

# Boolean Network Identification from Multiplex Time Series Data

M. Ostrowski<sup>1</sup>, L. Paulevé<sup>2</sup>, T. Schaub<sup>1</sup>, A. Siegel<sup>3</sup>, C. Guziolowski<sup>4</sup>

<sup>1</sup> Potsdam University, Germany

<sup>2</sup> CNRS/LRI, Université Paris-Sud, France

<sup>3</sup> CNRS/Inria-DYLISS, Rennes, France

<sup>4</sup> École Centrale de Nantes, IRCCyN, Nantes, France

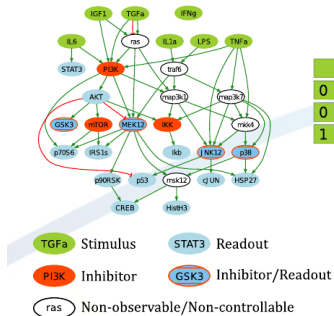
13th Conference on Computational Methods in Systems Biology  
16-18 September, 2015 - Nantes, France

## Challenge

**Refine a model** with respect to **time series (partial) observations**

## Challenge

### Phosphoproteomics



Stimuli	Inhibitors	Readouts
0 0 0 1 0 0 0	0 0 0 0 0 0 0	0.12 0.95 0.02 0.21 0.10
0 0 0 0 0 1 0	0 0 0 0 0 0 1	0.32 0.01 0.25 0.05 0.92
1 0 0 0 0 0 0	0 1 0 0 0 0 0	0.09 0.17 0.86 0.43 0.78

⋮  
 Combinatorial  
 perturbations

⋮  
 Phosphorylation  
 activity in [0,1]

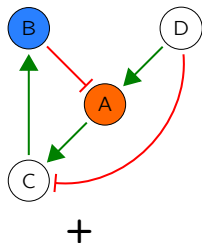
Typical approach:

- Early-response hypothesis: 2 points are considered.
- Pseudo steady-state approach: cannot infer cycles.
- Difficult to predict dynamical models that can reproduce the data.

⇒ move to more general challenge of learning model from time-series data.

## Challenge

Prior Knowledge Network (topology)



+

⇒

### Dynamical models

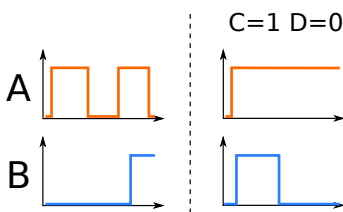
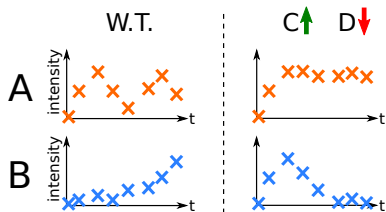
(Boolean networks)

$$A' = D \wedge \neg B$$

$$B' = C$$

$$C' = A \wedge \neg D ; C' = A \vee \neg D$$

Multiplex Time Series Data



## Our contribution

We want **all models** (Boolean Networks)

- 1 compatible with the prior knowledge network (topology);
- 2 that can reproduce the time series data.

### Necessary conditions for reproducing time series data

- Quickly invalidate models.
- No false negative.
- False positives can be filtered out *a posteriori* using model-checking.

### Distance between Boolean Networks and time series data

- When no valid models exist, find close ones (optimization).

### Implementation using Answer-Set Programming (ASP)

- Declarative approach.
- Efficient solver for solution enumeration and optimization.

## Boolean Networks

## Definition

**Specification:**  $F = (f_1, \dots, f_n)$

where each  $f_i : \mathbb{B}^n \rightarrow \mathbb{B}$ ,  $i \in \{1, \dots, n\}$ :

- associates the **next value of node  $i$**  to each global state;
- typically depends on a few other nodes.

**Semantics:** **general updating** mode.

$$\forall x, x' \in \mathbb{B}^n, x \neq x', \quad x \rightarrow x' \stackrel{\Delta}{\Leftrightarrow} \forall i \in \{1, \dots, n\}, x_i \neq x'_i \Rightarrow x'_i = f_i(x)$$

## Example

$$\begin{aligned} f_1(x) &= \neg x_2 \vee x_3 \\ f_2(x) &= x_1 \\ f_3(x) &= \neg x_2 \vee x_3 \end{aligned} \quad \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

Remarks: **non-deterministic dynamics**; no chronometry.

