# A Touristic Guide on the Updates of Boolean Networks

Athénaïs Vaginay
MCF éq. CODAG (GREYC, Unicaen)

Séminaire Algo, GREYC — 2024 Dec 10

# Outline

1. A light intro to systems biology and formal models in general
2. Introduction to Boolean automata networks
3. Classic update modes and their limitations
4. Most-permissive mode to the rescue
5. Limitations of MP
6. Conclusion

# Introduction

# A zoo of modelling approaches

Reaction network

continuous time Markov chain

ODEs

statistical models

informal diagrams

Petri net

Boolean transition system

Boolean automata network

- There are many modelling approaches of biological systems

# A zoo of modelling approaches

Reaction network

continuous time Markov chain

ODEs

statistical models

informal diagrams

Petri net

Boolean transition system

Boolean automata network

- There are many modelling approaches of biological systems

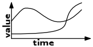
Model = rules A → B

**Reaction networks**

Transformation of reactants into products

Differential equations

$\dot{B} = f(A)$




Time series

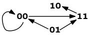
Continuous time and values

**Boolean networks**

Influence of regulators on regulees

Boolean functions (propositional logics)

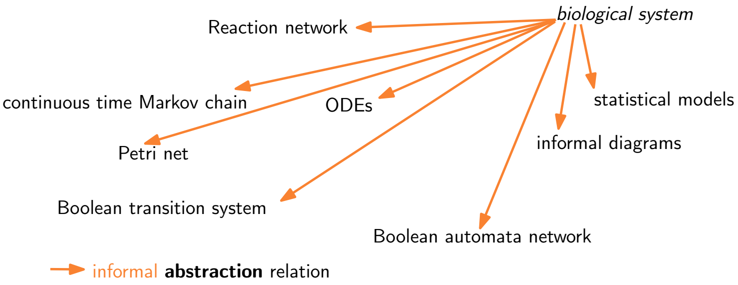
$B = f(A)$




Transition graph

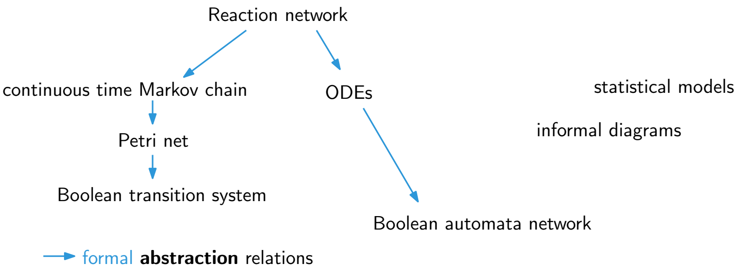
Discrete time, boolean values

# A zoo of modelling approaches




- There are many modelling approaches of biological systems
- A model is an informal abstraction of a biological system

# A zoo of modelling approaches




- There are many modelling approaches of biological systems
- A model is an informal abstraction of a biological system
- Claim: understanding the formal relationships of abstraction between modelling approaches improves the (automatic) model synthesis approaches. [Vaginay 2023]

# The notion of abstraction

## Definition (Abstraction)

Mapping between simulation traces of a **concrete** model and those of an **abstract** model, such that we can derive correct conclusions. [Fages, Soliman, 2008a]

$\Longrightarrow$ Analogy with abstract interpretation [Cousot, Cousot, 1977]

## Example

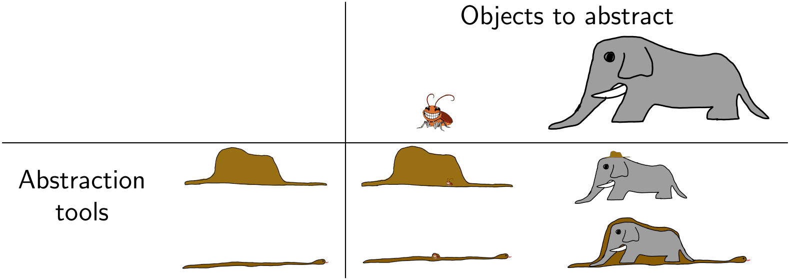
Given $x, y \in \mathbb{R}$, return the sign of $z = x + y$.

