Synthesis of Boolean Networks from Single Cell Trajectory-based Constraints

by an automatic inference of Boolean networks from static and dynamical knowledge on a system
Automatic synthesis of Boolean networks from static and dynamical information on a system

**STATIC KNOWLEDGE:**
- Graphs of known / assumed / inferred interactions between biological components
- Domain of compatible BNs (Prior Knowledge Network - PKN)

**DYNAMICAL INFORMATION:**
- Expression / activity measurements, at ≠ times, under ≠ conditions
- Dynamics to reproduce

Example:
- Gene expression in CMP:
  - Flt3 = 1
  - Gfi1 = 0
- Gene expression in macrophage:
  - Flt3 = 1
  - Gfi1 = 1

Graphs of known / assumed / inferred interactions between biological components.
Formalism: Boolean network (BN)

Some definitions

A configuration is a vector $x \in \{0, 1\}^n$

An observation is a vector $a \in \{0, 1, 'NA'\}^n$

A configuration $x$ is compatible with an observation $a$ if $\forall i \in [n], a_i=1 \Rightarrow x_i=1$ et $a_i=0 \Rightarrow x_i=0$

Example for a BN with 3 nodes:

$\Rightarrow$ the configuration 011 means:
- gene 1 is silenced
- genes 2 & 3 are expressed

A Boolean network of dimension $n$

is a function $f: \{0, 1\}^n \to \{0, 1\}^n$

$\forall i \in [n], f_i: \{0, 1\}^n \to \{0, 1\}$
Formalism: Boolean network (BN)

Some definitions

A **configuration** is a vector $x \in \{0, 1\}^n$

An **observation** is a vector $a \in \{0, 1, 'NA'\}^n$

A configuration $x$ is compatible with an observation $a$ if $\forall i \in [n], a_i=1 \Rightarrow x_i=1$ et $a_i=0 \Rightarrow x_i=0$

A **Boolean network of dimension** $n$

is a function $f: \{0, 1\}^n \to \{0, 1\}^n$

$\forall i \in [n], f_i: \{0, 1\}^n \to \{0, 1\}$

Example of a BN with 3 nodes:

$$
\begin{align*}
    f_1(x) & := \neg x_2 \\
    f_2(x) & := \neg x_1 \\
    f_3(x) & := \neg x_1 \land x_2
\end{align*}
$$

Asynchronous dynamics of $f$: 

A configuration $x$ is updated as follows:

1. Update $x_1$:
   - If $a_1=1$, then $x_1=1$
   - If $a_1=0$, then $x_1=0$

2. Update $x_2$:
   - If $a_2=1$, then $x_2=1$
   - If $a_2=0$, then $x_2=0$

3. Update $x_3$:
   - If $a_3=1$, then $x_3=1$
   - If $a_3=0$, then $x_3=0$

4. Update $x_2$:
   - If $a_2=1$, then $x_2=1$
   - If $a_2=0$, then $x_2=0$

5. Update $x_1$:
   - If $a_1=1$, then $x_1=1$
   - If $a_1=0$, then $x_1=0$

6. Update $x_3$:
   - If $a_3=1$, then $x_3=1$
   - If $a_3=0$, then $x_3=0$

7. Update $x_2$:
   - If $a_2=1$, then $x_2=1$
   - If $a_2=0$, then $x_2=0$

8. Update $x_1$:
   - If $a_1=1$, then $x_1=1$
   - If $a_1=0$, then $x_1=0$

9. Update $x_3$:
   - If $a_3=1$, then $x_3=1$
   - If $a_3=0$, then $x_3=0$
Formalism: Boolean network (BN)

Some definitions

A **configuration** is a vector $x \in \{0, 1\}^n$

An **observation** is a vector $a \in \{0, 1, 'NA'\}^n$

A configuration $x$ is compatible with an observation $a$ if $\forall i \in [n], a_i=1 \Rightarrow x_i=1$ et $a_i=0 \Rightarrow x_i=0$

A **Boolean network of dimension $n$** is a function $f: \{0, 1\}^n \rightarrow \{0, 1\}^n$

$\forall i \in [n], f_i: \{0, 1\}^n \rightarrow \{0, 1\}$

Example of a BN with 3 nodes:

- $f_1(x) := \neg x_2$
- $f_2(x) := \neg x_1$
- $f_3(x) := \neg x_1 \land x_2$

Asynchronous dynamics of $f$:

