Scalable Causality Analysis within Qualitative Biological Networks

Montpellier - 23rd November 2012

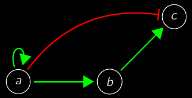
Loïc Paulevé

ETH Zürich (BISON group)

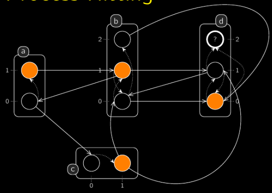
http://loicpauleve.name

Overview

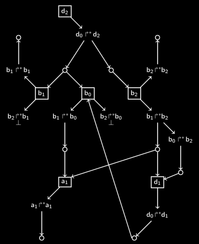
Biological network



Qualitative modelling with Process Hitting



Graph of Local Causality



Reachability analysis

Necessary processes

+ tractable for large-scale networks

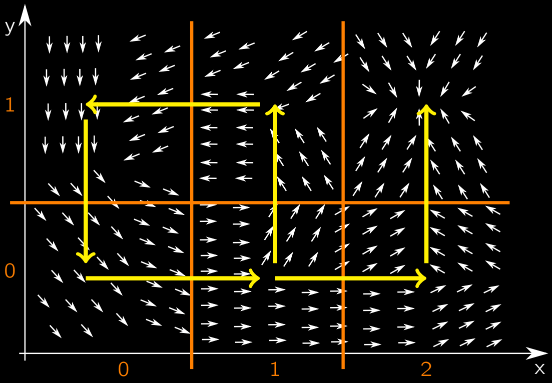
Loïc Paulevé

Outline

Outline

Qualitative models

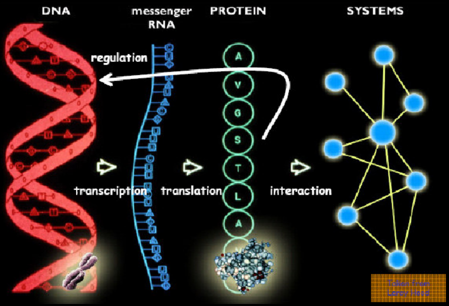
- Assume a quantization of the species population/concentration.
- Have a finite discrete state space (typically 2^n states).

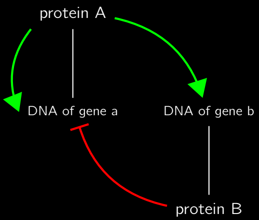


- We focus on Regulatory Networks...
- ...but the methods can be applied to any qualitative models.

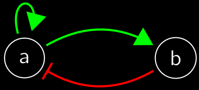
Biological Regulatory Networks (BRNs)

The Interaction Graph





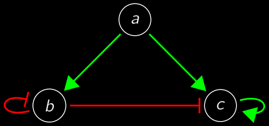
Interaction graph

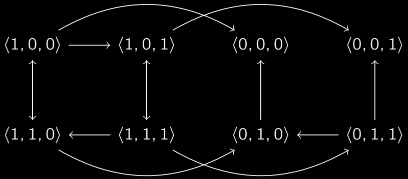


Qualitative Networks

- Each component has a finite set of qualitative levels ($\{0, 1, 2\}$).
- Functions associate the next level given the state of the regulators.

Boolean network example





$$f^a(a, b, c) = 0$$

$$f^b(a, b, c) = \begin{cases} 1 & \text{if } a = 1 \text{ and } b = 0 \\ 0 & \text{otherwise} \end{cases}$$

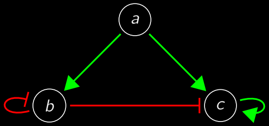
$$f^c(a, b, c) = \begin{cases} 1 & \text{if } b = 0 \text{ and } (a = 1 \text{ or } c = 1) \\ 0 & \text{otherwise} \end{cases};$$

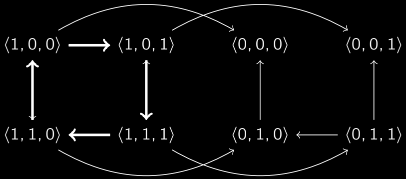
[René Thomas in Journal of Theoretical Biology, 1973]
[Richard, Comet, Bernot in Modern Formal Methods and App., 2006]

Qualitative Networks

- Each component has a finite set of qualitative levels ($\{0, 1, 2\}$).
- Functions associate the next level given the state of the regulators.

Boolean network example





$$f^a(a, b, c) = 0$$

$$f^b(a, b, c) = \begin{cases} 1 & \text{if } a = 1 \text{ and } b = 0 \\ 0 & \text{otherwise} \end{cases}$$

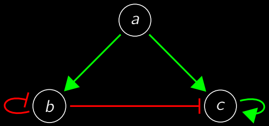
$$f^c(a, b, c) = \begin{cases} 1 & \text{if } b = 0 \text{ and } (a = 1 \text{ or } c = 1) \\ 0 & \text{otherwise} \end{cases};$$

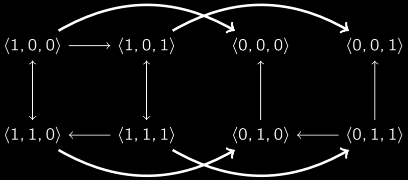
[René Thomas in Journal of Theoretical Biology, 1973]
[Richard, Comet, Bernot in Modern Formal Methods and App., 2006]

Qualitative Networks

- Each component has a finite set of qualitative levels ($\{0, 1, 2\}$).
- Functions associate the next level given the state of the regulators.

Boolean network example





$$f^a(a, b, c) = 0$$

$$f^b(a, b, c) = \begin{cases} 1 & \text{if } a = 1 \text{ and } b = 0 \\ 0 & \text{otherwise} \end{cases}$$

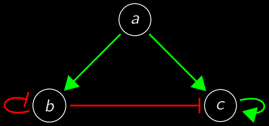
$$f^c(a, b, c) = \begin{cases} 1 & \text{if } b = 0 \text{ and } (a = 1 \text{ or } c = 1) \\ 0 & \text{otherwise} \end{cases};$$

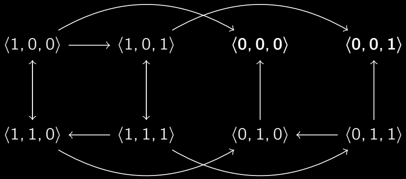
[René Thomas in Journal of Theoretical Biology, 1973]
[Richard, Comet, Bernot in Modern Formal Methods and App., 2006]

Qualitative Networks

- Each component has a finite set of qualitative levels ($\{0, 1, 2\}$).
- Functions associate the next level given the state of the regulators.