## Boolean Networks

## Definition

**Specification:**  $F = (f_1, \dots, f_n)$

where each  $f_i : \mathbb{B}^n \rightarrow \mathbb{B}$ ,  $i \in \{1, \dots, n\}$ :

- associates the **next value of node  $i$**  to each global state;
- typically depends on a few other nodes.

**Semantics:** **general updating** mode.

$$\forall x, x' \in \mathbb{B}^n, x \neq x', \quad x \rightarrow x' \stackrel{\Delta}{\Leftrightarrow} \forall i \in \{1, \dots, n\}, x_i \neq x'_i \Rightarrow x'_i = f_i(x)$$

## Example

$$\begin{aligned} f_1(x) &= \neg x_2 \vee x_3 \\ f_2(x) &= x_1 \\ f_3(x) &= \neg x_2 \vee x_3 \end{aligned} \quad \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

Remarks: **non-deterministic dynamics**; no chronometry.

## Boolean Networks

## Definition

**Specification:**  $F = (f_1, \dots, f_n)$

where each  $f_i : \mathbb{B}^n \rightarrow \mathbb{B}$ ,  $i \in \{1, \dots, n\}$ :

- associates the **next value of node  $i$**  to each global state;
- typically depends on a few other nodes.

**Semantics:** **general updating** mode.

$$\forall x, x' \in \mathbb{B}^n, x \neq x', \quad x \rightarrow x' \stackrel{\Delta}{\Leftrightarrow} \forall i \in \{1, \dots, n\}, x_i \neq x'_i \Rightarrow x'_i = f_i(x)$$

## Example

$$f_1(x) = \neg x_2 \vee x_3$$

$$f_2(x) = x_1$$

$$f_3(x) = \neg x_2 \vee x_3$$

$$\begin{pmatrix} 1 \\ \mathbf{0} \\ 0 \end{pmatrix} \longrightarrow \begin{pmatrix} 1 \\ \mathbf{1} \\ 0 \end{pmatrix}$$

Remarks: **non-deterministic dynamics**; no chronometry.



## Boolean Networks

## Definition

**Specification:**  $F = (f_1, \dots, f_n)$

where each  $f_i : \mathbb{B}^n \rightarrow \mathbb{B}$ ,  $i \in \{1, \dots, n\}$ :

- associates the **next value of node  $i$**  to each global state;
- typically depends on a few other nodes.

**Semantics:** **general updating** mode.

$$\forall x, x' \in \mathbb{B}^n, x \neq x', \quad x \rightarrow x' \stackrel{\Delta}{\Leftrightarrow} \forall i \in \{1, \dots, n\}, x_i \neq x'_i \Rightarrow x'_i = f_i(x)$$

## Example

$$f_1(x) = \neg x_2 \vee x_3$$

$$f_2(x) = x_1$$

$$f_3(x) = \neg x_2 \vee x_3$$

$$\begin{array}{ccc} \begin{pmatrix} 1 \\ 0 \\ \mathbf{0} \end{pmatrix} & \longrightarrow & \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} \\ \downarrow & & \\ \begin{pmatrix} 1 \\ 0 \\ \mathbf{1} \end{pmatrix} & & \end{array}$$

Remarks: **non-deterministic dynamics**; no chronometry.

## Boolean Networks

## Definition

**Specification:**  $F = (f_1, \dots, f_n)$

where each  $f_i : \mathbb{B}^n \rightarrow \mathbb{B}$ ,  $i \in \{1, \dots, n\}$ :

- associates the **next value of node  $i$**  to each global state;
- typically depends on a few other nodes.

**Semantics:** **general updating** mode.

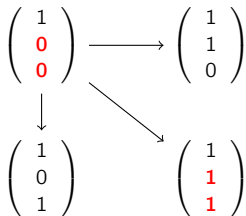
$$\forall x, x' \in \mathbb{B}^n, x \neq x', \quad x \rightarrow x' \stackrel{\Delta}{\Leftrightarrow} \forall i \in \{1, \dots, n\}, x_i \neq x'_i \Rightarrow x'_i = f_i(x)$$

## Example

$$f_1(x) = \neg x_2 \vee x_3$$

$$f_2(x) = x_1$$

$$f_3(x) = \neg x_2 \vee x_3$$



Remarks: **non-deterministic dynamics**; no chronometry.