- Fixpoints
Formalism: Boolean network (\(BN\))

Some definitions

A configuration is a vector \(x \in \{0, 1\}^n\)

An observation is a vector \(a \in \{0, 1, 'NA'\}^n\)

A configuration \(x\) is compatible with an observation \(a\) if \(\forall i \in [n], a_i=1 \Rightarrow x_i=1\) et \(a_i=0 \Rightarrow x_i=0\)

A Boolean network of dimension \(n\)

is a function \(f : \{0, 1\}^n \rightarrow \{0, 1\}^n\)

\(\forall i \in [n], f_i : \{0, 1\}^n \rightarrow \{0, 1\}\)

Semantics (synchronous, asynchronous, etc.):

strong impact on prediction of trajectories

➤ we rely on Most Permissive Boolean Networks
(Paulevé et al, Nature Comm. 2020)

 textbox:

brings stronger modelling guarantee w.r.t. to quantitative systems

➤ lower cost: avoid the state space explosion
Principle of the synthesis method

Satisfiability problem

We use logic programming with Answer-Set Programming to encode the synthesis problem:

- we obtain a big equation, where variables relate to the logical functions in the Boolean network.

Each solution = BN showing the complete bifurcation process matching with scRNA-seq data.

Solver: clingo
Can scale to BNs with thousands of components (genes) depending on the properties ➤ see ICTAI 2019 paper

Main lines of the logic program:

- the description of a BN
- the domain of its functions = PKN
- the way to compute its dynamic = semantics
- the properties of its dynamics = observations

The solver enumerates the solutions (solutions = BNs compatible with data = models)
Methodology to model from scRNA-seq

scRNA-seq differentiation data: gene measurements across cells at different stage of differentiation

1) From data, we use trajectory reconstruction (e.g. STREAM) to obtain differentiation branches and bifurcation points
Methodology to model from scRNA-seq

From scRNA-seq data to dynamical constraints

1) From data, we use **trajectory reconstruction** (e.g. STREAM) to obtain differentiation branches and bifurcation points

![Diagram of trajectory reconstruction with marked points](image)

2) For each extremity of branches, **we select a pool of cells** from which **we binarize activity of genes** (possibly unknown for some of them)

Or we can use statistics from STREAM, highlighting genes of interest (Transition Genes, Leaf Genes, Diverging Genes)
We translate the branches into Boolean dynamical properties:

a) **positive reachability:**
   there is a path from the beginning to the end of each branch

b) **negative reachability:**
   there is no path between the diverging branches

c) **stable properties:**
   leafs of the graph are interpreted as trap spaces or attractors (for now fixed points)

d) **universality in the properties of the reachable fixed points:**
   - we can ensure that, from a time point, no other fixed points than those given are reachable
   - we can account for observations in different mutants
4) The possible Boolean functions are generated from a **prior knowledge network** (PKN) 

Can be extract from interaction databases (e.g. could be a full export of SIGNOR)
Blood cell differentiation

scRNA-seq data: 1656 cells, 4768 genes
from mouse hematopoietic stem and progenitor cell differentiation

at each point: around 2400 genes with a binarized value
5 positive reachability (trajectory between successive points)
1 negative reachability (no trajectory between branches)
3 trap spaces (phenotype genes in final points stay fixed)

Prior knowledge network: SIGNOR (proteins / proteins-family / complexes / phenotypes / fusion proteins)
⇒ 5454 nodes, 18125 edges

1) optimisation for graph reduction (350 nodes with existential constraints)
2) model enumeration on the reduced graph
Thank you for your attention!
Do you have questions?

Our tool “BoNesis”: [github.com/bioasp/bonesis](https://github.com/bioasp/bonesis)

Synthesis of Boolean Networks from Biological Dynamical Constraints using Answer-Set Programming
Stéphanie Chevalier, Christine Froidevaux, Andrei Zinovyev, Loïc Paulevé

Synthesis and Simulation of Ensembles of Boolean Networks for Cell Fate Decision
Stéphanie Chevalier, Vincent Noël, Laurence Calzone, Andrei Zinovyev, Loïc Paulevé

Reconciling qualitative, abstract, and scalable modeling of biological networks
Loïc Paulevé, Juraj Kolcak, Thomas Chatain, Stefan Haar

📧 stephanie.chevalier@universite-paris-saclay.fr
📧 loic.pauleve@labri.fr
📧 andrei.zinovyev@curie.fr