Boolean network example





$$f^a(a, b, c) = 0$$

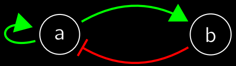
$$f^b(a, b, c) = \begin{cases} 1 & \text{if } a = 1 \text{ and } b = 0 \\ 0 & \text{otherwise} \end{cases}$$

$$f^c(a, b, c) = \begin{cases} 1 & \text{if } b = 0 \text{ and } (a = 1 \text{ or } c = 1) \\ 0 & \text{otherwise} \end{cases};$$

[René Thomas in Journal of Theoretical Biology, 1973]
[Richard, Comet, Bernot in Modern Formal Methods and App., 2006]

Dynamical properties from the Interaction Graph

An interaction graph can describe a large set of different dynamics.



Relationships between the interaction graph and dynamical properties:

- Multi-stationnarity requires a positive circuit (René Thomas conjecture) [Soule in ComPlexUs, 2003] [Richard, Comet in Discrete Appl. Math., 2007].
- Sustained oscillations require a negative circuit (René Thomas conjecture) [Remy, *et al.* in Adv. Appl. Math., 2008] [Richard in Adv. Appl. Math., 2010].
- The maximum number of fixed points can be characterized [Aracena in Bul. of Mathematical Biology, 2008]; [Richard in Discrete Appl. Math., 2009].
- Topological Fixed Points [Paulevé, Richard in CRAS 2010].
- etc.

(See [Paulevé, Richard at SASB'11] for a short survey).

Motivation and Challenges

Prove dynamical properties

- Fixed points (steady states) analysis;
- Reachability properties;
- Attractors characterisation.

Control dynamical properties

- Necessary or sufficient conditions.
- Key components/influences/parameters.

Motivation and Challenges

Prove dynamical properties

- Fixed points (steady states) analysis;
- Reachability properties;
- Attractors characterisation.

Control dynamical properties

- Necessary or sufficient conditions.
- Key components/influences/parameters.

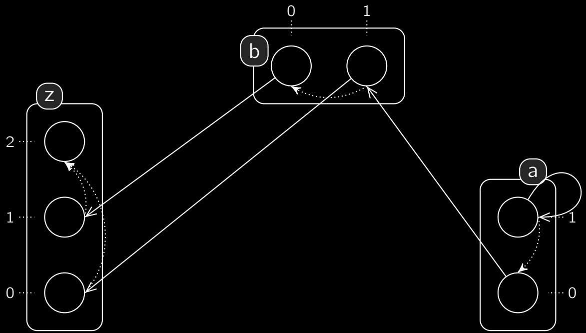
Large-scale models

- Lack of details (knowledge) for some interactions
  ⟶ avoid model enumeration.
- Numerous environment inputs: uncertainity for the initial conditions
  ⟶ handle multiple initial states at once.
- Work around the state-space combinatoric explosion
  ⟶ abstraction techniques.

Outline

The Process Hitting Framework

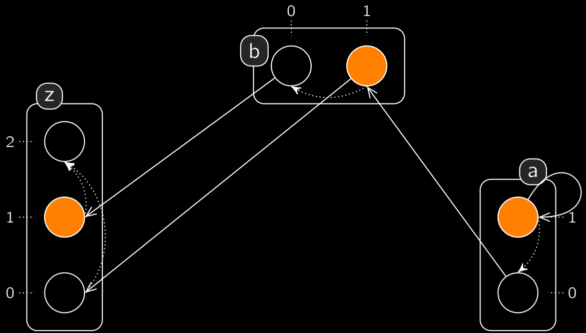
[Paulevé, Magnin, Roux in TCSB 2011]



- Sorts: a,b,z; Processes: $a_0, a_1, b_0, b_1, z_0, z_1, z_2$;
- Actions: a_0 *hits* b_1 to make it *bounce* to b_0, . . . ;
- States: $\langle a_1, b_1, z_1 \rangle$, $\langle a_0, b_1, z_1 \rangle$, $\langle a_0, b_0, z_1 \rangle$, . . . ;
- Restriction of Communicating Finite-State Machines (CFSM).

The Process Hitting Framework

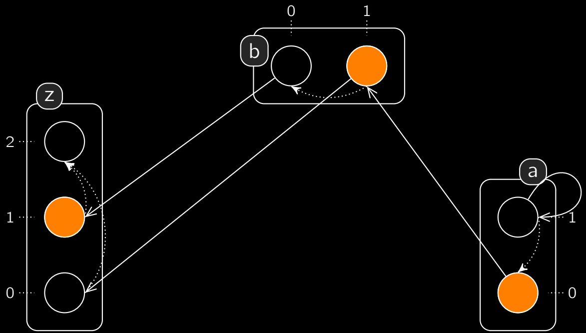
[Paulevé, Magnin, Roux in TCSB 2011]



- Sorts: a,b,z; Processes: $a_0, a_1, b_0, b_1, z_0, z_1, z_2$;
- Actions: a_0 *hits* b_1 to make it *bounce* to b_0, ...;
- States: $\langle a_1, b_1, z_1\rangle$, $\langle a_0, b_1, z_1\rangle$, $\langle a_0, b_0, z_1\rangle$, ...;
- Restriction of Communicating Finite-State Machines (CFSM).

The Process Hitting Framework

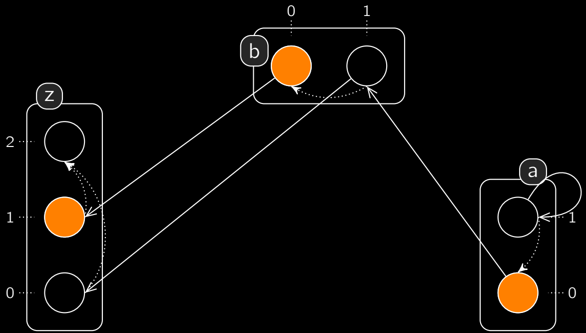
[Paulevé, Magnin, Roux in TCSB 2011]



- Sorts: a,b,z; Processes: $a_0, a_1, b_0, b_1, z_0, z_1, z_2$;
- Actions: a_0 *hits* b_1 to make it *bounce* to b_0, ...;
- States: $\langle a_1, b_1, z_1 \rangle$, $\langle a_0, b_1, z_1 \rangle$, $\langle a_0, b_0, z_1 \rangle$, ...;
- Restriction of Communicating Finite-State Machines (CFSM).

The Process Hitting Framework

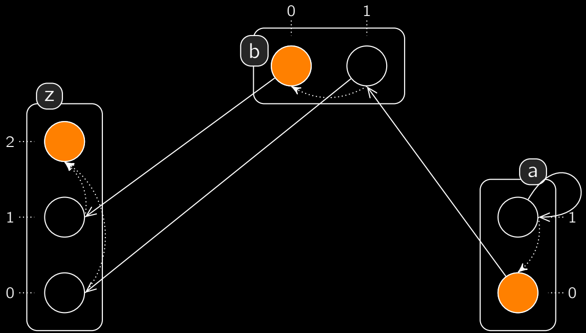
[Paulevé, Magnin, Roux in TCSB 2011]



- Sorts: a,b,z; Processes: $a_0, a_1, b_0, b_1, z_0, z_1, z_2$;
- Actions: a_0 *hits* b_1 to make it *bounce* to b_0, . . . ;
- States: $\langle a_1, b_1, z_1\rangle$, $\langle a_0, b_1, z_1\rangle$, $\langle a_0, b_0, z_1\rangle$, . . . ;
- Restriction of Communicating Finite-State Machines (CFSM).