## Boolean Networks

## Definition

**Specification:**  $F = (f_1, \dots, f_n)$

where each  $f_i : \mathbb{B}^n \rightarrow \mathbb{B}$ ,  $i \in \{1, \dots, n\}$ :

- associates the **next value of node  $i$**  to each global state;
- typically depends on a few other nodes.

**Semantics:** **general updating** mode.

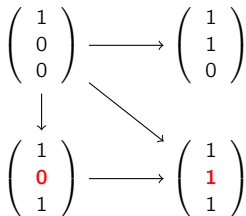
$$\forall x, x' \in \mathbb{B}^n, x \neq x', \quad x \rightarrow x' \stackrel{\Delta}{\Leftrightarrow} \forall i \in \{1, \dots, n\}, x_i \neq x'_i \Rightarrow x'_i = f_i(x)$$

## Example

$$f_1(x) = \neg x_2 \vee x_3$$

$$f_2(x) = x_1$$

$$f_3(x) = \neg x_2 \vee x_3$$



Remarks: **non-deterministic dynamics**; no chronometry.

## Boolean Networks

## Definition

**Specification:**  $F = (f_1, \dots, f_n)$

where each  $f_i : \mathbb{B}^n \rightarrow \mathbb{B}$ ,  $i \in \{1, \dots, n\}$ :

- associates the **next value of node  $i$**  to each global state;
- typically depends on a few other nodes.

**Semantics:** **general updating** mode.

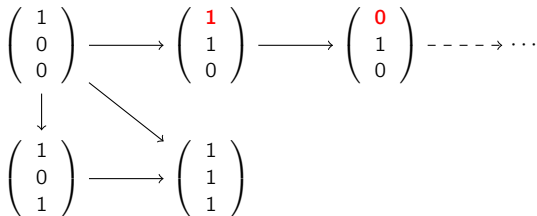
$$\forall x, x' \in \mathbb{B}^n, x \neq x', \quad x \rightarrow x' \stackrel{\Delta}{\Leftrightarrow} \forall i \in \{1, \dots, n\}, x_i \neq x'_i \Rightarrow x'_i = f_i(x)$$

## Example

$$f_1(x) = \neg x_2 \vee x_3$$

$$f_2(x) = x_1$$

$$f_3(x) = \neg x_2 \vee x_3$$



Remarks: **non-deterministic dynamics**; no chronometry.

## Boolean Networks

## Interaction graph and admissible models



$$\Leftrightarrow \exists x \in \mathbb{B}^n : f_2 \begin{pmatrix} 0 \\ x_2 \\ \vdots \end{pmatrix} < f_2 \begin{pmatrix} 1 \\ x_2 \\ \vdots \end{pmatrix}$$



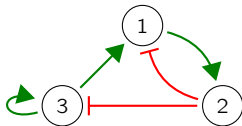
$$\Leftrightarrow \exists x \in \mathbb{B}^n : f_2 \begin{pmatrix} 0 \\ x_2 \\ \vdots \end{pmatrix} > f_2 \begin{pmatrix} 1 \\ x_2 \\ \vdots \end{pmatrix}$$

## Example

$$f_1(x) = \neg x_2 \vee x_3$$

$$f_2(x) = x_1$$

$$f_3(x) = \neg x_2 \vee x_3$$



## Boolean Networks

## Interaction graph and admissible models



$$\Leftrightarrow \exists x \in \mathbb{B}^n : f_2 \begin{pmatrix} 0 \\ x_2 \\ \vdots \end{pmatrix} < f_2 \begin{pmatrix} 1 \\ x_2 \\ \vdots \end{pmatrix}$$



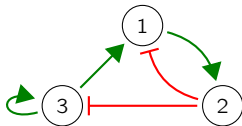
$$\Leftrightarrow \exists x \in \mathbb{B}^n : f_2 \begin{pmatrix} 0 \\ x_2 \\ \vdots \end{pmatrix} > f_2 \begin{pmatrix} 1 \\ x_2 \\ \vdots \end{pmatrix}$$

## Example

$$f_1(x) = \neg x_2 \vee x_3$$

$$f_2(x) = x_1$$

$$f_3(x) = \neg x_2 \vee x_3$$



A Boolean Network is **compatible with a Prior Knowledge Network (PKN)** iff

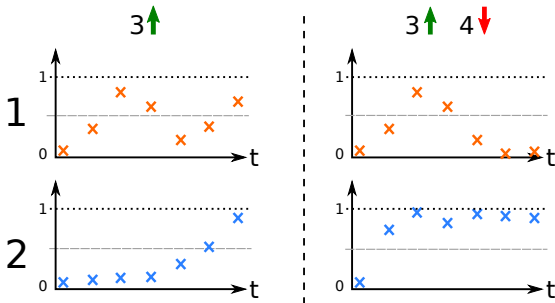
- its interaction graph is **included in the PKN**;
- its interaction graph is **simple** (no double-signed relations).