- Concrete algo: compute $z = x + y$ and then check the sign.
- Abstract algo: drop the precise values, use the **rule of signs**

| $+$ | p | n |
|---|---|---|
| p | p | ? |
| n | ? | n |

# The notion of abstraction

## Correctness and tightness, informally




Figures inspired from [Saint-Exupery 1943]

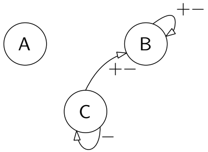
- The hat is not complete nor tight
- The snake is complete and tight.

# Boolean automata networks

# Boolean automata networks [Kauffman 1969] [Thomas 1973]

## A generalisation of cellular automata

Influence graph:




Local functions:

$$f_A := \square$$
$$f_B := (B \land \neg C) \lor (\neg B \land C)$$
$$f_C := \neg C$$

A Boolean automata network $f$ consists of $n$ local functions $\mathbb{B}^n \to \mathbb{B}$, with $\mathbb{B} = \{\blacksquare, \square\}$.

Given two automata $i$ and $j$, $i$ has a **monotone** influence over $i$ iff $i \xrightarrow{s} j \in G$ and not $i \xrightarrow{\neg s} j \in G$, $s = \{+, -\}$.

# Updates of the network

Given a Boolean network $f$, a configuration $x \in \mathbb{B}^n$, and an automaton $i$: compute the next configuration $x' \in \mathbb{B}^n$ resulting from the **update** of $i$.

$$\begin{cases} x'_j = x_j \text{ if } j \neq i \\ x'_i = \square \text{ if } f_i(x) \text{ returns False} \\ x'_i = \blacksquare \text{ if } f_i(x) \text{ returns True} \end{cases}$$

## Example

**Given:** $f = \{f_\mathrm{A} := \square; f_\mathrm{B} := (\mathrm{B} \land \neg\mathrm{C}) \lor (\neg\mathrm{B} \land \mathrm{C}); f_\mathrm{C} := \neg\mathrm{C}\}$ ; $x = \square\square\square$ ; update C.

**Result:** $x' = \square\square\blacksquare$ (since $f_C := \neg C$)

Representation as a **transition graph**:

$$\square\square\square \underset{\mathrm{C}}{\overset{\mathrm{C}}{\rightleftarrows}} \square\square\blacksquare$$

# Update modes

The update mode dictates which components can be updated at each step.
Classic update modes correspond to different compositions of the local update functions.

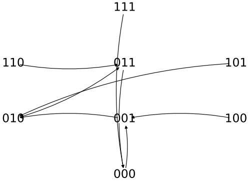
Example: a boolean network $f$ of size 3 ($\mathcal{A} = \{A, B, C\}$)

- synchronous: $\{\{A, B, C\}\}$
- asynchronous: $\{\{A\}, \{B\}, \{C\}\}$
- general asynchronous: $\mathcal{P}(\mathcal{A}) \setminus \emptyset$
- sequential blocks: $(\{A\}, \{B, C\})$
- parallel blocks: $\{(A, B, C, \ldots), (B, C, \ldots)\}$

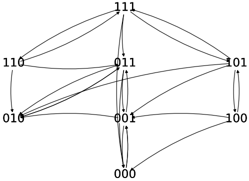
# Impact of updates modes on the transitions

$$f = \{\ f_A := 0, \quad f_B := (B \wedge \neg C) \vee (\neg B \wedge C), \quad f_C := \neg C\ \}$$

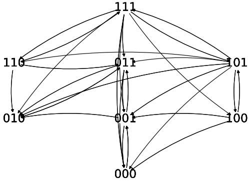
synchronous: $\{\{A, B, C\}\}$




async.: $\{\{A\}, \{B\}, \{C\}\}$




general async.: $\mathcal{P}(\mathcal{A}) \setminus \emptyset$

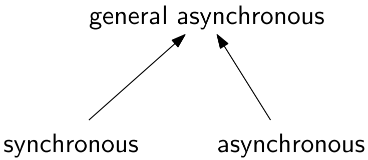



Here, ■ = 1 and □ = 0

# The impact of updates modes on the modelisation

1. Given a biological system $b$, and some known dynamics $D$
2. Derive a Boolean network $f$ and pick an update scheme $\mu$
3. If $f_\mu$ reproduces the known dynamics $D$:
4. Use $f_\mu$ to derive new knowledge $D'$ about the dynamics of $b$
5. $D$+=$D'$
6. GOTO 1
7. Else:
8. GOTO 2