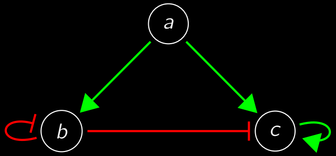
The Process Hitting Framework

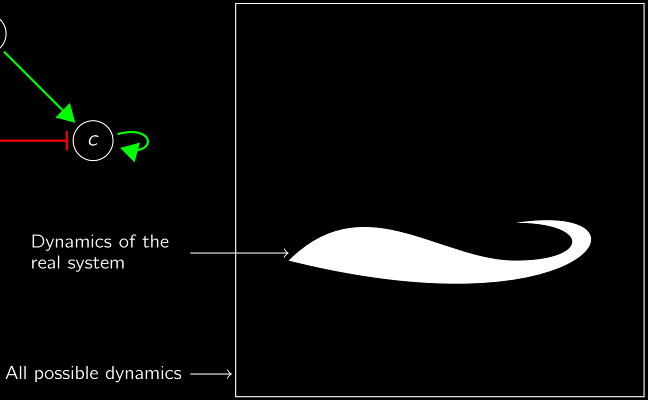
[Paulevé, Magnin, Roux in TCSB 2011]



- Sorts: a,b,z; Processes: $a_0, a_1, b_0, b_1, z_0, z_1, z_2$;
- Actions: a_0 *hits* b_1 to make it *bounce* to b_0, . . . ;
- States: $\langle a_1, b_1, z_1\rangle$, $\langle a_0, b_1, z_1\rangle$, $\langle a_0, b_0, z_1\rangle$, . . . ;
- Restriction of Communicating Finite-State Machines (CFSM).

Generalised Dynamics of Regulatory Networks

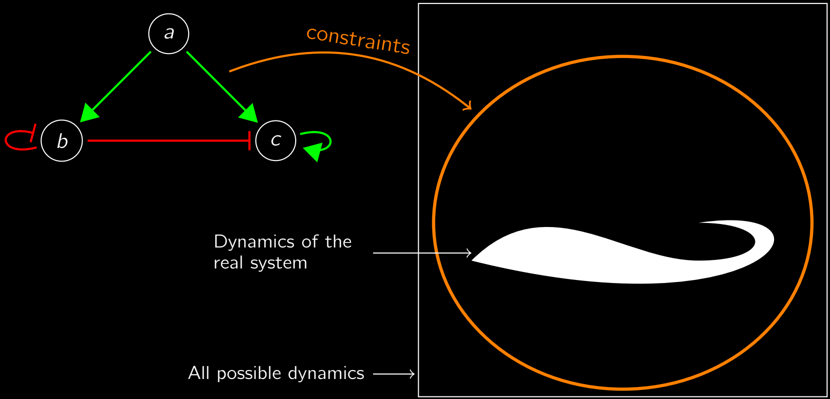




Dynamics over-approximation

- A component can not increase if none effective activator is present.
- A component can not decrease if none effective inhibitor is present.

Generalised Dynamics of Regulatory Networks



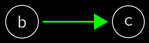
Dynamics over-approximation

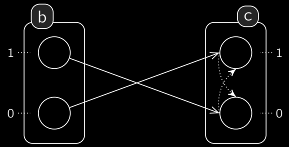
- A component can not increase if none effective activator is present.
- A component can not decrease if none effective inhibitor is present.

Generalized Dynamics of BRNs

- Idea: the most permissive dynamics [Paulevé, Magnin, Roux in TCSB 2011].
- Without knowledge of functions between components.

Boolean case:

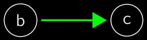


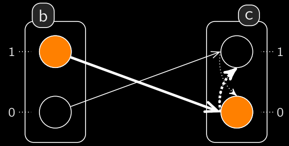


Generalized Dynamics of BRNs

- Idea: the most permissive dynamics [Paulevé, Magnin, Roux in TCSB 2011].
- Without knowledge of functions between components.

Boolean case:

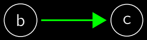


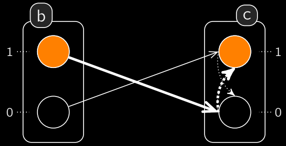


Generalized Dynamics of BRNs

- Idea: the most permissive dynamics [Paulevé, Magnin, Roux in TCSB 2011].
- Without knowledge of functions between components.

Boolean case:

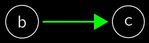


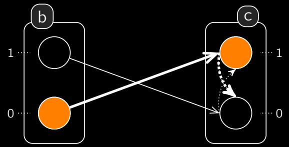


Generalized Dynamics of BRNs

- Idea: the most permissive dynamics [Paulevé, Magnin, Roux in TCSB 2011].
- Without knowledge of functions between components.

Boolean case:

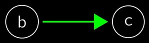


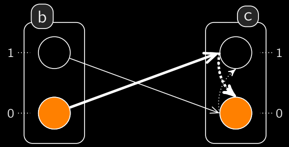


Generalized Dynamics of BRNs

- Idea: the most permissive dynamics [Paulevé, Magnin, Roux in TCSB 2011].
- Without knowledge of functions between components.

Boolean case:

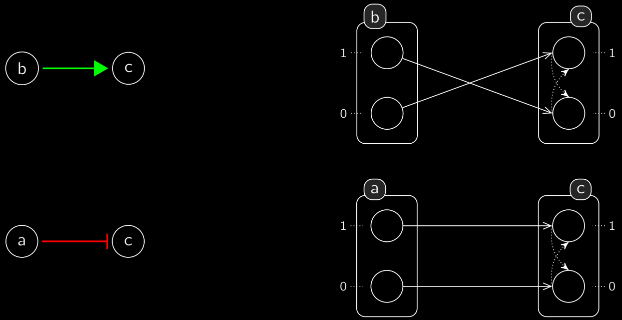




Generalized Dynamics of BRNs

- Idea: the most permissive dynamics [Paulevé, Magnin, Roux in TCSB 2011].
- Without knowledge of functions between components.

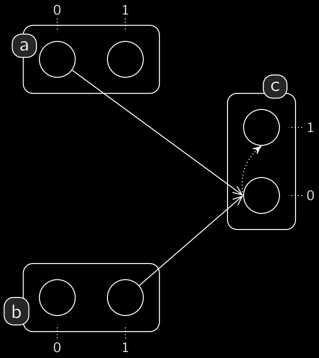
Boolean case:



Note: this construction can be easily extended to multi-valued components.