(we used DNF representations generated from the PKN)

## Multiplex Time Series Data

### Phosphoproteomics data

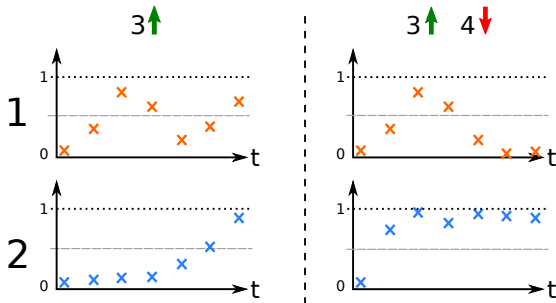
- Different experimental settings
- Partial observations
- Normalized intensities



## Multiplex Time Series Data

## Phosphoproteomics data

- Different experimental settings
- Partial observations
- Normalized intensities



## Binarization with 0.5 threshold

$$\text{with } f_3(x) = 1, \quad \begin{pmatrix} 0 \\ 0 \\ ? \end{pmatrix} \rightarrow^* \begin{pmatrix} 1 \\ 0 \\ ? \end{pmatrix} \rightarrow^* \begin{pmatrix} 0 \\ 0 \\ ? \end{pmatrix} \rightarrow^* \begin{pmatrix} 1 \\ 1 \\ ? \end{pmatrix}$$

$$\text{with } f_3(x) = 1, f_4(x) = 0, \quad \begin{pmatrix} 0 \\ 0 \\ ? \end{pmatrix} \rightarrow^* \begin{pmatrix} 0 \\ 1 \\ ? \end{pmatrix} \rightarrow^* \begin{pmatrix} 1 \\ 1 \\ ? \end{pmatrix} \rightarrow^* \begin{pmatrix} 0 \\ 1 \\ ? \end{pmatrix}$$



## A necessary condition for reachability

Verifying if  $x \rightarrow^* x'$  is hard (exact model-checking; NP-complete)  
 $\Rightarrow$  check a **weaker condition** first.

## A necessary condition for reachability

Verifying if  $x \rightarrow^* x'$  is hard (exact model-checking; NP-complete)  
 $\Rightarrow$  check a **weaker condition** first.

## Meta-states

- Each node has its value in  $\mathbb{M} \triangleq \{\boxed{0}, \boxed{1}, \boxed{0\ 1}\}$ .
- If  $u \in \mathbb{M}^n$ ,  $S(u) \triangleq \{x \in \mathbb{B}^n \mid \forall i \in \{1, \dots, n\}, x_i \in u_i\}$ :

$$u = \begin{pmatrix} \boxed{0} \\ \boxed{0\ 1} \\ \boxed{1} \\ \boxed{0\ 1} \end{pmatrix} \quad S(u) = \left\{ \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 1 \\ 1 \end{pmatrix} \right\}$$

$$(\boxed{0\ 0} \equiv \boxed{0}; \boxed{1\ 1} \equiv \boxed{1}; \boxed{1\ 0} \equiv \boxed{0\ 1})$$

## A necessary condition for reachability

**Meta-states semantics** ( $u \Rightarrow v$  with  $u, v \in \mathbb{M}^n$ )

- $g_i(u) = u_i \cup \{f_i(x) \mid x \in S(u)\}$ ;
- Asynchronous updating: only one node value changes.

$$\begin{pmatrix} u_{1..i-1} \\ \boxed{a} \\ u_{i+1..n} \end{pmatrix} \Rightarrow \begin{pmatrix} u_{1..i-1} \\ \boxed{0} \quad \boxed{1} \\ u_{i+1..n} \end{pmatrix} \quad \text{if } \exists x \in u : f_i(x) \neq a$$

## A necessary condition for reachability

**Meta-states semantics** ( $u \Rightarrow v$  with  $u, v \in \mathbb{M}^n$ )

- $g_i(u) = u_i \cup \{f_i(x) \mid x \in S(u)\}$ ;
- Asynchronous updating: only one node value changes.

