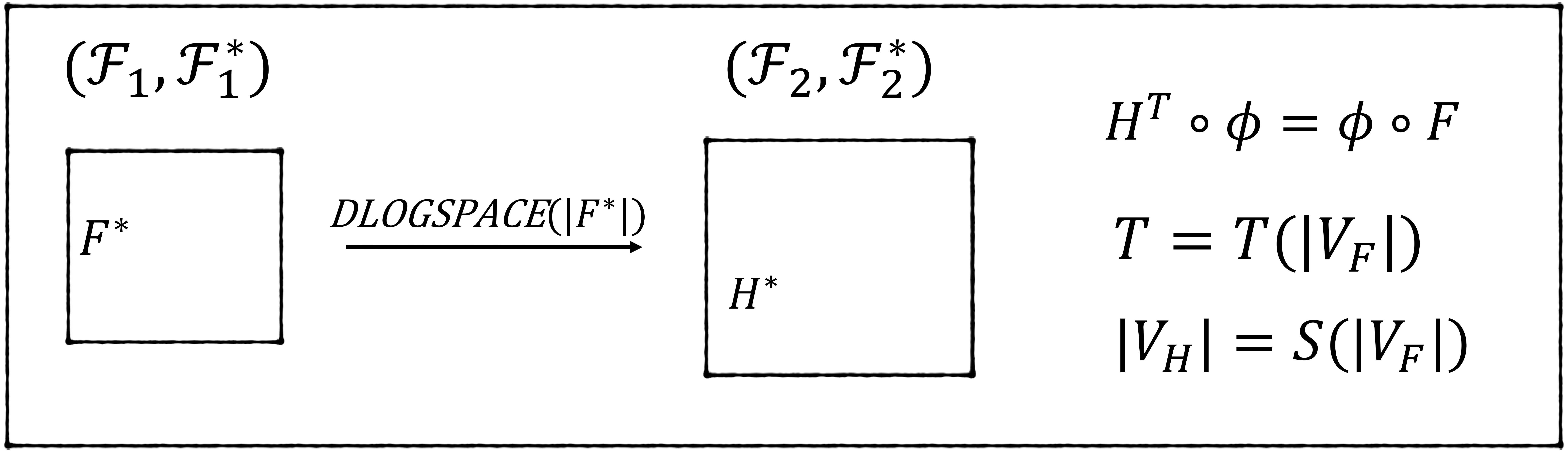
- Each node is represented by a block.
- States of the simulated network F are coded by an **injective** map ϕ.
- Each time step of F is simulated in T time steps by H



$$H^T \circ \phi = \phi \circ F$$



Simulation between families of automata networks



From now on, we consider that T, S are **polynomial functions.**

Simulation and universality.

- Simulation allow us to transfer computational properties from one family to another:

Lemma. $\mathcal{F}_1 \preccurlyeq_S^T \mathcal{F}_2 \Longrightarrow PRED_{\mathcal{F}_1} \leq_L PRED_{\mathcal{F}_2}$

- A family $\mathcal{F}$ is **strongly universal** if it can simulate any bounded degree automata network in linear space and constant time.

(**Bounded degree automata networks** are AN that can be represented by bounded degree graphs).

Consequences of universality

Theorem. A **strongly universal** family satisfies the following properties:

