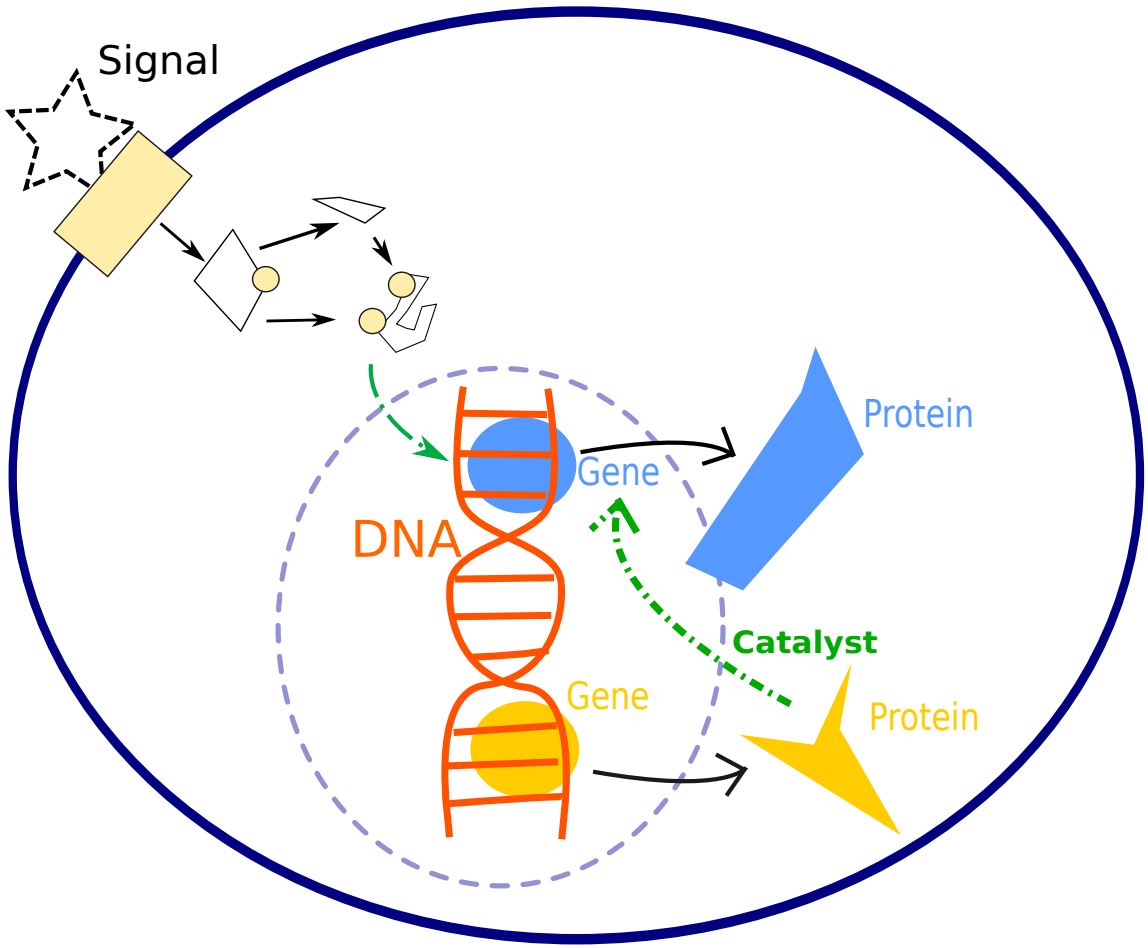
Boolean Network Identification from Perturbation Time Series Data

M. Ostrowski[1], L. Paulevé[2], T. Schaub[1], A. Siegel[3], C. Guziolowski[4]

[1] Potsdam University, Germany
[2] CNRS/LRI, Université Paris-Sud, France
[3] CNRS/Inria-DYLISS, Rennes, France
[4] École Centrale de Nantes, LS2N, Nantes, France

CMSB 2015 - 13th Conference on Computational Methods in Systems Biology
Extended version in BioSystems journal (2016)

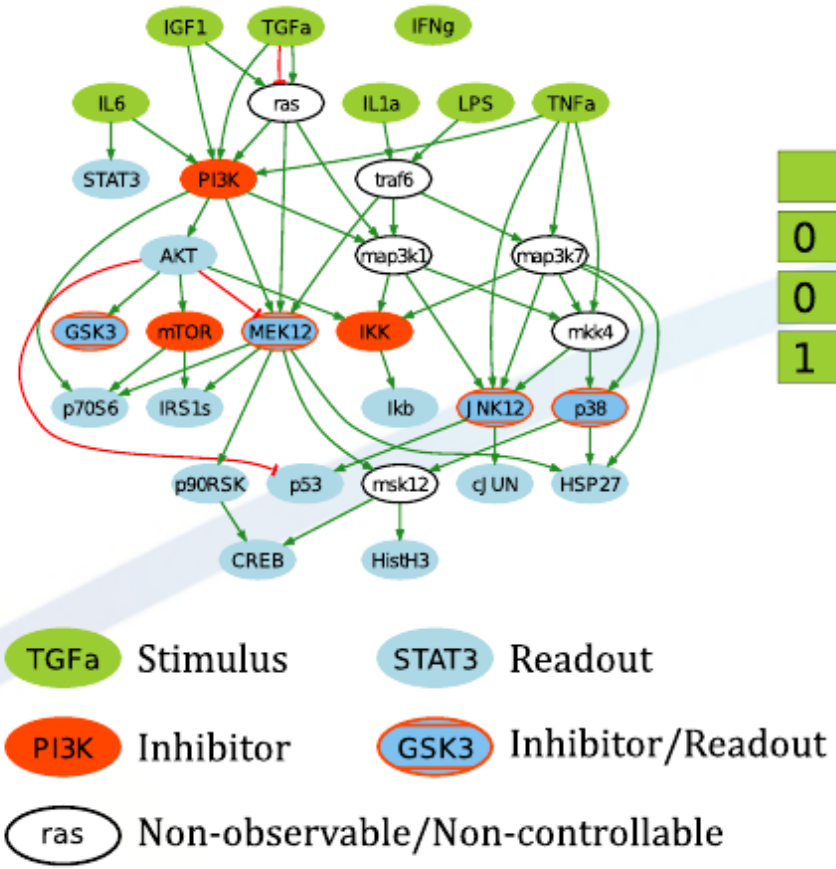
Gene and signalling networks



Data on Signalling Networks

Phosphoproteomics

Putative (signed) causal network

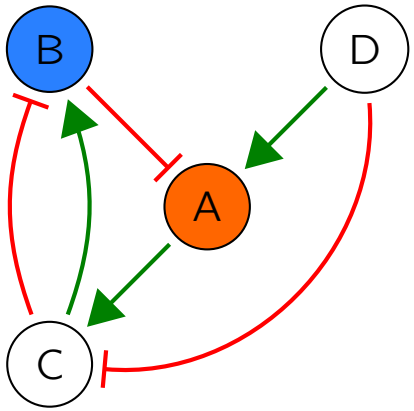


Stimuli	Inhibitors	Readouts
0 0 0 1 0 0 0	0 0 0 0 0 0 0	0.12 0.95 0.02 0.21 0.10
0 0 0 0 0 1 0	0 0 0 0 0 0 1	0.32 0.01 0.25 0.05 0.92
1 0 0 0 0 0 0	0 1 0 0 0 0 0	0.09 0.17 0.86 0.43 0.78
⋮	⋮	⋮
Combinatorial perturbations		Phosphorylation activity in [0,1]

Partial state observations at discrete times.