$$\begin{pmatrix} u_{1..i-1} \\ \boxed{a} \\ u_{i+1..n} \end{pmatrix} \Rightarrow \begin{pmatrix} u_{1..i-1} \\ \boxed{0} \quad \boxed{1} \\ u_{i+1..n} \end{pmatrix} \quad \text{if } \exists x \in u : f_i(x) \neq a$$

## Example

$$f_1(x) = \neg x_2 \vee x_3$$

$$f_2(x) = x_1$$

$$f_3(x) = \neg x_2 \vee x_3$$

$$\begin{pmatrix} \boxed{1} \\ \boxed{0} \\ \boxed{0} \end{pmatrix}$$

## A necessary condition for reachability

**Meta-states semantics** ( $u \Rightarrow v$  with  $u, v \in \mathbb{M}^n$ )

- $g_i(u) = u_i \cup \{f_i(x) \mid x \in S(u)\}$ ;
- Asynchronous updating: only one node value changes.

$$\begin{pmatrix} u_{1..i-1} \\ \boxed{a} \\ u_{i+1..n} \end{pmatrix} \Rightarrow \begin{pmatrix} u_{1..i-1} \\ \boxed{0} \quad \boxed{1} \\ u_{i+1..n} \end{pmatrix} \quad \text{if } \exists x \in u : f_i(x) \neq a$$

## Example

$$f_1(x) = \neg x_2 \vee x_3$$

$$f_2(x) = x_1$$

$$f_3(x) = \neg x_2 \vee x_3$$

$$\begin{pmatrix} \boxed{1} \\ \boxed{0} \\ \boxed{0} \end{pmatrix} \Rightarrow \begin{pmatrix} \boxed{1} \\ \boxed{0} \quad \boxed{1} \\ \boxed{0} \end{pmatrix}$$

## A necessary condition for reachability

**Meta-states semantics** ( $u \Rightarrow v$  with  $u, v \in \mathbb{M}^n$ )

- $g_i(u) = u_i \cup \{f_i(x) \mid x \in S(u)\}$ ;
- Asynchronous updating: only one node value changes.

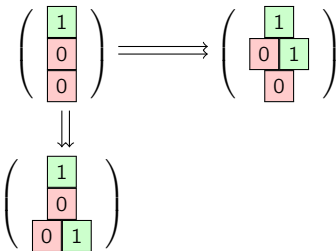
$$\begin{pmatrix} u_{1..i-1} \\ \boxed{a} \\ u_{i+1..n} \end{pmatrix} \Rightarrow \begin{pmatrix} u_{1..i-1} \\ \boxed{0} \quad \boxed{1} \\ u_{i+1..n} \end{pmatrix} \quad \text{if } \exists x \in u : f_i(x) \neq a$$

## Example

$$f_1(x) = \neg x_2 \vee x_3$$

$$f_2(x) = x_1$$

$$f_3(x) = \neg x_2 \vee x_3$$



## A necessary condition for reachability

**Meta-states semantics** ( $u \Rightarrow v$  with  $u, v \in \mathbb{M}^n$ )

- $g_i(u) = u_i \cup \{f_i(x) \mid x \in S(u)\}$ ;
- Asynchronous updating: only one node value changes.

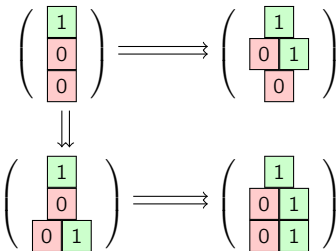
$$\begin{pmatrix} u_{1..i-1} \\ \boxed{a} \\ u_{i+1..n} \end{pmatrix} \Rightarrow \begin{pmatrix} u_{1..i-1} \\ \boxed{0} \quad \boxed{1} \\ u_{i+1..n} \end{pmatrix} \quad \text{if } \exists x \in u : f_i(x) \neq a$$

## Example

$$f_1(x) = \neg x_2 \vee x_3$$

$$f_2(x) = x_1$$

$$f_3(x) = \neg x_2 \vee x_3$$



## A necessary condition for reachability

**Meta-states semantics** ( $u \Rightarrow v$  with  $u, v \in \mathbb{M}^n$ )

- $g_i(u) = u_i \cup \{f_i(x) \mid x \in S(u)\}$ ;
- Asynchronous updating: only one node value changes.

