Efficient Static Analysis of Dynamical Properties using the Process Hitting

10 janvier 2012

Loïc Paulevé

École Polytechnique / LIX (équipe AMIB)
pauleve@lix.polytechnique.fr
http://loicpauleve.name

Work with Morgan Magnin and Olivier (F.) Roux (PhD thesis)

Overview

Context

- Computer science for systems biology.
- Abstract (discrete) modelling.

Problematics

- Scalable formal verification.
- Properties of interest:
  - Fixpoint enumeration.
  - Reachability properties.
  - Control (drug targets).

Overview

Context

- Computer science for systems biology.
- Abstract (discrete) modelling.

Problematics

- Scalable formal verification.
- Properties of interest:
  - Fixpoint enumeration.
  - Reachability properties.
  - Control (drug targets).

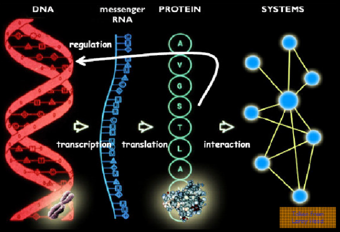
The Process Hitting [Paulevé, Magnin, Roux in TCSB 2011]

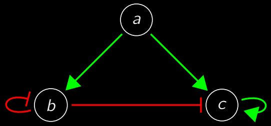
- Subclass of Communicating Finite State Machines / Petri Nets / etc.
- Suitable to model Biological Regulatory Networks.
- Efficient static analysis of dynamical properties.
  [Paulevé, Magnin, Roux in MSCS 2012]

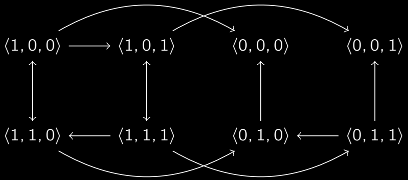
Outline

Outline

Biological Regulatory Networks







$f^a(x) = 0$

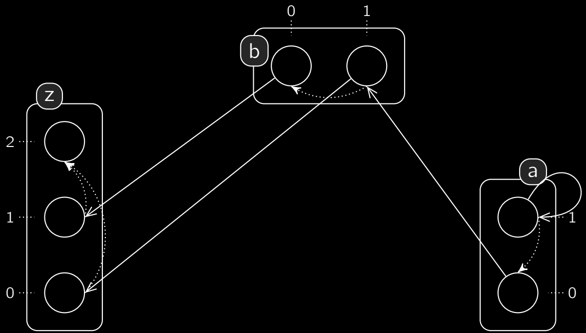
$f^b(x) = x[a] \wedge \neg x[b]$

$f^c(x) = \neg x[b] \wedge (x[a] \vee x[c])$

[René Thomas in Journal of Theoretical Biology, 1973]

The Process Hitting Framework

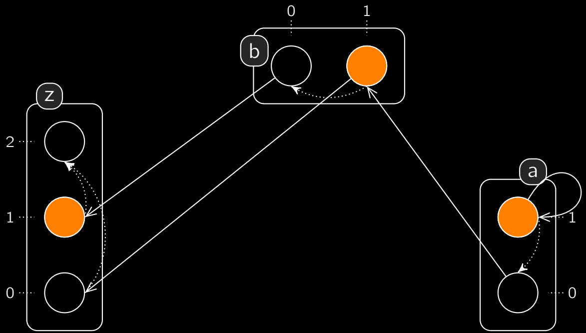
[Paulevé, Magnin, Roux in TCSB 2011]



- 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

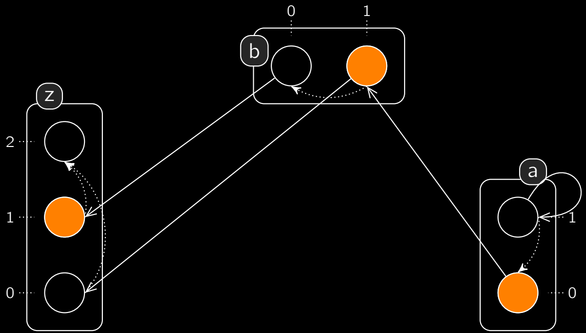
[Paulevé, Magnin, Roux in TCSB 2011]