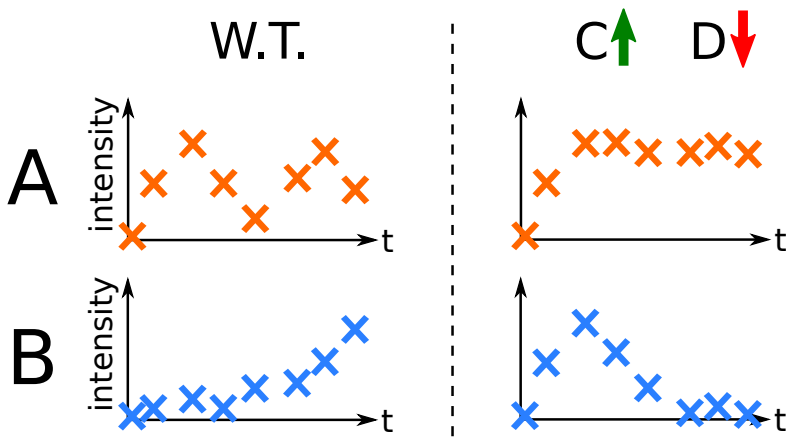
Challenge

Prior Knowledge Network (PKN)
(over-approx of causal network)



+

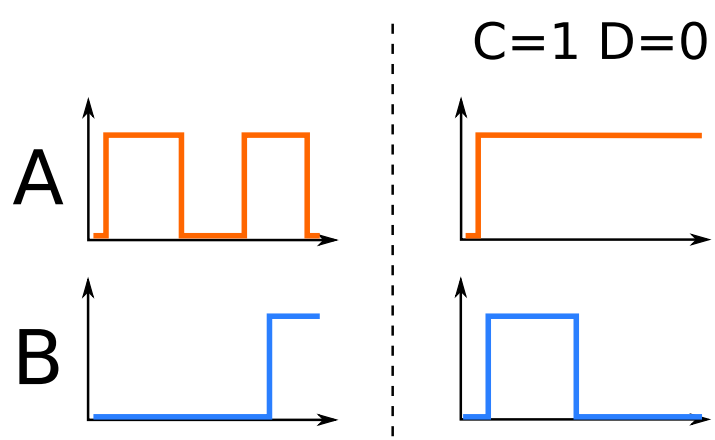
Perturbation Time Series Data



⇒

Dynamical models
(Boolean networks)
$A' = D \wedge \neg B$
$B' = C$
$C' = A \wedge \neg D$; $C' = A \vee \neg D$

+ PKN score



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 model exist, find close ones (optimization).
- Distance between causal network and time series data

Implementation using Answer-Set Programming (ASP)

- Declarative approach (specify the constraints, not the algorithm)
- Efficient solver for solution enumeration and optimization.

Outline

Outline

Gene networks

also applies to signalling networks

Ordinary Differential Equations (ODEs)

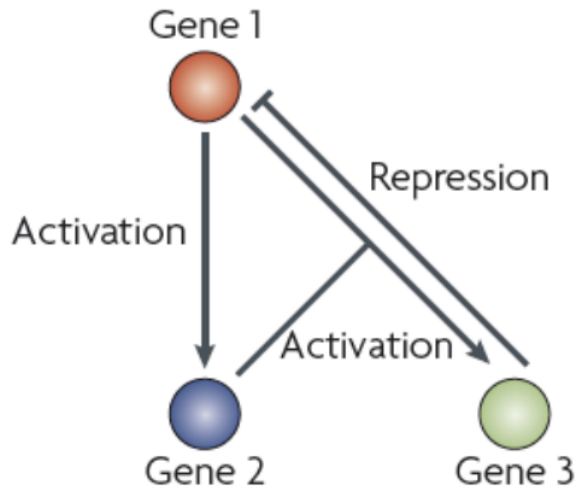
a

$$\frac{d(gene_1)}{dt} = k_{1,s} \cdot \frac{1}{1 + k_{1,3} \cdot gene_3} - k_{1,d} \cdot gene_1$$

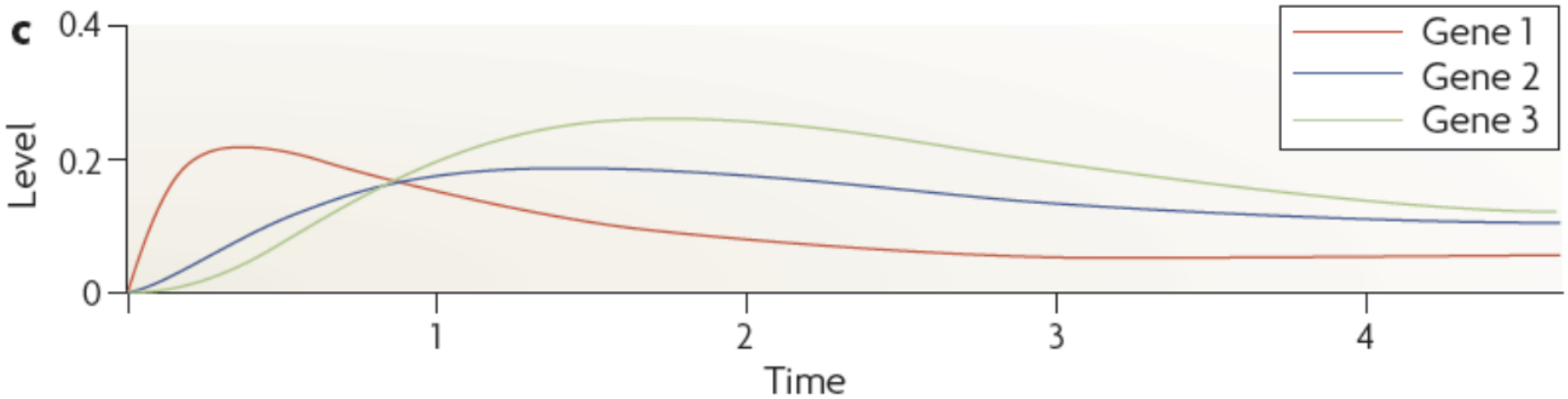
$$\frac{d(gene_2)}{dt} = k_{2,s} \cdot \frac{k_{2,1} \cdot gene_1}{1 + k_{2,1} \cdot gene_1} - k_{2,d} \cdot gene_2$$