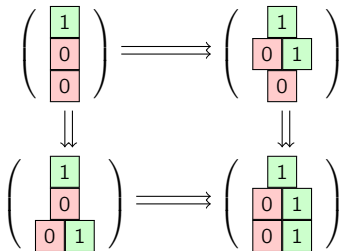
$$\begin{pmatrix} u_{1..i-1} \\ \boxed{a} \\ u_{i+1..n} \end{pmatrix} \Rightarrow \begin{pmatrix} u_{1..i-1} \\ \boxed{0} \quad \boxed{1} \\ u_{i+1..n} \end{pmatrix} \quad \text{if } \exists x \in u : f_i(x) \neq a$$

## Example

$$f_1(x) = \neg x_2 \vee x_3$$

$$f_2(x) = x_1$$

$$f_3(x) = \neg x_2 \vee x_3$$





## A necessary condition for reachability

**Meta-states semantics** ( $u \Rightarrow v$  with  $u, v \in \mathbb{M}^n$ )

- $g_i(u) = u_i \cup \{f_i(x) \mid x \in S(u)\}$ ;
- Asynchronous updating: only one node value changes.

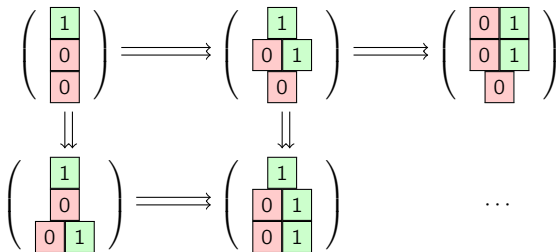
$$\begin{pmatrix} u_{1..i-1} \\ \boxed{a} \\ u_{i+1..n} \end{pmatrix} \Rightarrow \begin{pmatrix} u_{1..i-1} \\ \boxed{0} \quad \boxed{1} \\ u_{i+1..n} \end{pmatrix} \quad \text{if } \exists x \in u : f_i(x) \neq a$$

### Example

$$f_1(x) = \neg x_2 \vee x_3$$

$$f_2(x) = x_1$$

$$f_3(x) = \neg x_2 \vee x_3$$



## A necessary condition for reachability

## Theorem

$$\begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \rightarrow^* \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \text{ only if } \exists \begin{pmatrix} \boxed{x_1} \\ \vdots \\ \boxed{x_n} \end{pmatrix} \Rightarrow^* \begin{pmatrix} \boxed{y_1} \quad \boxed{\phantom{0}} \\ \vdots \\ \boxed{y_n} \quad \boxed{\phantom{0}} \end{pmatrix} = v$$

where if  $x_i = y_i$  with  $\boxed{y_i} \quad \boxed{\phantom{0}} = \boxed{0} \quad \boxed{1}$ ,  $\exists x' \in S(v) : f_i(x') = x_i$ .

Verifying  $u \Rightarrow^* v$  is easier than  $x \rightarrow^* y$ :

- $\Rightarrow$  is strictly monotonous ( $S(u) \subsetneq S(v)$ );
- no cycles;
- traces have at most  $n$  steps (until fixed point).

But is it discriminative enough?

$\Rightarrow$  check false positive rate.

## Overview &amp; Implementation

## Basic scheme

Input: Data + Prior Knowledge Network (PKN)

### 0: Binarized Data

$$\text{with } f_3(x) = 1, \quad \begin{pmatrix} 0 \\ 0 \\ ? \end{pmatrix} \rightarrow^* \begin{pmatrix} 0 \\ 1 \\ ? \end{pmatrix} \rightarrow^* \begin{pmatrix} 0 \\ 0 \\ ? \end{pmatrix} \rightarrow^* \begin{pmatrix} 1 \\ 1 \\ ? \end{pmatrix}$$

$$\text{with } f_3(x) = 1, f_4(x) = 0, \quad \begin{pmatrix} 0 \\ 0 \\ ? \end{pmatrix} \rightarrow^* \begin{pmatrix} 0 \\ 1 \\ ? \end{pmatrix} \rightarrow^* \begin{pmatrix} 1 \\ 1 \\ ? \end{pmatrix} \rightarrow^* \begin{pmatrix} 0 \\ 1 \\ ? \end{pmatrix}$$