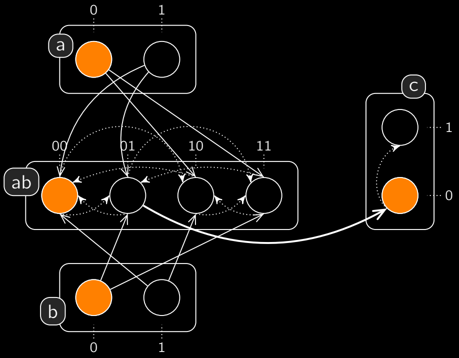
Refining with Cooperation

- Idea: $c_0 \rightarrow c_1$ when a_0 and b_1 are present.
- Introduction of a cooperative sort reflecting the state of the sorts a and b.



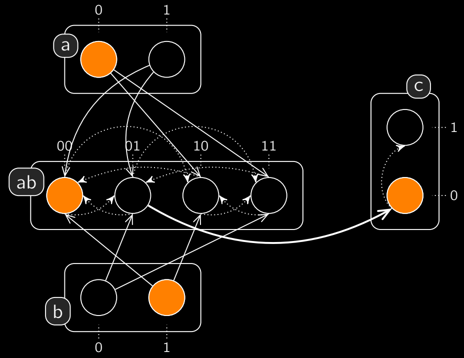
Refining with Cooperation

- Idea: $c_0 \rightarrow c_1$ when a_0 and b_1 are present.
- Introduction of a cooperative sort reflecting the state of the sorts a and b.



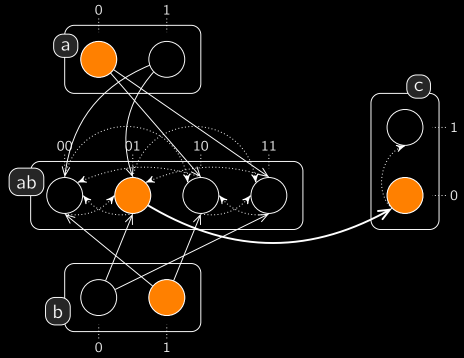
Refining with Cooperation

- Idea: $c_0 \rightarrow c_1$ when a_0 and b_1 are present.
- Introduction of a cooperative sort reflecting the state of the sorts a and b.



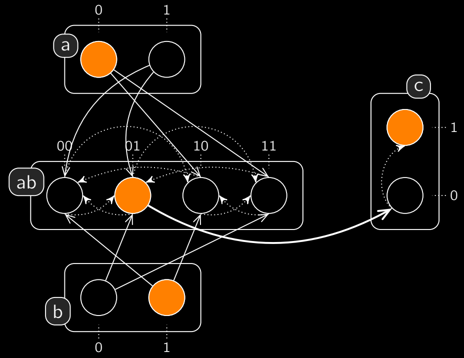
Refining with Cooperation

- Idea: $c_0 \rightarrow c_1$ when a_0 and b_1 are present.
- Introduction of a cooperative sort reflecting the state of the sorts a and b.



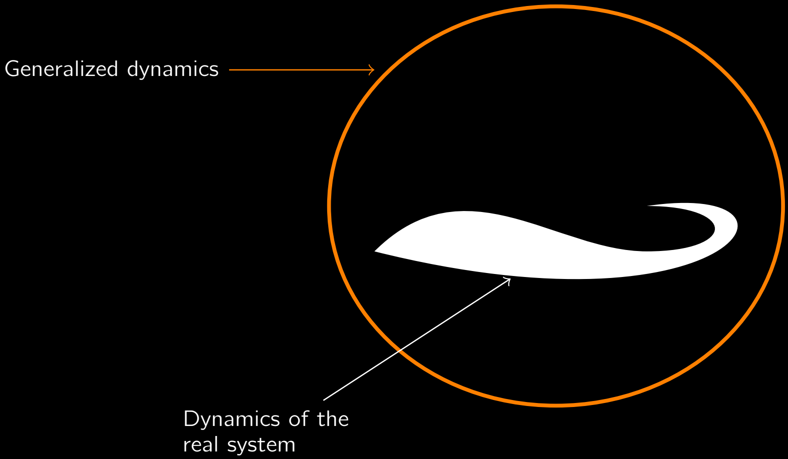
Refining with Cooperation

- Idea: $c_0 \upharpoonright c_1$ when a_0 and b_1 are present.
- Introduction of a cooperative sort reflecting the state of the sorts a and b.

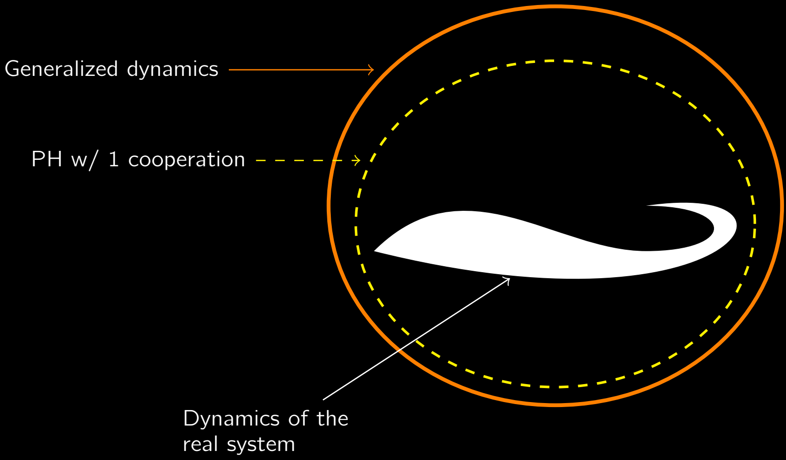


$\Rightarrow$ introduce a temporal shift; similar to complexes.