- 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

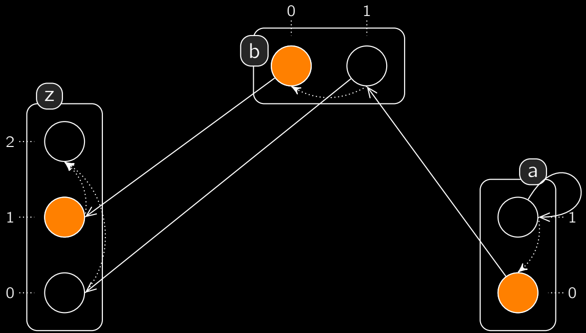
[Paulevé, Magnin, Roux in TCSB 2011]



- 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

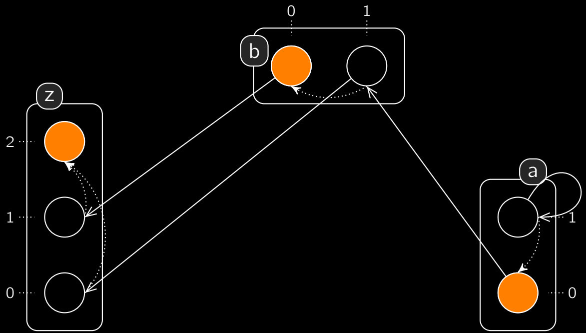
[Paulevé, Magnin, Roux in TCSB 2011]



- 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).

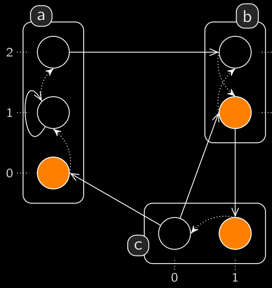
The Process Hitting Framework

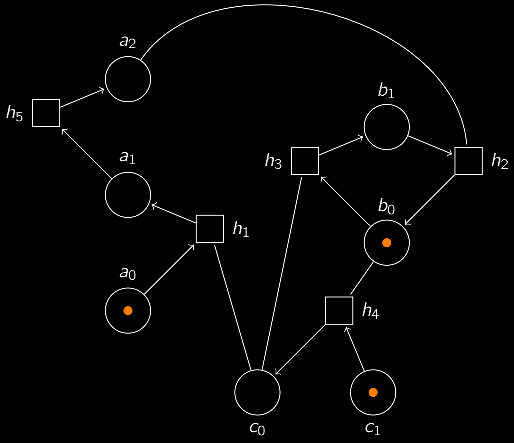
[Paulevé, Magnin, Roux in TCSB 2011]



- 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).

Subclass of Petri Nets





Outline

Static Analysis of Process Hittings

Intuition

- Simplicity of the Process Hitting ⇒ models with simple structures.
- Efficient static derivation of dynamical properties.

Static Analysis of Process Hittings

Intuition

- Simplicity of the Process Hitting ⇒ models with simple structures.
- Efficient static derivation of dynamical properties.

Fixed Points

- Reduction to the n-cliques of an n-partite graph.

Static Analysis of Process Hittings

Intuition

- Simplicity of the Process Hitting $\Rightarrow$ models with simple structures.
- Efficient static derivation of dynamical properties.

Fixed Points

- Reduction to the n-cliques of an n-partite graph.

Successive reachability properties $\mathrm{EF}\ a_i \wedge (\mathrm{EF}\ b_j \wedge \dots)$

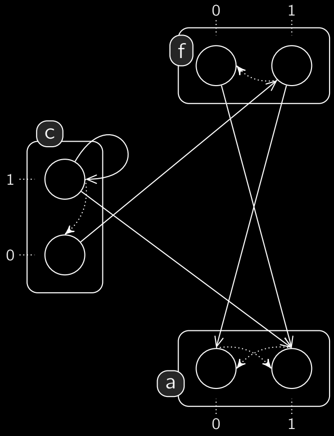
- Limited complexity but may be inconclusive (Yes/No/Inconc).
- Abstract interpretation techniques.
- Extraction of key processes (towards control).