### 1. Find all BNs compatible with the PKN that satisfy

$$\text{with } f_3(x) = 1, \quad \begin{pmatrix} 0 \\ 0 \\ ? \end{pmatrix} \Rightarrow^* \begin{pmatrix} 0 \\ 1 \\ ? \end{pmatrix} \Rightarrow^* \begin{pmatrix} 0 \\ 0 \\ ? \end{pmatrix} \Rightarrow^* \begin{pmatrix} 1 \\ 1 \\ ? \end{pmatrix}$$

$$\text{with } f_3(x) = 1, f_4(x) = 0, \quad \begin{pmatrix} 0 \\ 0 \\ ? \end{pmatrix} \Rightarrow^* \begin{pmatrix} 0 \\ 1 \\ ? \end{pmatrix} \Rightarrow^* \begin{pmatrix} 1 \\ 1 \\ ? \end{pmatrix} \Rightarrow^* \begin{pmatrix} 0 \\ 1 \\ ? \end{pmatrix}$$

(initial value of non-observed nodes is the same across all experiments)

### 2. (optional) Filter out false positives using model-checking

⇒ No false negatives!

## Overview &amp; Implementation

## Optimization scheme

When no admissible Boolean Network can reproduce the data...

⇒ find all BNs with the minimum errors.

- Allow mis-matches with the binarized data;
- Minimize the mean square error w.r.t. continuous data.

But we are over-approximating the satisfaction criteria (optimistic):

- returned  $\widehat{\text{mse}}$  is smaller than the true minimum mse;
- $\widehat{\text{mse}} = \text{mse}$  if there is at least one true positive (w.r.t. errors).

### Implementation

- Answer-Set Programming (declarative approach);
- Gringo+Clasp<sup>1</sup>: efficient solver with solutions optimization and enumeration.

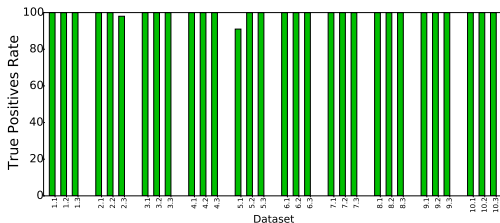
<sup>1</sup>: <http://potassco.sourceforge.net>

## Evaluation

### True positives rate

#### Synthetic data

- 10 PKNs random subset of a master PKN;
- PKNs gather between 13 and 17 nodes;
- 3 BNs/PKN; 6 observed nodes; 16 time points; 10 experimental settings.



#### Results

- Time for enumeration: < 2s;
- Results: 12-2,640 BNs (subset-minimal specs.); up to 54,000 for all.
- Posterior model-checking: 1-300s/BN.

## Evaluation

### Other benchmarks

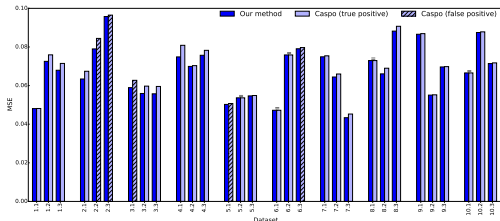
Comparison with [MacNamara Phys. Biol. 2012]

- Same data, recover the predicted BN, and others.
- Faster by **several orders of magnitude**;
- **Exhaustive** identification.

Comparison with **2-points inference methods** using Caspo

[Guziolowski et al., Bioinformatics 2013]

- Advantage of taking into account the dynamics (more accurate predictions);
- Can infer **feedback loops**.



### Summary

- Boolean Networks compatible with a PKN that can reproduce time series data.
- Two stages: (1) **quick over-approximation** of resulting set; (2) filter out.
- Declarative implementation.

### Features

- **Exhaustive** identification.
- **No false negatives** for exact matching.
- **Optimization** in case of (necessary) errors (MSE under-estimated)  
⇒ towards scoring PKNs.

### Future work

- Adaptive and multiple discretization thresholds.
- Address larger and real networks ( $\geq 100$  nodes).

# Boolean Network Identification from Multiplex Time Series Data

M. Ostrowski<sup>1</sup>, L. Paulevé<sup>2</sup>, T. Schaub<sup>1</sup>, A. Siegel<sup>3</sup>, C. Guziolowski<sup>4</sup>

<sup>1</sup> Potsdam University, Germany

<sup>2</sup> CNRS/LRI, Université Paris-Sud, France

<sup>3</sup> CNRS/Inria-DYLISS, Rennes, France

<sup>4</sup> École Centrale de Nantes, IRCCyN, Nantes, France

13th Conference on Computational Methods in Systems Biology  
16-18 September, 2015 - Nantes, France