$$\frac{d(gene_3)}{dt} = k_{3,s} \cdot \frac{k_{3,1} \cdot gene_1 \cdot k_{3,2} \cdot gene_2}{(1 + k_{3,1} \cdot gene_1) \cdot (1 + k_{3,2} \cdot gene_2)} - k_{3,d} \cdot gene_3$$

b



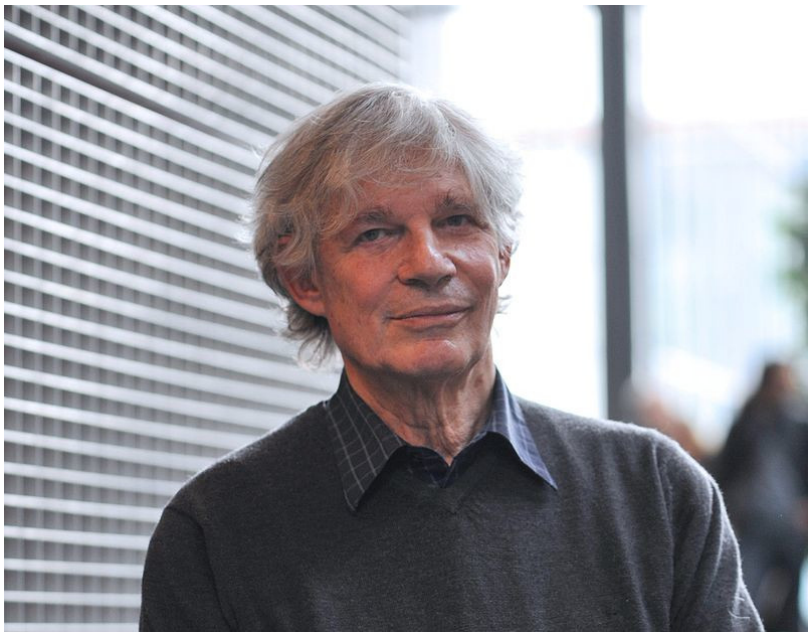
c



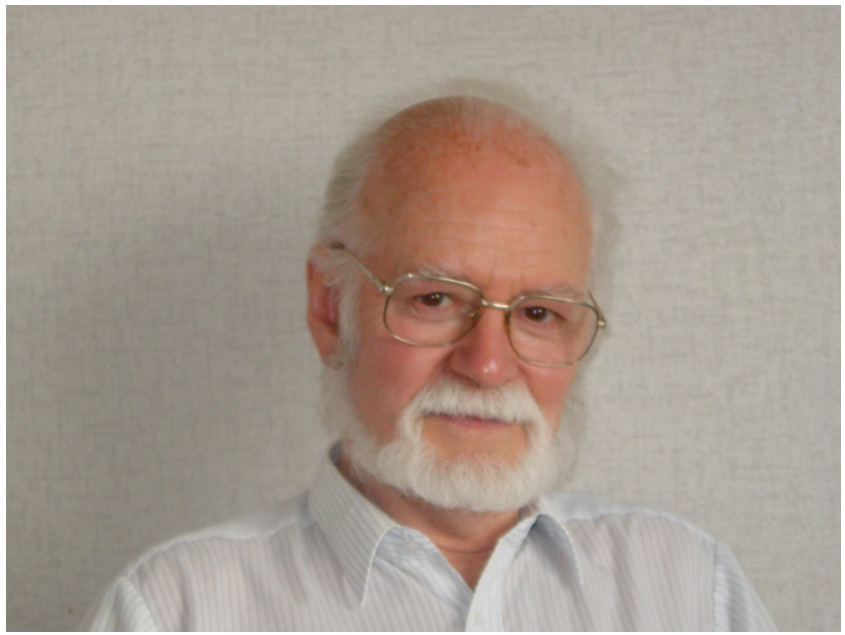
$\Rightarrow$ too many parameters; overly precise.