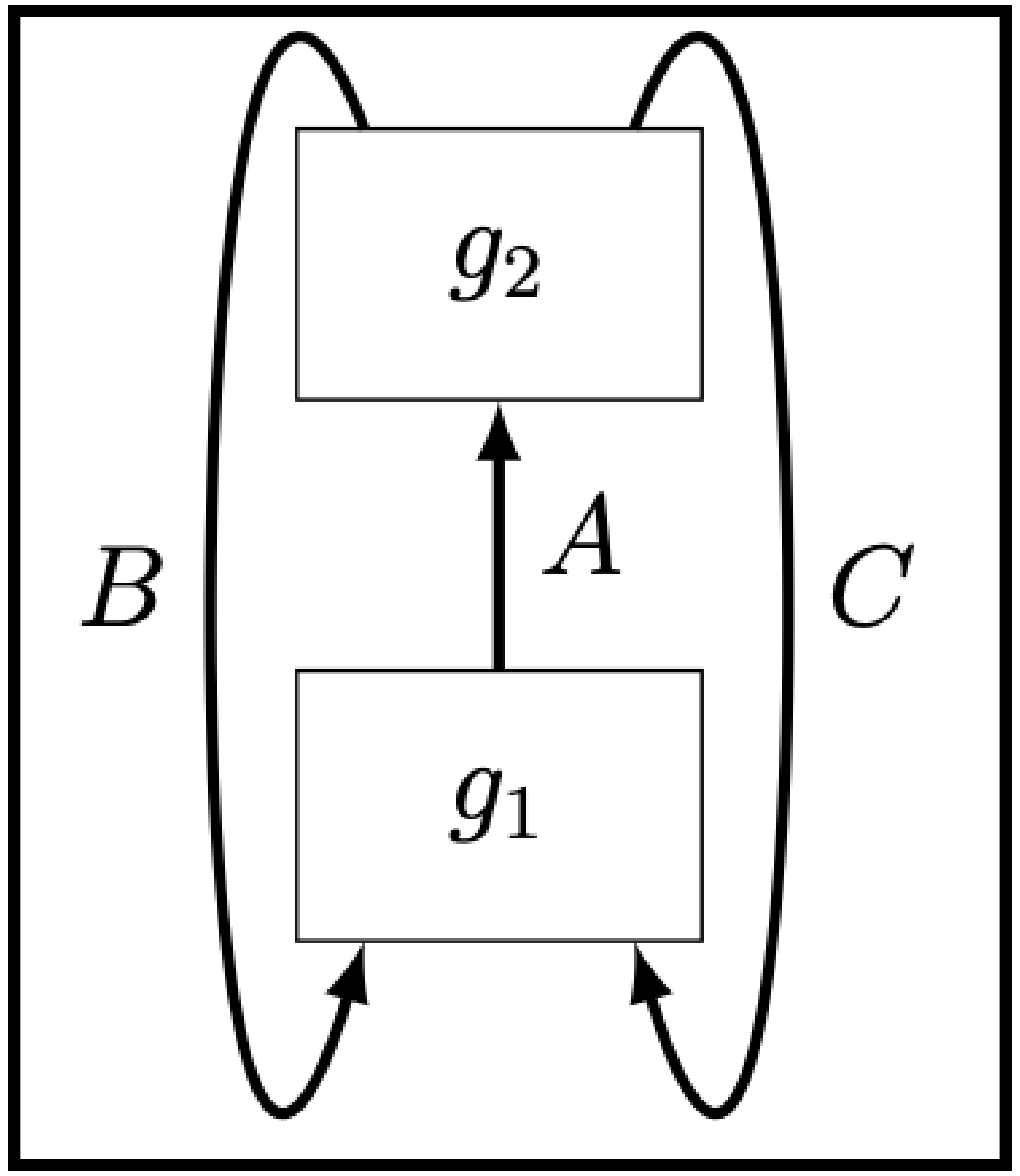
- It admits attractors of exponential period in the size of the network.
- Unary Prediction is **P-**hard.
- Binary Prediction is **PSPACE**-hard.
- Prediction Change is **PSPACE**-hard.

So now… How do we show that a family is strongly universal?

Idea: Find a family that is "easy" to simulate which is also strongly universal.