Outline

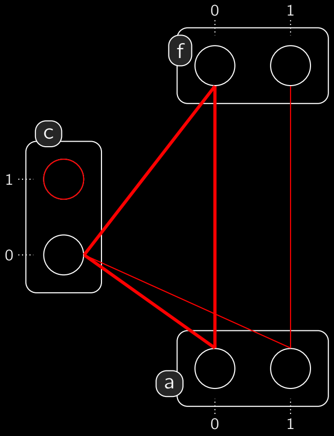
Fixed Points

[Paulevé, Magnin, Roux in TCSB 2011]

Process Hitting



Hitless graph

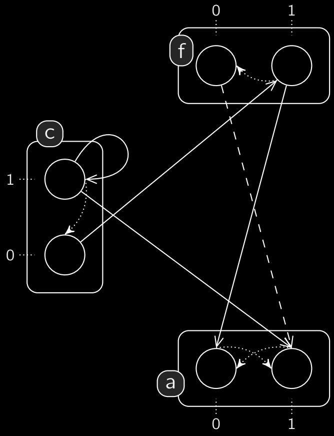


n-cliques are fixed points

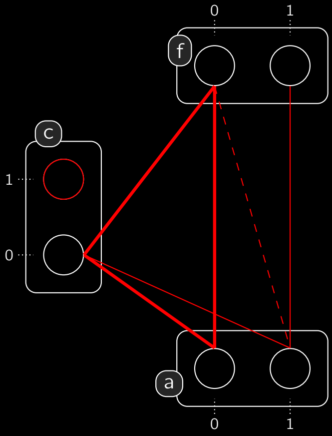
Fixed Points

[Paulevé, Magnin, Roux in TCSB 2011]

Process Hitting



Hitless graph

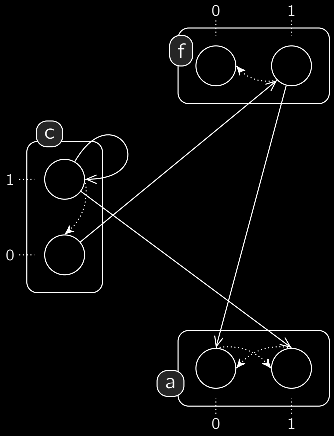


n-cliques <u>are</u> fixed points

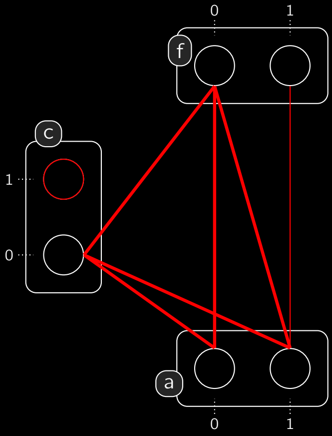
Fixed Points

[Paulevé, Magnin, Roux in TCSB 2011]