Qualitative modelling of gene networks

around 1970



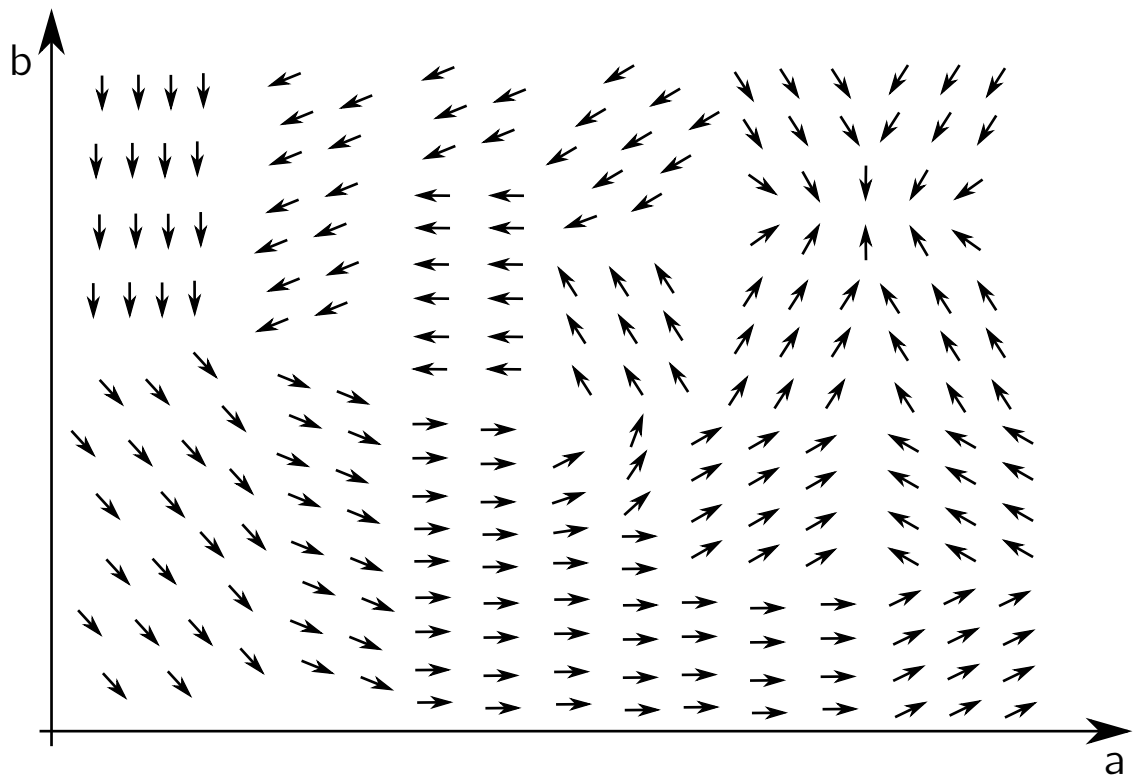
Stuart Kauffman



René Thomas

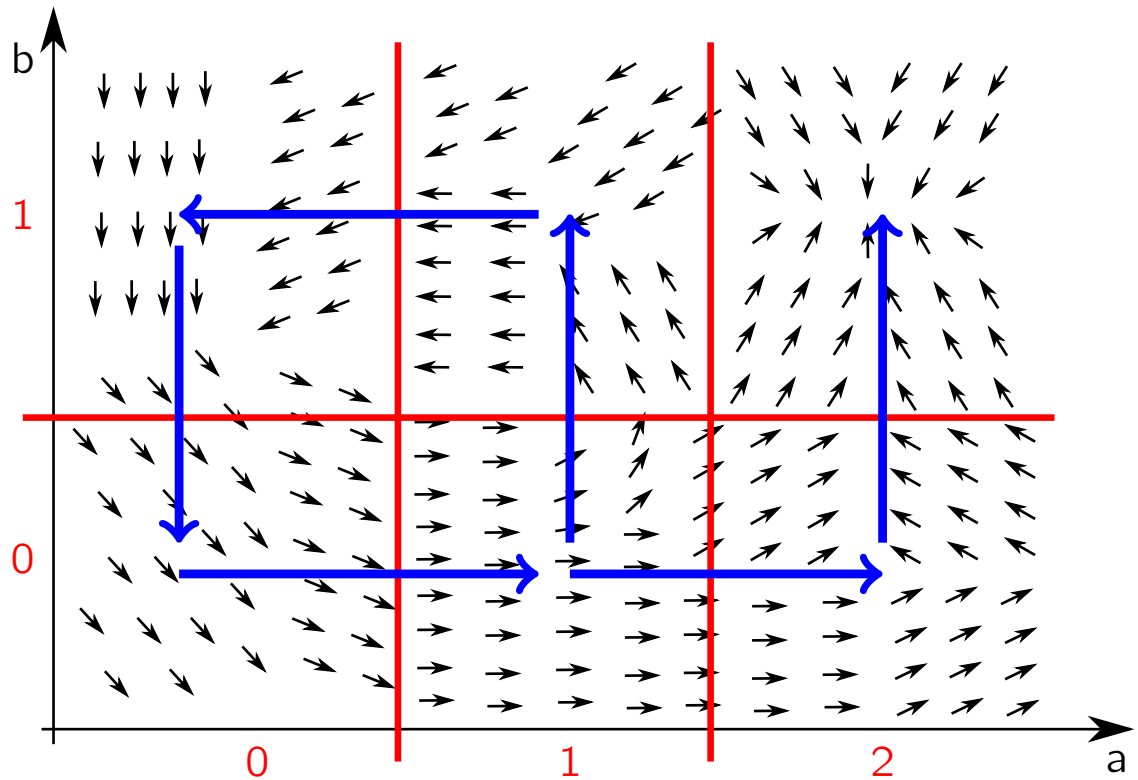
A gene is either ON or OFF; active/medium/inactive.
No chronometry; logical time only (succession of states).

Qualitative models



Qualitative models

Quantization of continuous phase space



Boolean Networks

Definition

Specification: $F = (f_1, \cdots, f_n)$
where each $f_i : \mathbb{B}^n \to \mathbb{B}$, $i \in \{1, \cdots, n\}$:

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

Semantics: general updating mode.

$$\forall x, x' \in \mathbb{B}^n, x \neq x', \qquad x \to x' \overset{\Delta}{\Leftrightarrow} \forall i \in \{1, \ldots, n\}, x_i \neq x'_i \Rightarrow x'_i = f_i(x)$$

Example

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

$$f_2(x) = x_1$$

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

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

Remarks: non-deterministic dynamics; no chronometry; **exponential state space**.

Boolean Networks

Definition

Specification: $F = (f_1, \cdots, f_n)$
where each $f_i : \mathbb{B}^n \to \mathbb{B}$, $i \in \{1, \cdots, n\}$:

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

Semantics: general updating mode.

$$\forall x, x' \in \mathbb{B}^n, x \neq x', \qquad x \to x' \overset{\Delta}{\Leftrightarrow} \forall i \in \{1, \ldots, n\}, x_i \neq x'_i \Rightarrow x'_i = f_i(x)$$

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; **exponential state space**.

Boolean Networks

Definition

Specification: $F = (f_1, \cdots, f_n)$
where each $f_i : \mathbb{B}^n \to \mathbb{B}$, $i \in \{1, \cdots, n\}$:

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

Semantics: general updating mode.

$$\forall x, x' \in \mathbb{B}^n, x \neq x', \qquad x \to x' \overset{\Delta}{\Leftrightarrow} \forall i \in \{1, \ldots, n\}, x_i \neq x'_i \Rightarrow x'_i = f_i(x)$$

Example

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

$f_2(x) = x_1$

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

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

$$\downarrow$$

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

Remarks: non-deterministic dynamics; no chronometry; **exponential state space**.

Boolean Networks

Definition

Specification: $F = (f_1, \cdots, f_n)$
where each $f_i : \mathbb{B}^n \to \mathbb{B}$, $i \in \{1, \cdots, n\}$:

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

Semantics: general updating mode.

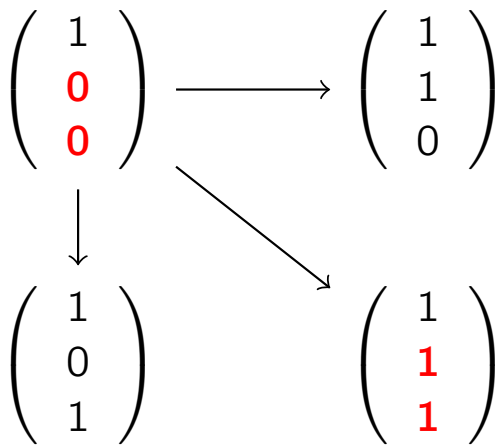
$$\forall x, x' \in \mathbb{B}^n, x \neq x', \qquad x \to x' \overset{\Delta}{\Leftrightarrow} \forall i \in \{1, \ldots, n\}, x_i \neq x'_i \Rightarrow x'_i = f_i(x)$$

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; **exponential state space**.

Boolean Networks

Definition

Specification: $F = (f_1, \cdots, f_n)$
where each $f_i : \mathbb{B}^n \to \mathbb{B}$, $i \in \{1, \cdots, n\}$:

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

Semantics: general updating mode.