# Inclusion hierarchy of update modes



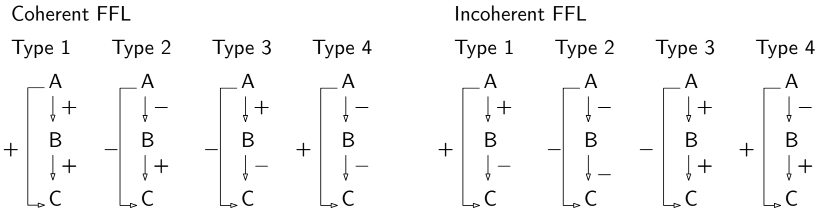

[Paulevé, Séné 2022]

General async. mode captures **correctly** (but **not tightly**) the transitions captured by the sync. and asyc. modes.

# An annoying biological system

The feed-forward loop (FFL): a three-nodes regulation motif that is *really* common in biological systems ; more frequent than expected random.

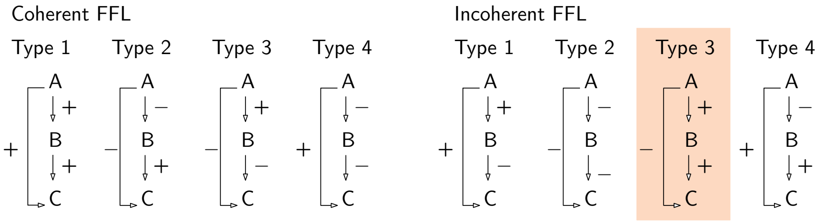
Each of the three arrows are either $+$ or $-$ $\implies$ 8 combinaisons.




[Mangan, Alon 2003]

# An annoying biological system

The feed-forward loop (FFL): a three-nodes regulation motif that is *really* common in biological systems ; more frequent than expected random.
Each of the three arrows are either $+$ or $-$ $\implies$ 8 combinaisons.



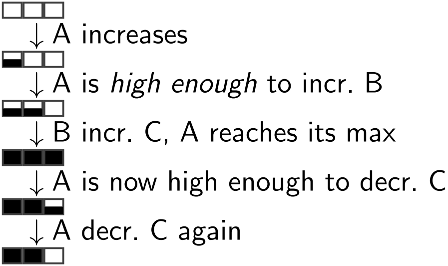

[Mangan, Alon 2003]

# An annoying biological system

## The incoherent feed-forward loop of type 3

$$f_{\mathsf{A}} := \blacksquare, \quad f_{\mathsf{B}} := \mathsf{A}, \quad f_{\mathsf{C}} := \neg\mathsf{A} \wedge \mathsf{B}$$

No classic update scheme can predict the transient activation of automata 3 that is **observed experimentally** and **modelled** with multivalued automata networks.




$$F_{\mathsf{A}} := +1 \text{ always;} \quad F_{\mathsf{B}} := +1 \text{ if } A = 2; \quad F_{\mathsf{C}} := \begin{cases} +1 \text{ if } \mathsf{B} > 2 \wedge \mathsf{A} < 3 \\ -1 \text{ if } \mathsf{A} > 3 \end{cases}$$

# A big problem

We have an example for which the general asynchronous mode is not complete enough. Does it mean **Boolean networks are not expressive enough ?**

# A big problem

We have an example for which the general asynchronous mode is not complete enough. Does it mean **Boolean networks are not expressive enough ?**

"Yes, *obviously*, because of their Boolean nature..."

# A big problem

We have an example for which the general asynchronous mode is not complete enough. Does it mean **Boolean networks are not expressive enough ?**

"Yes, *obviously*, because of their Boolean nature..."