$\mathcal{G}$-Networks

$$\mathcal{G} = \{g_1 : Q^2 \to Q, g_2 : Q \to Q^2\}$$



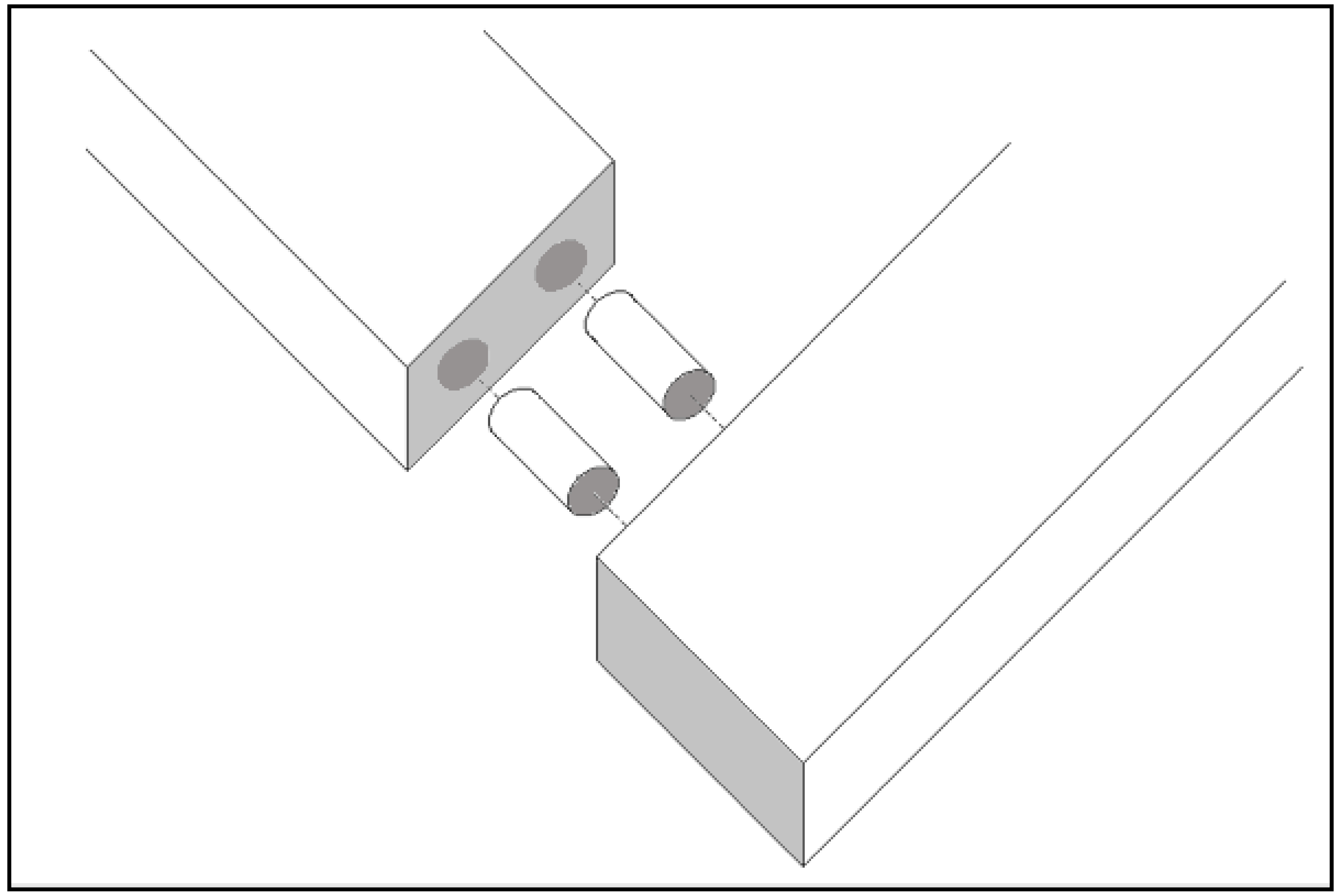
$$\mathcal{G}_{mon,2} = \{\wedge : \{0,1\}^2 \to \{0,1\}^2, \vee : \{0,1\}^2 \to \{0,1\}^2\}$$

Theorem. The family of all $\mathcal{G}_{mon,2}$-networks is **strongly universal.**