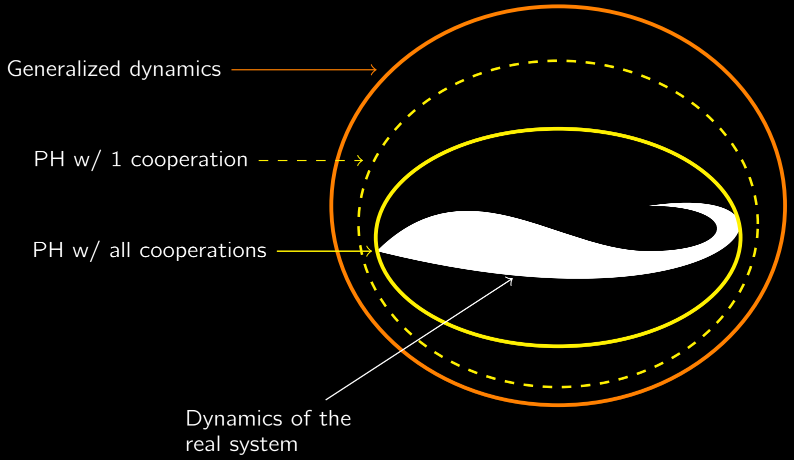
Abstraction Relationships



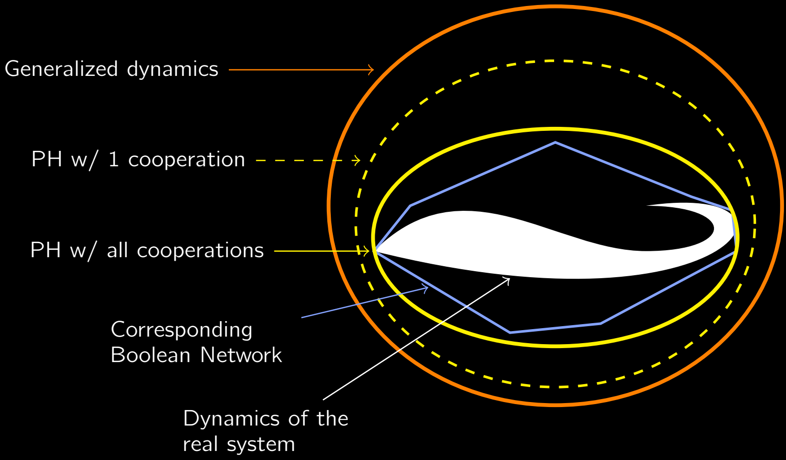
Abstraction Relationships



Abstraction Relationships

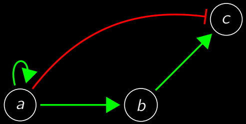


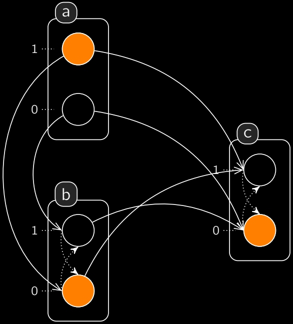
Abstraction Relationships

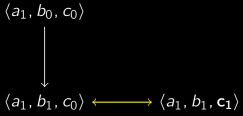


Toy example

Incoherent feed-forward loop

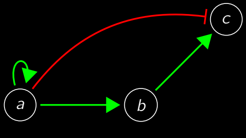


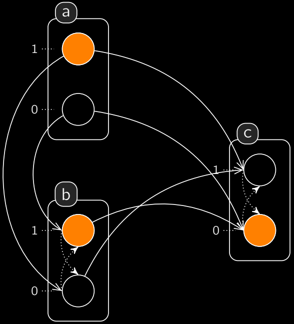




Toy example

Incoherent feed-forward loop





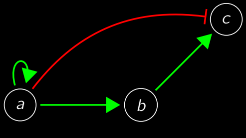
$\langle a_1, b_0, c_0 \rangle$

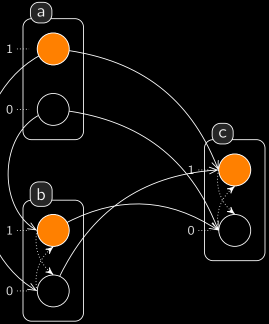
↓

$\langle a_1, b_1, c_0 \rangle \longleftrightarrow \langle a_1, b_1, \mathbf{c_1} \rangle$

Toy example

Incoherent feed-forward loop





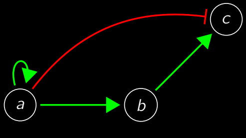
$\langle a_1, b_0, c_0\rangle$

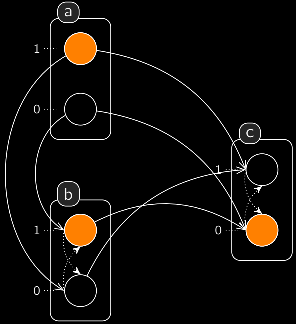
$\downarrow$

$\langle a_1, b_1, c_0\rangle \longleftrightarrow \langle a_1, b_1, \mathbf{c_1}\rangle$

Toy example

Incoherent feed-forward loop





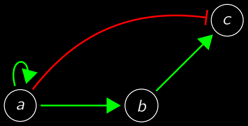
$\langle a_1, b_0, c_0 \rangle$

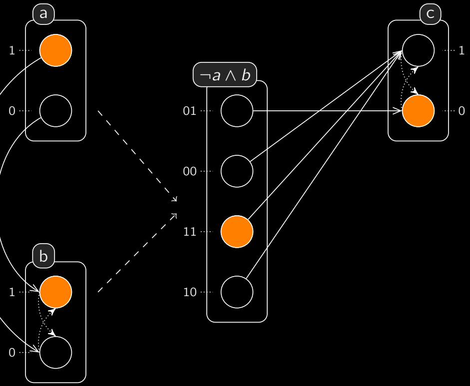
$\downarrow$

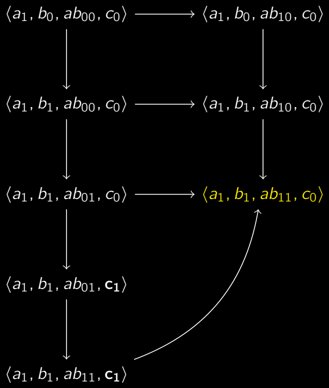
$\langle a_1, b_1, c_0 \rangle \longleftrightarrow \langle a_1, b_1, \mathbf{c_1} \rangle$

Toy example

Incoherent feed-forward loop







Other work around the Process Hitting

Stochastic and time dimension

[Paulevé et al. in TCSB 2011] [Paulevé, PhD thesis]

- Markovian and non-Markovian stochastic semantics.
- Simulation, probabilistic model-checking.

Process Hitting to Boolean Networks

[Folschette, Paulevé, Inoue, Magnin, Roux at CMSB'12]

- Inference of the Interaction Graph from a Process Hitting.
- A Process Hitting can abstract at once different Boolean Networks.

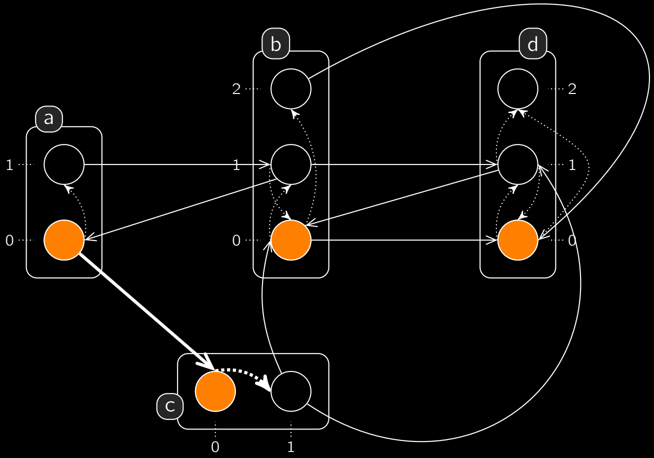
Static analysis of fixed points (steady states)

[Paulevé et al. in TCSB 2011]

- Reduction to the search for N-cliques in N-partite graphs.
- Efficient enumeration.