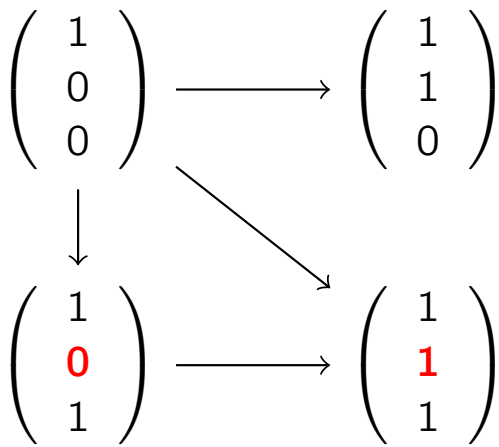
$$\forall x, x' \in \mathbb{B}^n, x \neq x', \qquad x \to x' \overset{\Delta}{\Leftrightarrow} \forall i \in \{1, \ldots, n\}, x_i \neq x'_i \Rightarrow x'_i = f_i(x)$$

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; **exponential state space**.

Boolean Networks

Definition

Specification: $F = (f_1, \cdots, f_n)$
where each $f_i : \mathbb{B}^n \to \mathbb{B}$, $i \in \{1, \cdots, n\}$:

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

Semantics: general updating mode.

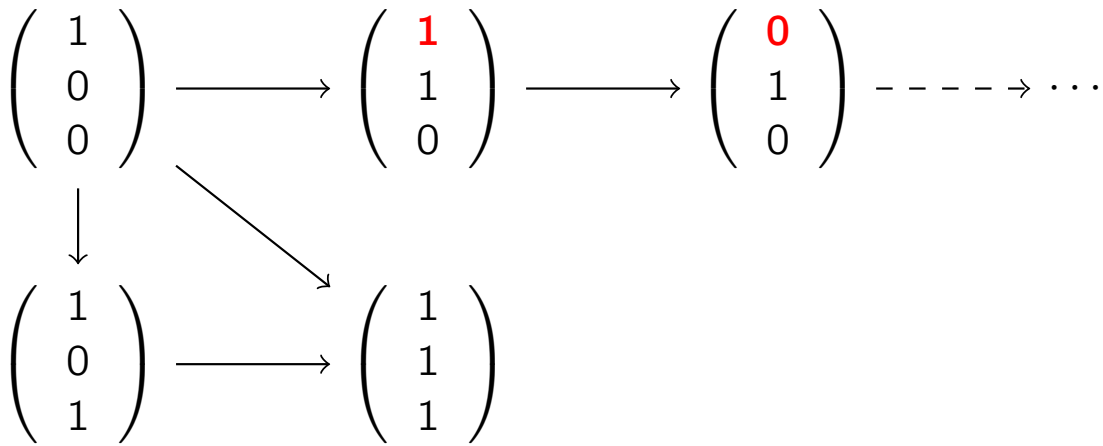
$$\forall x, x' \in \mathbb{B}^n, x \neq x', \qquad x \to x' \overset{\Delta}{\Leftrightarrow} \forall i \in \{1, \ldots, n\}, x_i \neq x'_i \Rightarrow x'_i = f_i(x)$$

Example

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

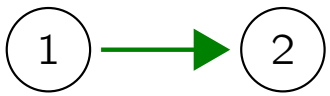
$f_2(x) = x_1$

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



Remarks: non-deterministic dynamics; no chronometry; **exponential state space**.

Influence graph (signed causal graph)



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

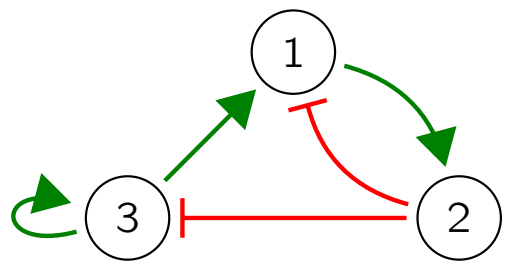
$$\overset{\Delta}{\Leftrightarrow} \exists x \in \mathbb{B}^n : f_2 \begin{pmatrix} 0 \\ x_2 \\ \vdots \end{pmatrix} > f_2 \begin{pmatrix} 1 \\ x_2 \\ \vdots \end{pmatrix}$$

Example

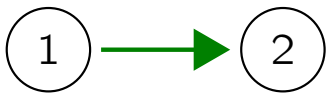
$f_1(x) = \neg x_2 \vee x_3$

$f_2(x) = x_1$

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



Influence graph (signed causal graph)



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

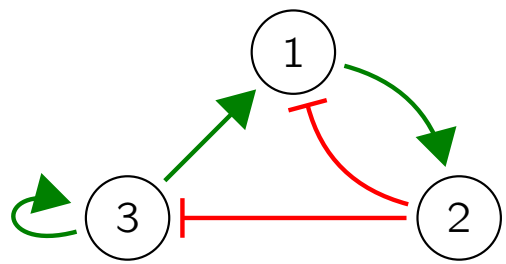
$$\overset{\Delta}{\Leftrightarrow} \exists x \in \mathbb{B}^n : f_2 \begin{pmatrix} 0 \\ x_2 \\ \vdots \end{pmatrix} > f_2 \begin{pmatrix} 1 \\ x_2 \\ \vdots \end{pmatrix}$$

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

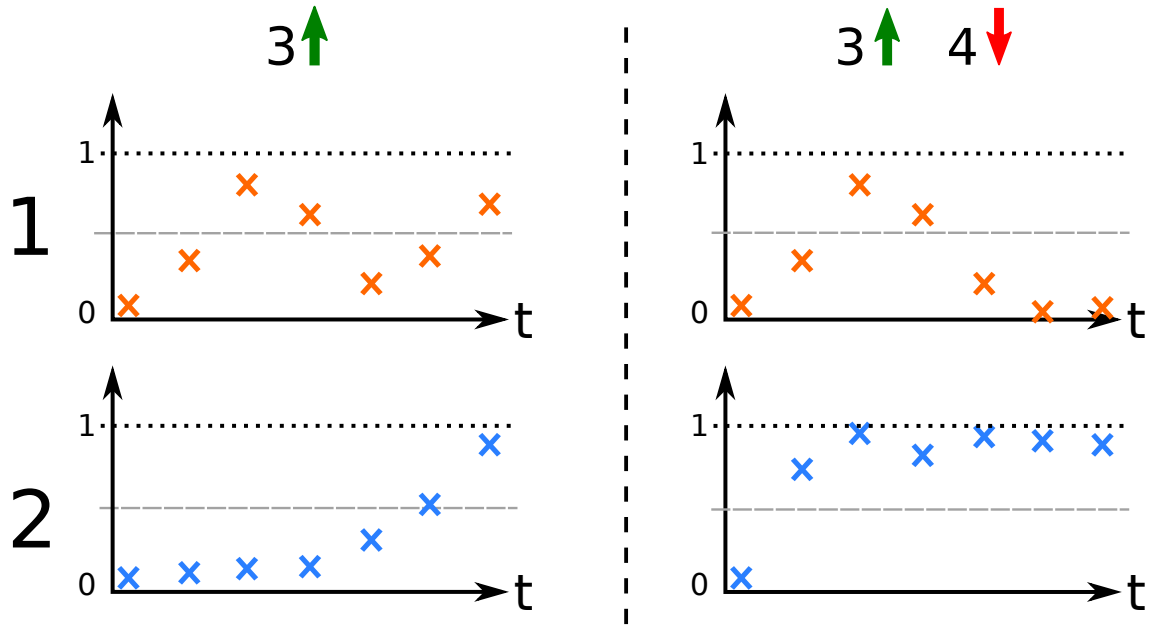
- its influence graph is included in the PKN;
- its influence graph is simple (no double-signed relations).