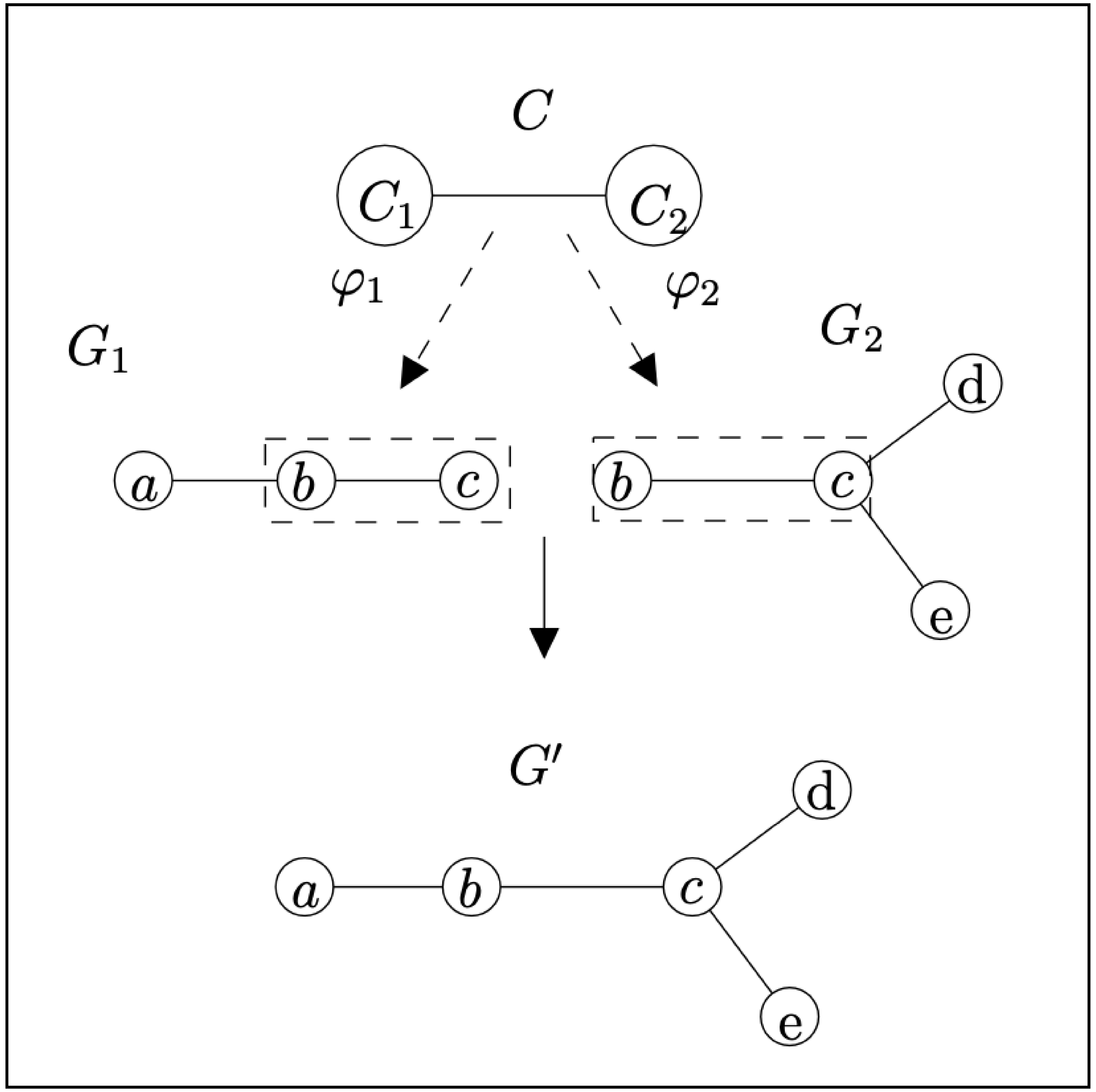
How can we use this modular structure in our simulation framework?

Simulate gates independently and then, “glue” them (Symmetry vs Asymmetry).

Glueing



source:
https://en.wikipedia.org/wiki/Dowel#/media/File:Woodworking-joint-butt-dowel.gif



Dynamical constraints for glueing

How do we simulate asymmetry inside a symmetric simulator?

Coherent $\mathcal{G}-$gadgets

- Pseudo-orbits (orbits in which not every node respects the global rule):

  -simulate the input/output relations of a gate inside the gadget.

- Constant simulation time.

Theorem: If a CSAN family $\mathcal{F}$ admits coherent $\mathcal{G}$-gadgets then, it simulates the family of all $\mathcal{G}$-networks in **constant time** and **linear space.**

Coherent gadgets

Corollary: If a CSAN family $\mathcal{F}$ admits coherent $\mathcal{G}_{mon,2}$-gadgets then, it is **strongly universal.**

Corollary: If a CSAN family $\mathcal{F}$ admits coherent $\mathcal{G}_{conj}$-gadgets then, it has attractors with **super-polynomial period.**

Summary of results

Family/Update schemes	Parallel $\subsetneq$	Block sequential $\subsetneq$	Local clocks $\subsetneq$	General periodic
Conjunctive networks	BP	BP	BP	**SPA+NC PRED**
Locally positive	BP	BP	**SU**	**SU**
Signed conjunctive	BP	**SU**	**SU**	**SU**
Min-max networks	BP	**SU**	**SU**	**SU**

- BP = Bounded period attractors
- SPA = Super polynomial period attractors
- SU = Strongly universal

Perspectives

1. **Glueing as a general operation on automata networks.**
2. **Study of $\mathcal{G}$-networks.**
3. **Update schemes: going beyond periodic schemes.**