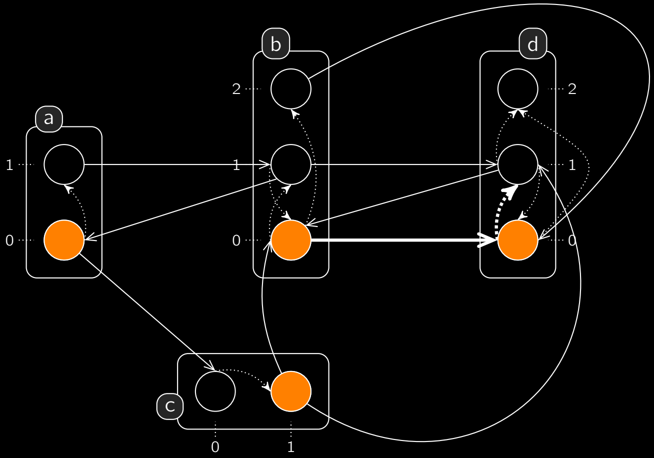
Outline

Scenarios



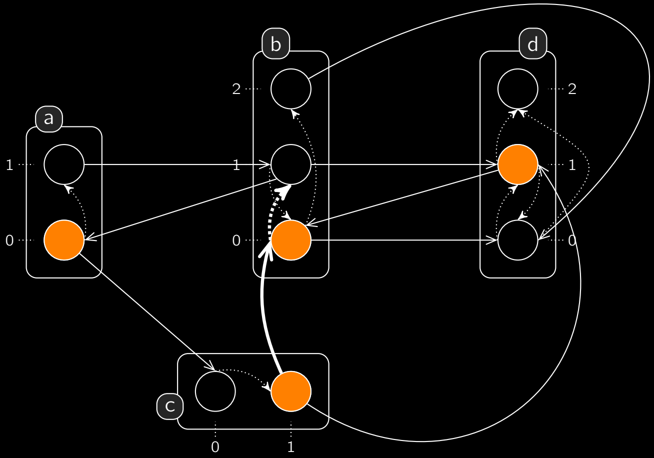
$a_0 \rightarrow c_0 \upharpoonright c_1 :: b_0 \rightarrow d_0 \upharpoonright d_1 :: c_1 \rightarrow b_0 \upharpoonright b_1 :: b_1 \rightarrow d_1 \upharpoonright d_2$

Scenarios



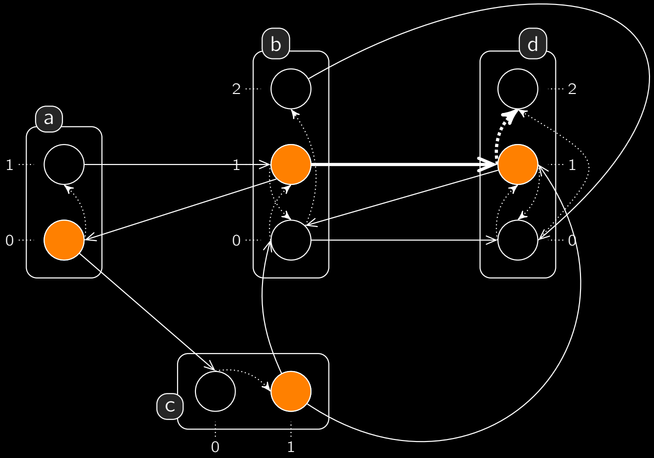
$a_0 \rightarrow c_0 \upharpoonright c_1 :: b_0 \rightarrow d_0 \upharpoonright d_1 :: c_1 \rightarrow b_0 \upharpoonright b_1 :: b_1 \rightarrow d_1 \upharpoonright d_2$

Scenarios



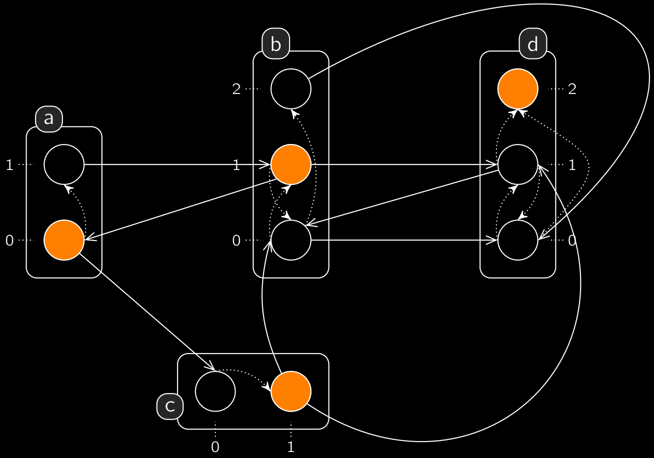
$a_0 \rightarrow c_0 \upharpoonright c_1 :: b_0 \rightarrow d_0 \upharpoonright d_1 :: c_1 \rightarrow b_0 \upharpoonright b_1 :: b_1 \rightarrow d_1 \upharpoonright d_2$

Scenarios



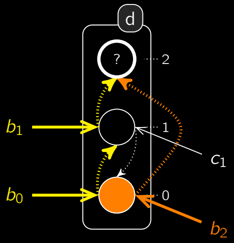
$a_0 \rightarrow c_0 \upharpoonright c_1 :: b_0 \rightarrow d_0 \upharpoonright d_1 :: c_1 \rightarrow b_0 \upharpoonright b_1 :: b_1 \rightarrow d_1 \upharpoonright d_2$

Scenarios



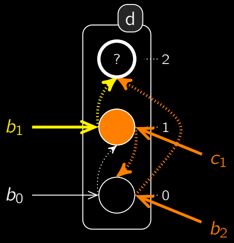
$$a_0 \rightarrow c_0 \Rsh c_1 :: b_0 \rightarrow d_0 \Rsh d_1 :: c_1 \rightarrow b_0 \Rsh b_1 :: b_1 \rightarrow d_1 \Rsh d_2$$

Local Causality

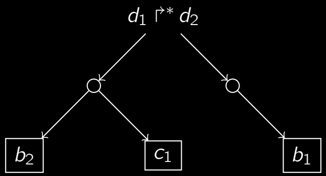


- $\mathrm{sol}(d_0 \Rsh^* d_2) = \{ b_0 \rightarrow d_0 \Rsh d_1 :: b_1 \rightarrow d_1 \Rsh d_2, \; b_2 \rightarrow d_0 \Rsh d_2 \}$;
- $\mathrm{sol}^{\wedge}(d_0 \Rsh^* d_2) = \{\{b_0, b_1\}, \{b_2\}\}$.