Outline

Perturbation Time Series Data

Phosphoproteomics data

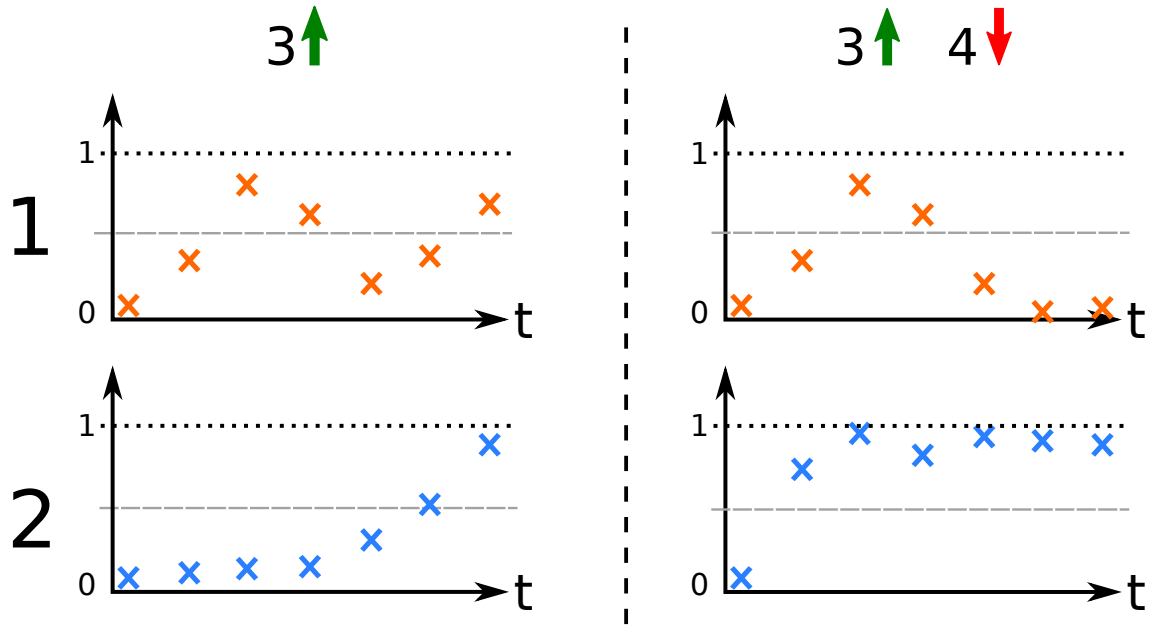
- Different experimental settings
- Partial observations
- Normalized intensities



Perturbation Time Series Data

Phosphoproteomics data

- Different experimental settings
- Partial observations
- Normalized intensities



Binarization with 0.5 threshold

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

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

A necessary condition for reachability

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

A necessary condition for reachability

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

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

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

A necessary condition for reachability

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

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

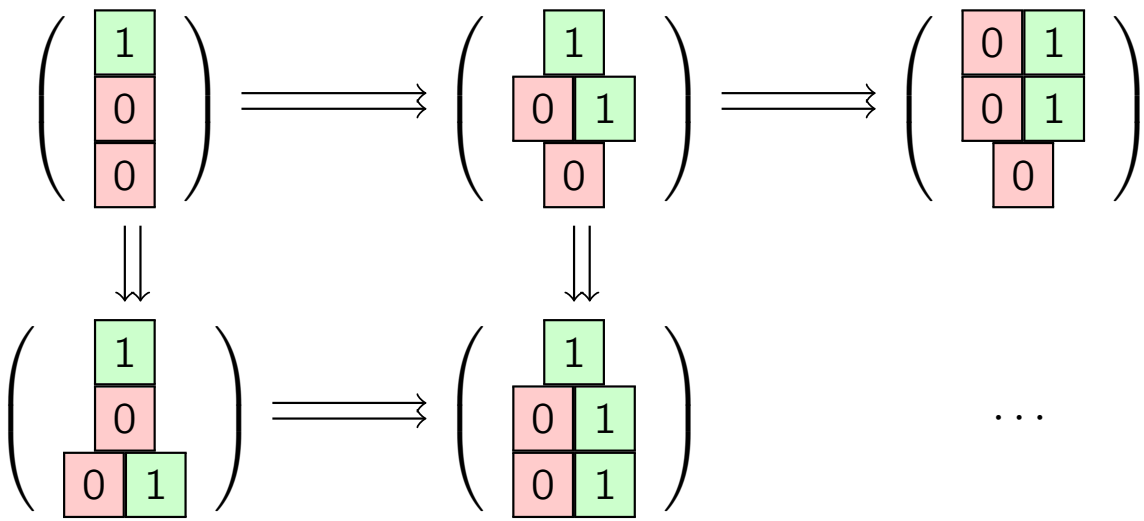
$$\begin{pmatrix} u_{1..i-1} \\ \boxed{a} \\ u_{i+1..n} \end{pmatrix} \rightrightarrows \begin{pmatrix} u_{1..i-1} \\ \boxed{0}\boxed{1} \\ u_{i+1..n} \end{pmatrix} \qquad \text{if } \exists x \in u : f_i(x) \neq a$$

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

(abstract interpretation)

Theorem

$$\begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \to^* \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \textit{ only if } \begin{pmatrix} \boxed{x_1} \\ \vdots \\ \boxed{x_n} \end{pmatrix} \rightrightarrows^* \begin{pmatrix} \boxed{y_1}\boxed{} \\ \vdots \\ \boxed{y_n}\boxed{} \end{pmatrix} = v$$

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

Verifying **u** $\rightrightarrows^*$ **v is easier than x** $\to^*$ **y**:

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

But is it discriminative enough?
$\Rightarrow$ check false positive rate.