Process Hitting



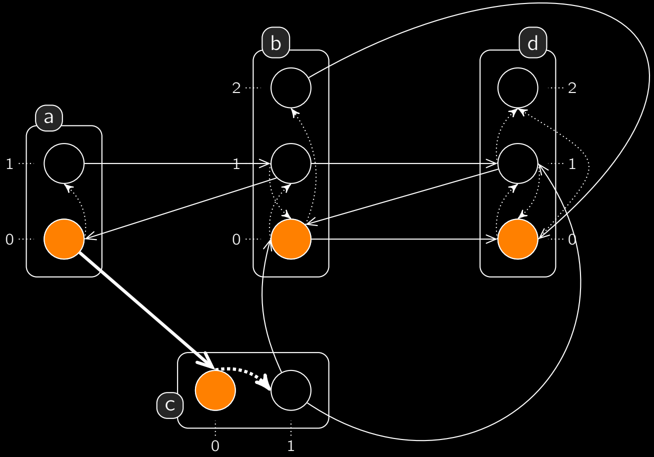
Hitless graph



n-cliques <u>are</u> fixed points

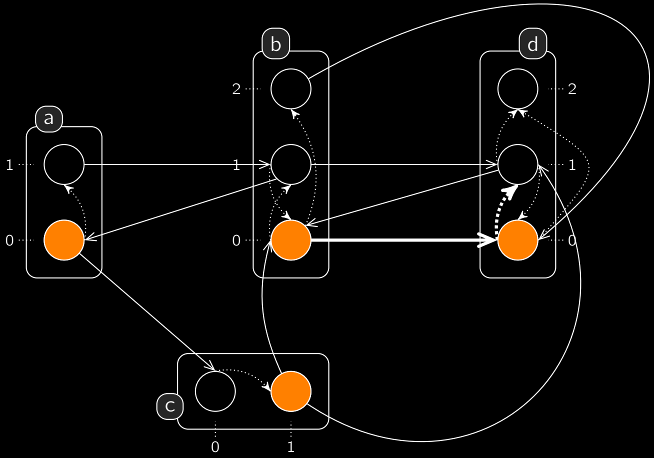
Outline

Scenarios



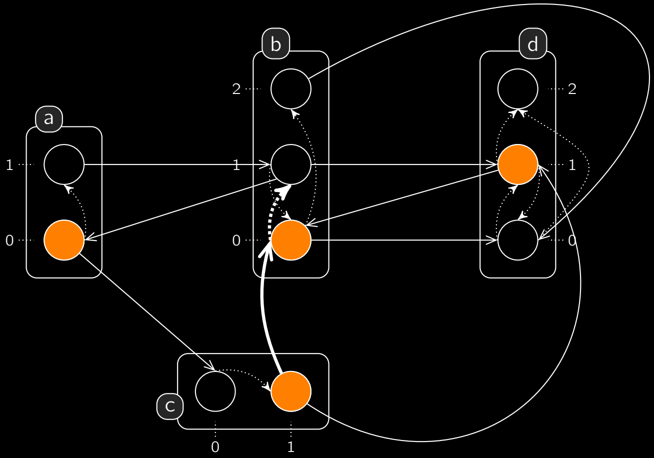
$$a_0 \rightarrow c_0 \Rsh c_1 :: b_0 \rightarrow d_0 \Rsh d_1 :: c_1 \rightarrow b_0 \Rsh b_1 :: b_1 \rightarrow d_1 \Rsh d_2$$

Scenarios



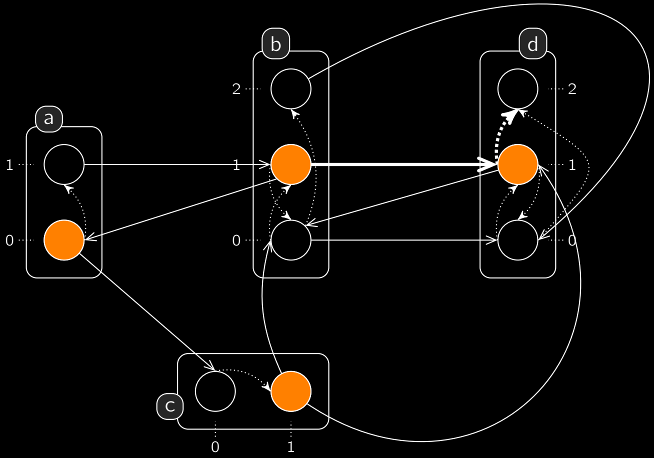
$a_0 \rightarrow c_0 \upharpoonright c_1 :: b_0 \rightarrow d_0 \upharpoonright d_1 :: c_1 \rightarrow b_0 \upharpoonright b_1 :: b_1 \rightarrow d_1 \upharpoonright d_2$

Scenarios



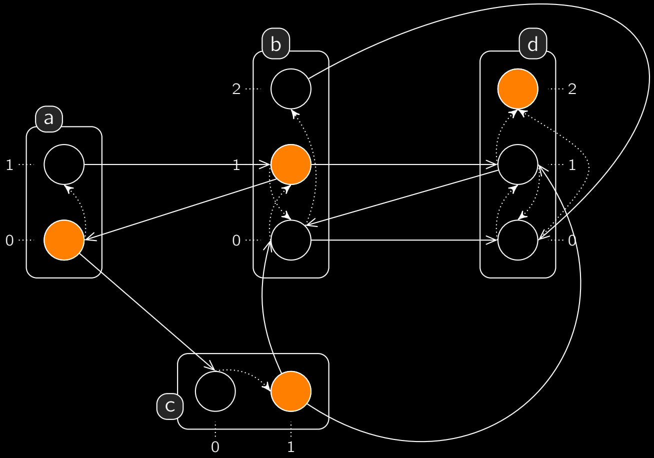
$a_0 \rightarrow c_0 \Rsh c_1 :: b_0 \rightarrow d_0 \Rsh d_1 :: c_1 \rightarrow b_0 \Rsh b_1 :: b_1 \rightarrow d_1 \Rsh d_2$

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



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

Static Analysis of Successive Reachability Properties

[Paulevé, Magnin, Roux in MSCS 2012]

Successive Reachability $\mathcal{R}$

- Given a Process Hitting $\mathcal{PH}$ with an initial state,
- is it possible to reach the process a_i? ...
- then the process b_j? ... etc.