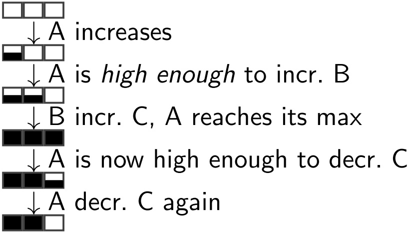
And yet... :))))

# The Most-Permissive mode (MP) to the rescue

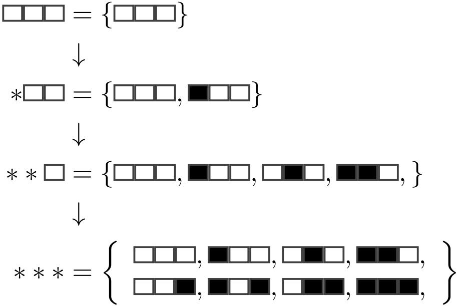
[Paulevé et al. 2020]

- Not a composition of local update functions [Paulevé, Sené 2021]
- Uses an intermediary state $*$ that is the *superposition* of both □ and ■.

Multivalued world




Permissive and Boolean worlds



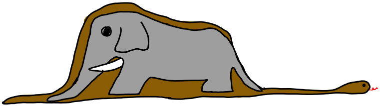
$\Longrightarrow$ Thanks to MP, ■■■ is reachable from □□□. :)))

# Guaranties of MP

MP is the **tightest complete** abstraction of **multivalued refinements** of Boolean networks (*i.e.*, of all the functions $F : \mathbb{N}^n \to \{-1, 0, +1\}^n$ whose dynamics are *compatible* with $f$).

complete
: Given a BN $f$, if an automata update can occur in a multivalued refinement of the BN, MP captures it.

tight
: Given a dynamics predicted by the MP, it is justified by the async. dynamics of at least one multivalued refinement of the BN.



# MP is not tight *enough*

A Boolean network $f$: $f_{\mathsf{A}} := \blacksquare$; $f_{\mathsf{B}} := \mathsf{A}$; $f_{\mathsf{C}} := \mathsf{B}$
Its influence graph is **monotone**: $\mathsf{A} \xrightarrow{+} \mathsf{B} \xrightarrow{+} \mathsf{C}$

MP mode predicts: $\square\square\square \rightarrow *\square\square \rightarrow **\square \rightarrow ***$
That makes $\blacksquare\square\blacksquare$ reachable from $\square\square\square$.
Justified by some multivalued refinements of $f$ that use non-monotonous influence, such as:

$$F_{\mathsf{A}} := +1 \text{ always}$$
$$F_{\mathsf{B}} := \begin{cases} +1 \text{ if } \mathsf{A} = 1 \text{ or } \mathsf{A} = 3 \\ -1 \text{ if } \mathsf{A} = 2 \end{cases}$$
$$F_{\mathsf{C}} := +1 \text{ if } \mathsf{B} > 0$$

$$\mathsf{A}_{/3} \xrightarrow{+-} \mathsf{B}_{/1} \xrightarrow{+} \mathsf{C}_{/1}$$

# Monotonic Most-Permissive

**We want:** to compute the dynamics of $f$ starting from a config. $x$ but to prevent some components from evolving non monotonically.
**Which components ?** those whose behaviour is exclusively monotone in **all** the multivalued refinements of $f$.

We now use ◪ and ◩ instead of $*$.

## Example

With the Boolean network $f$ : $f_A :=$ ■; $f_B :=$ A; $f_C :=$ B
when a component started increasing, it cannot go back to 0.

□□□ → ◪□□ → ◪◪□ → ◪◪◪ → ... → ■■■

In place of "...": any sequences of configurations in {◪, ■}$^3$.

# Monotonic Most-Permissive

## Naïve approach

Given a Boolean network $f$ and a configuration $x$.

- Build the set $\mathbf{F}$ of all the multivalued strong refinements of $f$ (those influence graph are subset of the influence graph of $f$)
- For each automaton $i$
  - Check whether there is $F \in \mathbf{F}$ such that $i$ can evolve non-monotonically.
  - If not and $i$ only increases: $i \in E^+$.
    If not and $i$ only decreases: $i \in E^-$.
- Apply the MP mode on $f$ with the constraint that
  $i \in E^+ \implies$ no transition $x \to x'$ where $x'_i =$ ⬕.
  $i \in E^- \implies$ no transition $x \to x'$ where $x'_i =$ ◪.

# Monotonic Most-Permissive

Questions:

- **How to build the sets** $E^+$ **and** $E^-$ **directly from the influence graph of** $f$ **?** So far: we have a **correct** algorithm. It is **complete ?**
- **Is the monotonic MP complete ?** yes. Is the monotonic MP **tight ?** Don't know yet.