Overview

Input: Data + Prior Knowlodge Network (PKN)

0: Binarized Data

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

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

1. Find all BNs compatible with the PKN that satisfy

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

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

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

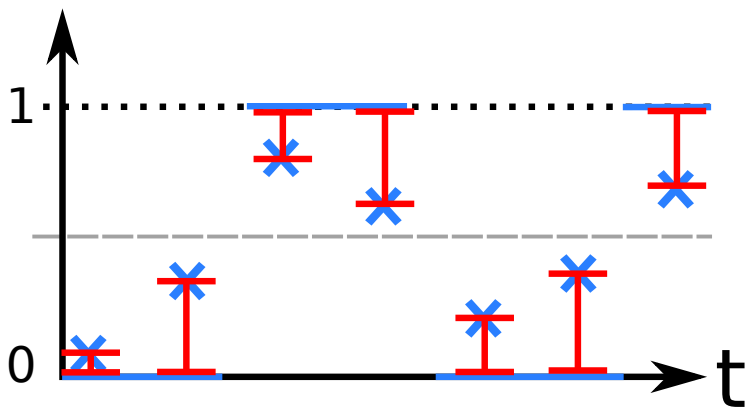
2. (optional) Filter out false positives using model-checking

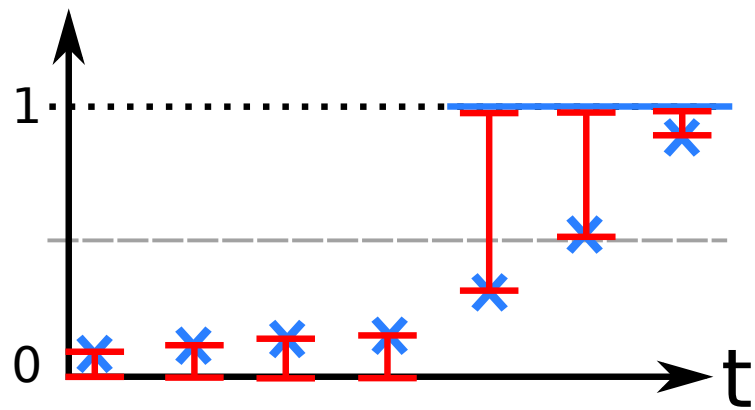
$\Rightarrow$ No false negatives!

Optimization/BN scoring

When no admissible Boolean Network can reproduce the data. . .
$\Rightarrow$ find all BNs with the minimum errors.

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

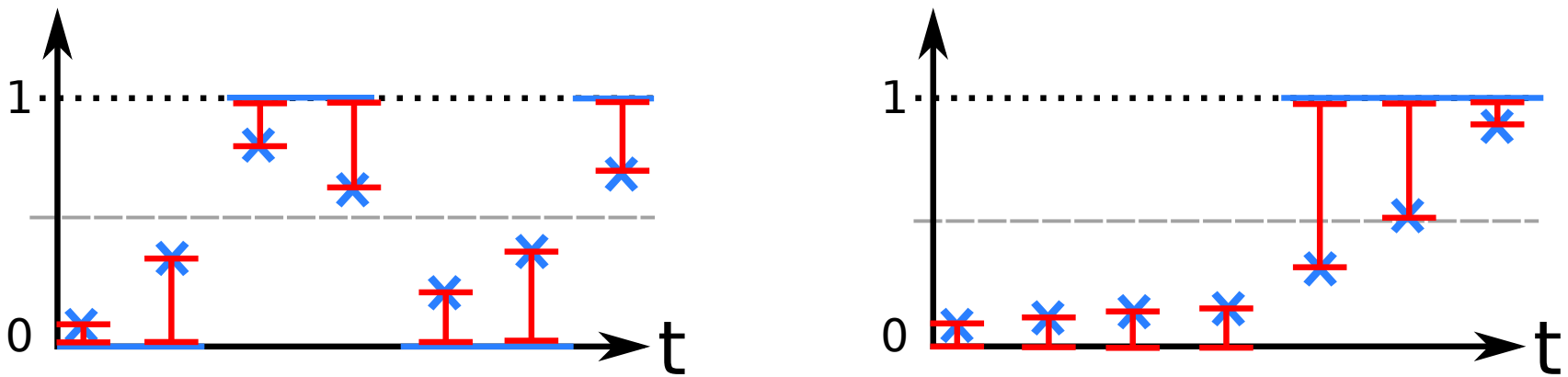




Optimization/BN scoring

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.

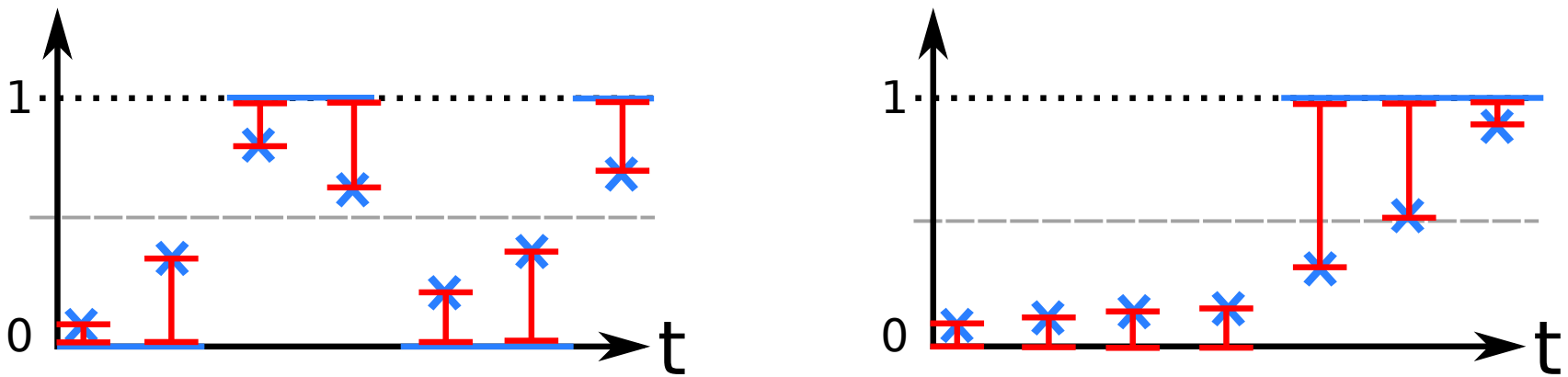


Remark: all returned BNs have the same error ⇒ **causal graph score**
(RMSE(BinData) when there exists a BN which can reproduce the data).

Optimization/BN scoring

When no admissible Boolean Network can reproduce the data...
$\Rightarrow$ find all BNs with the minimum errors.

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



Remark: all returned BNs have the same error $\Rightarrow$ **causal graph score**
(RMSE(BinData) when there exists a BN which can reproduce the data).

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

- for sure, $\widehat{\mathrm{mse}} \leq \mathrm{mse}$;
- $\widehat{\mathrm{mse}} = \mathrm{mse}$ if there is at least one true positive (w.r.t. errors).

Outline

Implementation

Solver

- Answer-Set Programming (declarative approach: specify constraints);
- Gringo+Clasp[1]: efficient solver with solutions optimization and enumeration.

[1]: `http://potassco.sourceforge.net`

Implementation

Solver

- Answer-Set Programming (declarative approach: specify constraints);
- Gringo+Clasp[1]: efficient solver with solutions optimization and enumeration.