Local Causality

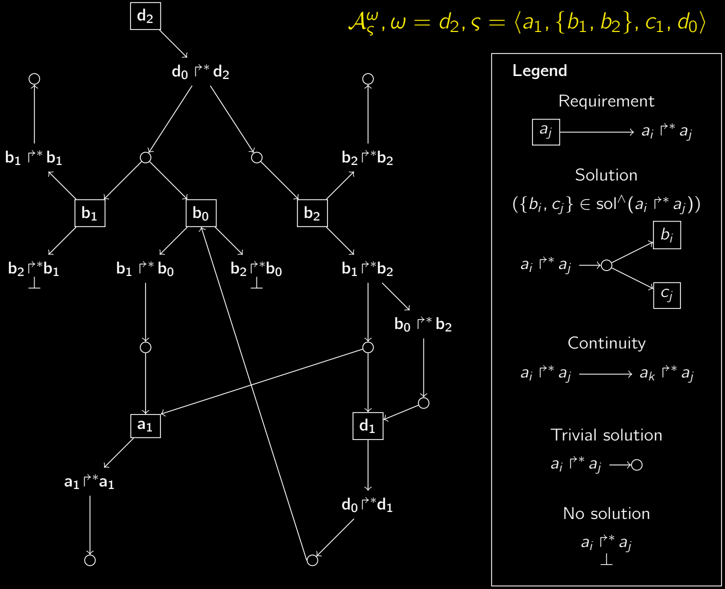


- $\mathrm{sol}(d_0 \Rsh^* d_2) = \{b_0 \to d_0 \Rsh d_1 :: b_1 \to d_1 \Rsh d_2,$
  $b_2 \to d_0 \Rsh d_2\};$
- $\mathrm{sol}^\wedge(d_0 \Rsh^* d_2) = \{\{b_0, b_1\}, \{b_2\}\}.$
- $\mathrm{sol}(d_1 \Rsh^* d_2) = \{b_1 \to d_1 \Rsh d_2,$
  $c_1 \to d_1 \Rsh d_0 :: b_2 \to d_0 \Rsh d_2\};$
- $\mathrm{sol}^\wedge(d_1 \Rsh^* d_2) = \{\{b_1\}, \{b_2, c_1\}\}.$



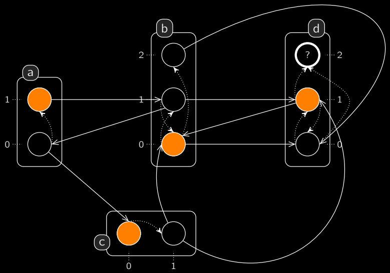
Graph of Local Causality

$\mathcal{A}^{\omega}_{\varsigma}, \omega = d_2, \varsigma = \langle a_1, \{b_1, b_2\}, c_1, d_0 \rangle$



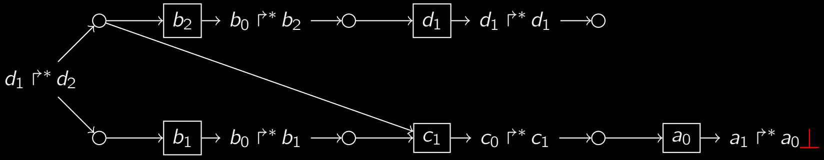
Un-ordered Over-approximation

Example



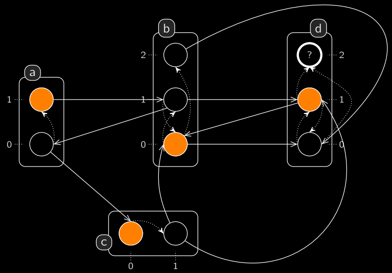
Necessary condition for reaching d_2:
There exists a traversal of $\mathcal{A}_\varsigma^\omega$ such that:

- objective → follow at least one solution;
- process → follow all objectives;
- no cycle.



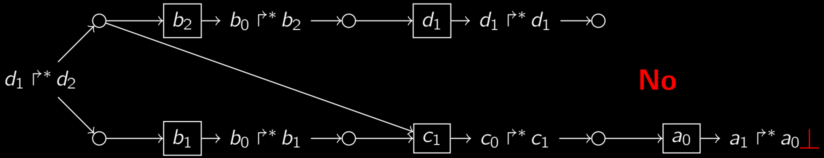
Un-ordered Over-approximation

Example



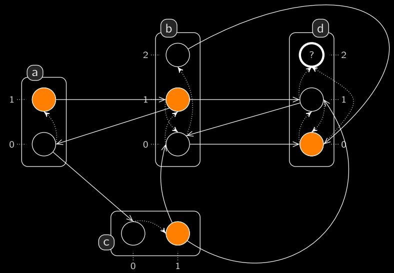
Necessary condition for reaching d_2:
There exists a traversal of $\mathcal{A}_\varsigma^\omega$ such that:

- objective $\rightarrow$ follow at least one solution;
- process $\rightarrow$ follow all objectives;
- no cycle.



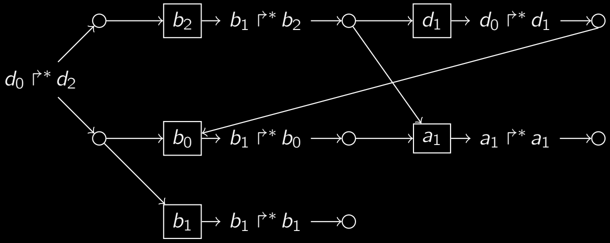
Un-ordered Over-approximation

Example



Necessary condition for reaching d_2:
There exists a traversal of $\mathcal{A}^{\omega}_{\varsigma}$ such that:

- objective → follow at least one solution;
- process → follow all objectives;
- no cycle.



Inconc

Static Analysis of Successive Reachability

Over-approximations

- Un-ordered approximation.
- Ordered approximation.
- Ordered approximation with occurences order constraints.

No / Inconc

⇑

Successive Process Reachability (reach a_i, then b_k, etc.)

⇑

Under-approximations

- Un-ordered approximation.
- Ordered approximation.

Yes / Inconc

Static Analysis of Successive Reachability

Over-approximations

- Un-ordered approximation.
- Ordered approximation.
- Ordered approximation with occurences order constraints.

No / Inconc

⇑

Successive Process Reachability (reach a_i, then b_k, etc.)

⇑

Under-approximations

- Un-ordered approximation.
- Ordered approximation.

Yes / Inconc

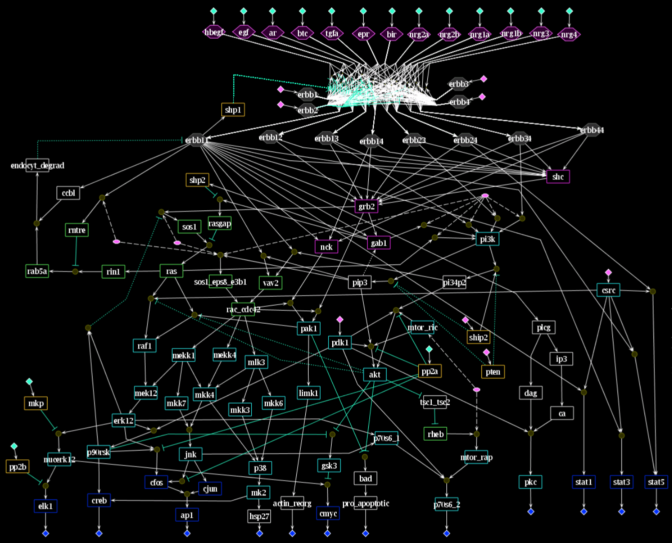
Complexity

$\Longrightarrow$ efficient with a small number of processes per sort, while a very large number of sorts can be handled.