**Hélène Siboulet**, Théo Roncalli, Loïc Paulevé

# Conclusion and perspectives

# Conclusion and perspectives

- **Boolean automata networks** are a very simple modelisation framework
- The choice of **update mode** is crucial
- Thanks to the **Most-Permissive mode**, Boolean networks can capture everything that is captured in the multivalued world.
- Necessary to tweek the MP mode, depending on what we really want to capture.
- **Monotonic MP** would be a useful restriction of MP

# Conclusion and perspectives

- **Boolean automata networks** are a very simple modelisation framework
- The choice of **update mode** is crucial
- Thanks to the **Most-Permissive mode**, Boolean networks can capture everything that is captured in the multivalued world.
- Necessary to tweek the MP mode, depending on what we really want to capture.
- **Monotonic MP** would be a useful restriction of MP

**Biology gives fun problems to computer scientists.**

# Thank you for your attention.

# Appendix

# Multivalued refinement of a BN

A multivalued network $F$ is a function $F : \mathfrak{N}^n \to \{-1, 0, +1\}$.
$F$ is a refinement of a Boolnet $f$ if for any multivalued configuration $X \in \mathfrak{N}^n$ and for any automata $a$:

$$F_a(X) > 0 \implies \exists x \in \beta(X) : f_a(x) = 1$$
$$F_a(X) < 0 \implies \exists x \in \beta(X) : f_a(x) = 0$$

where $\beta(X)$ is all the possible binarisations of $X$ respecting that

- $X_a = 0 \implies x_a = 0$
- $X_a = m_a \implies x_a = 1$

# Multivalued refinement of a BN – Example

A Boolean network $f$

$f_A := ■; f_B := A; f_C := B$

Its influence graph is **monotone**

$A \xrightarrow{+} B \xrightarrow{+} C$

A multivalued refinement of $f$:

$F_A := +1$ always

$F_B := +1$ if $A = 1$ or $A = 3$

$\quad\quad -1$ otherwise

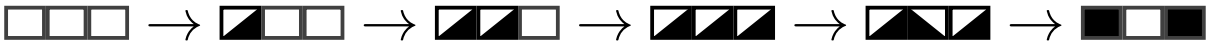
$F_C := +1$ if $B > 0$

$A_{/3} \xrightarrow{+-} B_{/1} \xrightarrow{+} C_{/1}$

Asynchronous execution of $F$:

$000 \xrightarrow{A} 100 \xrightarrow{B} 110 \xrightarrow{C} 111 \xrightarrow{A} 211 \xrightarrow{B} 201 \xrightarrow{A} 301$

In MP:



# References I


- [Cousot, Cousot, 1977]
  P. Cousot, R. Cousot,
  Abstract interpretation: A unified lattice model for static analysis of programs by construction or approximation of fixpoints,
  1977
- [Fages, Soliman, 2008a]
  F. Fages, S. Soliman,
  Abstract Interpretation and Types for Systems Biology,
  *Theoretical Computer Science*, vol. 403, pp. 52–70, 2008
- [Kauffman 1969]
  S.A. Kauffman
  Metabolic stability and epigenesis in randomly connected nets.
  1969
- [Mangan, Alon 2003]
  S. Mangan, U. Alon
  Structure and function of the feed-forward loop network motif
  2003

# References II


- [Paulevé, Séné 2022]
  L. Paulevé, S. Séné
  Boolean networks and their dynamics: the impact of updates.,
  2022
- [Paulevé, Sené 2021]
  L. Paulevé, S. Sené
  Non-deterministic updates of Boolean networks
  2021
- [Paulevé et al. 2020]
  L. Paulevé, Juri. Kolcak, T. Chatain, S. Haar
  Reconciling qualitative, abstract, and scalable modeling of biological networks,
  2020
- [Saint-Exupery 1943]
  A. de Saint-Exupéry
  Le petit prince
  1943

# References III


- [Thomas 1973]
  R. Thomas
  Boolean formalization of genetic control circuits
  1973
- [Vaginay 2023]
  A. Vaginay
  Synthesis of Boolean Networks from the Structure and Dynamics of Reaction Networks,
  2023