Static Analysis of Successive Reachability Properties

[Paulevé, Magnin, Roux in MSCS 2012]

Successive Reachability $\mathcal{R}$

- Given a Process Hitting $\mathcal{PH}$ with an initial state,
- is it possible to reach the process a_i? ...
- then the process b_j? ... etc.

Difficulties: combinatorial explosion of dynamics to explore.

Static Analysis of Successive Reachability Properties

[Paulevé, Magnin, Roux in MSCS 2012]

Successive Reachability $\mathcal{R}$

- Given a Process Hitting $\mathcal{PH}$ with an initial state,
- is it possible to reach the process a_i? ...
- then the process b_j? ... etc.

Difficulties: combinatorial explosion of dynamics to explore.

Chosen approach

Over-approximations
$\mathcal{PH}$ does not satisfy $\mathcal{P} \Longrightarrow \mathcal{R}$ is impossible.

Under-approximations
$\mathcal{PH}$ satisfies $\mathcal{Q} \Longrightarrow \mathcal{R}$ is possible.

Requirement: checking $\mathcal{P}$ ($\mathcal{Q}$) is fast.

Abstract Interpretation of Scenarios

Scenarios – Successively playable actions.

- E.g. $\delta = a_0 \rightarrow c_0 \upharpoonright c_1 :: b_0 \rightarrow d_0 \upharpoonright d_1 :: c_1 \rightarrow b_0 \upharpoonright b_1 :: b_1 \rightarrow d_1 \upharpoonright d_2$.

Context — For each sort, subset of initial processes.

- E.g. $\varsigma = \langle a_0, \{b_0, b_2\}, c_0, d_0 \rangle$.

Abstract Interpretation of Scenarios

Scenarios – Successively playable actions.

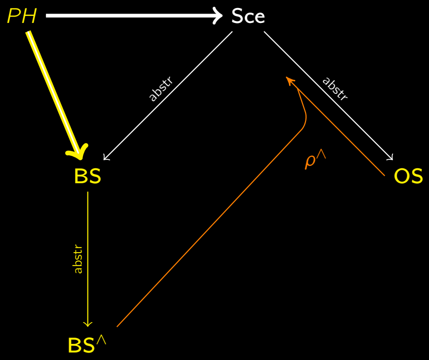
- E.g. $\delta = a_0 \rightarrow c_0 \upharpoonright c_1 :: b_0 \rightarrow d_0 \upharpoonright d_1 :: c_1 \rightarrow b_0 \upharpoonright b_1 :: b_1 \rightarrow d_1 \upharpoonright d_2$.

Context — For each sort, subset of initial processes.

- E.g. $\varsigma = \langle a_0, \{b_0, b_2\}, c_0, d_0 \rangle$.

Overall approach

- 2 orthogonal abstractions;
- Bounce Sequences **BS**;
- Objective Sequences **OS**;
- Concretization: $\gamma_\varsigma : \mathbf{OS} \mapsto \wp(\mathbf{Sce})$;
- Refinements: $\rho : \mathbf{OS} \mapsto \wp(\mathbf{OS})$;
- $\gamma_\varsigma(\omega) = \gamma_\varsigma(\rho(\omega))$.