[1]: `http://potassco.sourceforge.net`

Boolean functions are represented in DNF (disjonction of conjonctions).

Set of **solutions**:

$$f_1(x) = x_2$$

$$f_1(x) = x_1 \wedge x_4$$

$$f_1(x) = x_1 \vee x_3$$

$$f_1(x) = x_1 \vee (x_3 \wedge x_4)$$

Implementation

Solver

- Answer-Set Programming (declarative approach: specify constraints);
- Gringo+Clasp[1]: efficient solver with solutions optimization and enumeration.

[1]: `http://potassco.sourceforge.net`

Boolean functions are represented in DNF (disjonction of conjonctions).

Set of **minimal solutions**:

$f_1(x) = x_2$ cardinal-minimal

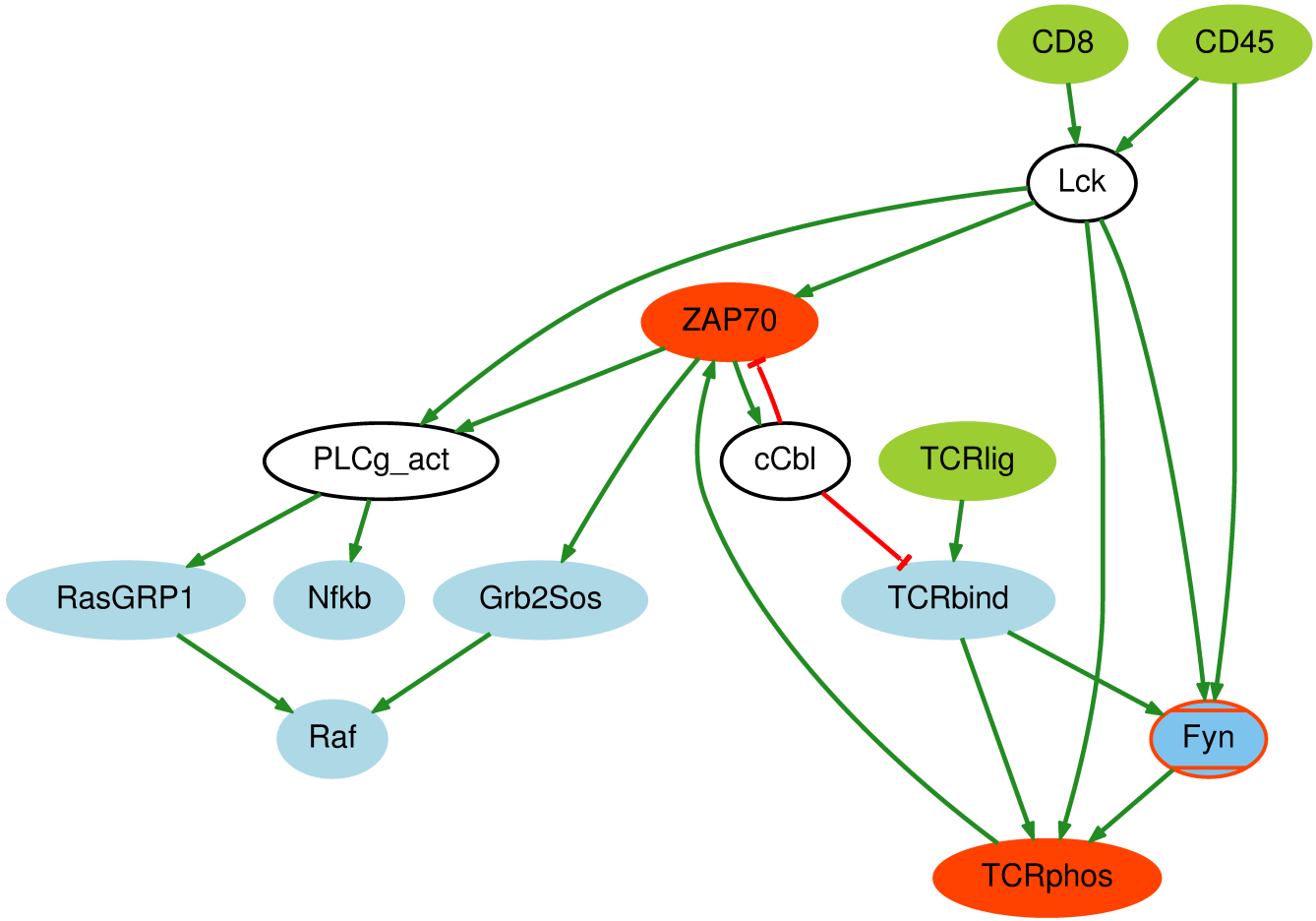
$f_1(x) = x_1 \wedge x_4$

$f_1(x) = x_1 \vee x_3$ subset-minimal

$f_1(x) = x_1 \vee (x_3 \wedge x_4)$

$\Rightarrow$ in general, we are interested in subset-minimal solutions

Evaluation of true positive rate



- Generate 3 different BNs with compatible causal network (inc in PKN)
- Generate perturbation time series synthetic data
- Identification with PKN + data; evaluate TP rate (FN=0)
  remark: RMSE=RMSE(BinData)

Evaluation of true positive rate

		subset-minimal		
Model	Space	First	Total	TP
Case-Study A	2^{21}	$<$ 1s	54 (2s)	100%
TNFα-EGF		1s	64 (3s)	100%
13 nodes, 16 edges		$<$1s	36 (3s)	100%
Case-Study B.1	2^{37}	1s	5,544 (3min)	100%
TCR signaling		1s	2,901 (90s)	100%
14 nodes, 22 edges		1s	6,510 (4min)	100%
Case-Study B.2	2^{49}	1s	73,962 (1h40)	100%
TCR signaling		1s	68,338 (1h30)	78%
16 nodes, 25 edges		1s	74,757 (1h40)	96%
Case-Study B.3	2^{106}	5s	$>$100,000	-
TCR signaling		5s	$>$100,000	-
40 nodes, 58 edges		5s	$>$100,000	-

Similar results with optimization with errors (altered PKN as input)
See [Ostrowksi et al BioSystems 2016] paper.

Conclusion

Summary

- Identify dynamical models compatible with a prior causal network. . .
- . . . and can reproduce perturbation time series data.
- Artificial intelligence + model verification (ASP/SAT + abstract interpretation)

Features

- Exhaustive identification; no false negatives (but false positives).
- Optimization in case of errors (MSE under-estimated)
  $\Rightarrow$ scoring prior causal network
- No issue with cycles in the causal network.
- Results can refine the prior causal network.

Future work

- Adaptative and multiple discretization thresholds.
- Use of causal network scoring? (for learning a better PKN)
  include the number of BN solutions in the score?
- Coupling with machine learning? (CTBNs, . . .)