[Paulevé, Magnin, Roux in Mathematical Structures in Computer Science, 2012]

EGFR/ErbB Signalling Network

(104 components)



[Samaga, *et al.* in PLoS Comput Biol, 2009]

Process Hitting

193 sorts,
748 processes,
2356 actions:
$\approx 2 \cdot 10^{96}$ states.

Execution times

- Real biological models.
- Wide-range of biological/arbitrary reachability analysis.
- Always conclusive.

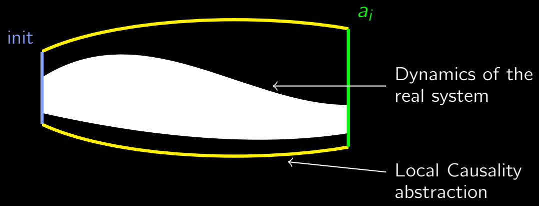
Model	sorts	procs	actions	states	Biocham[1]	libDDD[2]	PINT[3]
egfr20	35	196	670	2^{64}	[3s-KO]	[1s-150s]	0.007s
tcrsig40	54	156	301	2^{73}	[1s-KO]	[0.6s-KO]	0.004s
tcrsig94	133	448	1124	2^{194}	KO	KO	0.030s
egfr104	193	748	2356	2^{320}	KO	KO	0.050s

[1] `http://contraintes.inria.fr/biocham` (using NuSMV2)
[2] `http://move.lip6.fr/software/DDD`
[3] `http://process.hitting.free.fr`

Necessary Sets of Processes for Reachability

Settings

- reachability of a_i (level i of component a);
- from partially-determined initial condition (set of initial states).



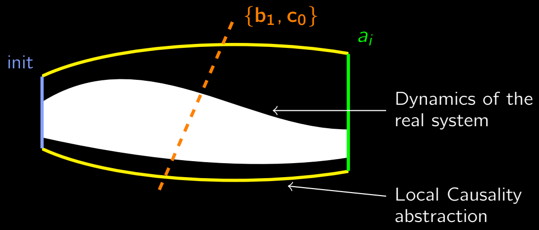
- All traces use, at one point, at least one process of a cut set.
- Disabling all processes of a cut set should prevent reachability in the real system.
- Otherwise, the model is not an over-approximation.

We restrict ourselves to necessary N-sets of processes.

Necessary Sets of Processes for Reachability

Settings

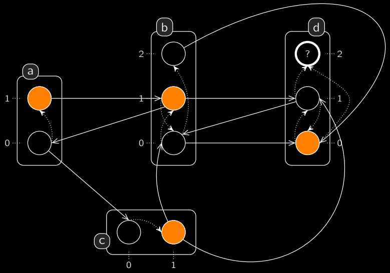
- reachability of a_i (level i of component a);
- from partially-determined initial condition (set of initial states).



- All traces use, at one point, at least one process of a cut set.
- Disabling all processes of a cut set should prevent reachability in the real system.
- Otherwise, the model is not an over-approximation.

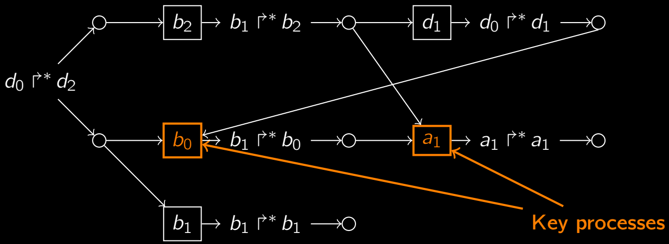
We restrict ourselves to necessary N-sets of processes.

Extraction of Key Processes



Necessary condition for reaching d_2:
There exists a traversal of $\mathcal{A}^{\omega}_{\varsigma}$ such that:

- objective → follow at least one solution;
- process → follow all objectives;
- no cycle.



Formal analysis of the whole PID

Pathway Interaction Database

- Inductions, inhibitions, transcriptional regulation, complex formations, . . .
- More than 9000 interacting components.
- Large environment (3000 entry-points).

Graph of Local Causality

- From Process Hitting model (boolean interpretation).
- (Independent) reachability of active SNAIL, p15INK4b, p21CIP1.
- 20 000 nodes, including 5600 processes (biological or cooperative).

N-sets of necessary biological processes

N	Exec. time	$SNAIL_1$	$p15INK4b_1$	$p21CIP1_1$
1	0.9s	1	1	1
2	1.6s	+6	+6	+0
3	5.4s	+0	+92	+0
4	39s	+30	+60	+0
5	8.3m	+90	+80	+0
6	2.6h	+930	+208	+0

Outline

Conclusion

The Process Hitting framework

- Qualitative asynchronous modelling.
- Different levels of dynamics abstractions (partial knowledge on cooperations).
- Automatic encoding of Boolean Networks (over-approximation).

Abstract causality analysis

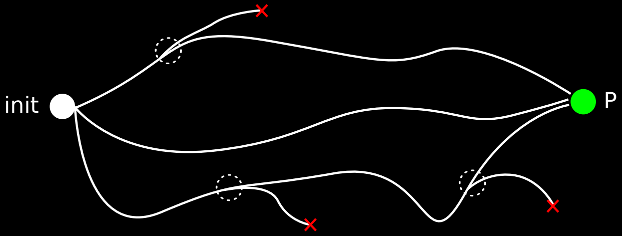
- Local causality reasonning.
- Over- and under-approximation of reachability properties.
- Extract necessary sets of processes (potential therapeutic targets).
- Tractable on very large networks.

Implementation: PINT software - `http://process.hitting.free.fr`

Future work

Process Hitting with Priorities

- Static split of actions into priority classes.
- An action can be played only if none action with higher priority can be played.
- $\Longrightarrow$ different time-scales;
- $\Longrightarrow$ enhanced expressivity (with 3 classes: Petri Nets).



Link with continuous and stochastic models

Acknowledgement

IRCCyN, Nantes MeForBio

- **Olivier Roux** (PhD supervisor)
- **Morgan Magnin** (PhD co-supervisor)
- Maxime Folschette

IRISA, Rennes Dyliss

- Geoffroy Andrieux (PID model)

ETH Zürich BISON

- Heinz Koeppl

ANR BioTempo.