Two Orthogonal Abstractions

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

Abstraction by Objective Sequences

- $c_0 \Rsh^* c_1 :: d_0 \Rsh^* d_1 :: b_0 \Rsh^* b_1 :: d_1 \Rsh^* d_2$;

Two Orthogonal Abstractions

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

Abstraction by Objective Sequences

- $c_0 \upharpoonright^* c_1 :: d_0 \upharpoonright^* d_1 :: b_0 \upharpoonright^* b_1 :: d_1 \upharpoonright^* d_2$;
- $b_0 \upharpoonright^* b_1 :: d_0 \upharpoonright^* d_2$

Two Orthogonal Abstractions

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

Abstraction by Objective Sequences

- $c_0 \Rsh^* c_1 :: d_0 \Rsh^* d_1 :: b_0 \Rsh^* b_1 :: d_1 \Rsh^* d_2$;
- $b_0 \Rsh^* b_1 :: d_0 \Rsh^* d_2$
- $d_0 \Rsh^* d_2$, …

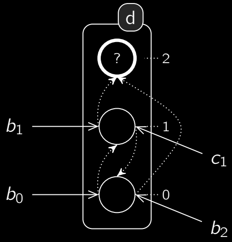
Two Orthogonal Abstractions

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

Abstraction by Objective Sequences

- $c_0 \Rsh^* c_1 :: d_0 \Rsh^* d_1 :: b_0 \Rsh^* b_1 :: d_1 \Rsh^* d_2$;
- $b_0 \Rsh^* b_1 :: d_0 \Rsh^* d_2$
- $d_0 \Rsh^* d_2$, ...

Abstraction by Bounce Sequences



E.g.: $b_0 \to d_0 \Rsh d_1 :: b_1 \to d_1 \Rsh d_2$ $(d_0 \Rsh^* d_2)$

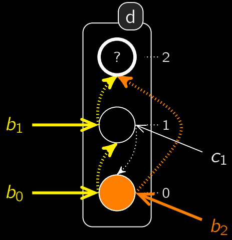
Two Orthogonal Abstractions

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

Abstraction by Objective Sequences

- $c_0 \Rsh^* c_1 :: d_0 \Rsh^* d_1 :: b_0 \Rsh^* b_1 :: d_1 \Rsh^* d_2$;
- $b_0 \Rsh^* b_1 :: d_0 \Rsh^* d_2$
- $d_0 \Rsh^* d_2$, ...

Abstraction by Bounce Sequences



E.g.: $b_0 \rightarrow d_0 \Rsh d_1 :: b_1 \rightarrow d_1 \Rsh d_2$ $(d_0 \Rsh^* d_2)$
$\Rightarrow$ can be computed off-line:

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

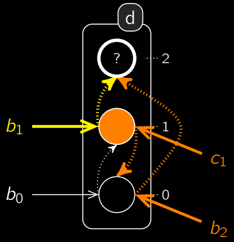
Two Orthogonal Abstractions

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

Abstraction by Objective Sequences

- $c_0 \Rsh^* c_1 :: d_0 \Rsh^* d_1 :: b_0 \Rsh^* b_1 :: d_1 \Rsh^* d_2$;
- $b_0 \Rsh^* b_1 :: d_0 \Rsh^* d_2$
- $d_0 \Rsh^* d_2$, ...

Abstraction by Bounce Sequences



E.g.: $b_0 \rightarrow d_0 \Rsh d_1 :: b_1 \rightarrow d_1 \Rsh d_2$ $(d_0 \Rsh^* d_2)$
$\Rightarrow$ can be computed off-line:

- $\mathbf{BS}(d_0 \Rsh^* d_2) = \{b_0 \rightarrow d_0 \Rsh d_1 :: b_1 \rightarrow d_1 \Rsh d_2,$
  $b_2 \rightarrow d_0 \Rsh d_2]\}$;
- $\mathbf{BS}^\wedge(d_0 \Rsh^* d_2) = \{\{b_0, b_1\}, \{b_2\}\}$.
- $\mathbf{BS}(d_1 \Rsh^* d_2) = \{b_1 \rightarrow d_1 \Rsh d_2,$
  $c_1 \rightarrow d_1 \Rsh d_0 :: b_2 \rightarrow d_0 \Rsh d_2\}$;
- $\mathbf{BS}^\wedge(d_1 \Rsh^* d_2) = \{\{b_1\}, \{b_2, c_1\}\}$.

Objective Sequence Refinements

$$\gamma_\varsigma(\omega) = \{\delta \in \textbf{Sce} \mid \omega \text{ abstracts } \delta \wedge \mathrm{support}(\delta) \subseteq \varsigma\}.$$

Idea: the more details we know, the better $\gamma_\varsigma(\omega)$ should be understood.

Objective Sequence Refinements

$$\gamma_\varsigma(\omega) = \{\delta \in \textbf{Sce} \mid \omega \text{ abstracts } \delta \wedge \mathrm{support}(\delta) \subseteq \varsigma\}.$$

Idea: the more details we know, the better $\gamma_\varsigma(\omega)$ should be understood.

Objective Refinement by $\mathbf{BS}^\wedge$: $\rho^\wedge$

$\mathbf{Obj} \times \wp(\mathbf{BS}^\wedge)$	$\wp(\mathbf{OS})$
$d_0 \Rsh^* d_2$, $\{\{b_0, b_1\}, \{b_2\}\}$	$\star \Rsh^* b_0 :: b_0 \Rsh^* b_1 :: d_0 \Rsh^* d_2$, $\star \Rsh^* b_1 :: b_1 \Rsh^* b_0 :: d_0 \Rsh^* d_2$, $\star \Rsh^* b_2 :: d_0 \Rsh^* d_2$
$\gamma_\varsigma(d_0 \Rsh^* d_2)$	$= \gamma_\varsigma(\rho^\wedge(d_0 \Rsh^* d_2, \mathbf{BS}^\wedge(d_0 \Rsh^* d_2)))$

Objective Sequence Refinements

$$\gamma_\varsigma(\omega) = \{\delta \in \textbf{Sce} \mid \omega \text{ abstracts } \delta \wedge \mathrm{support}(\delta) \subseteq \varsigma\}.$$

Idea: the more details we know, the better $\gamma_\varsigma(\omega)$ should be understood.

Objective Refinement by $\mathbf{BS}^\wedge$: $\rho^\wedge$

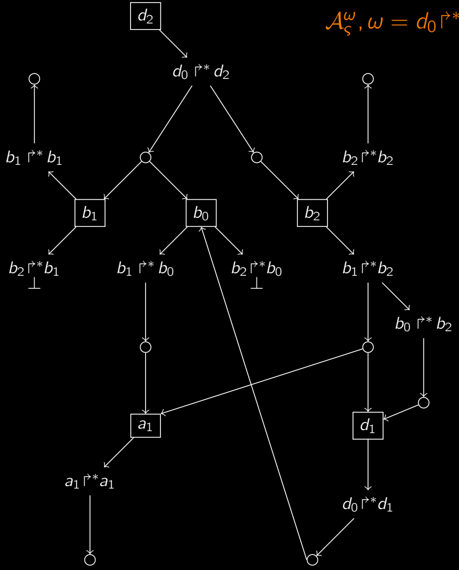
$\mathbf{Obj} \times \wp(\mathbf{BS}^\wedge)$	$\wp(\mathbf{OS})$
$d_0 \upharpoonright^* d_2$, $\{\{b_0, b_1\}, \{b_2\}\}$	$\star \upharpoonright^* b_0 :: b_0 \upharpoonright^* b_1 :: d_0 \upharpoonright^* d_2$, $\star \upharpoonright^* b_1 :: b_1 \upharpoonright^* b_0 :: d_0 \upharpoonright^* d_2$, $\star \upharpoonright^* b_2 :: d_0 \upharpoonright^* d_2$
$\gamma_\varsigma(d_0 \upharpoonright^* d_2)$	$= \gamma_\varsigma(\rho^\wedge(d_0 \upharpoonright^* d_2, \mathbf{BS}^\wedge(d_0 \upharpoonright^* d_2)))$

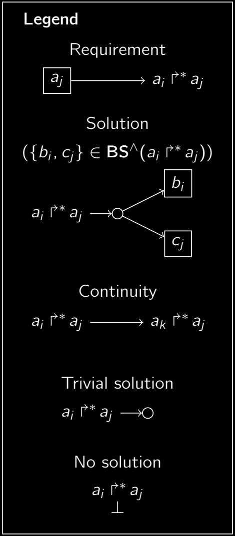
Generalization to OS refinements: $\widetilde{\rho}$

$\mathbf{OS} \times \wp(\mathbf{BS}^\wedge)$	$\wp(\mathbf{OS})$
$\omega, \mathbf{BS}^\wedge$	$\mathrm{interleave}\begin{pmatrix}\omega' \\ \omega_{1..n-1}\end{pmatrix} :: \omega_{n..\lvert\omega\rvert}$ where $n \in \mathbb{I}^\omega$ and $\omega' :: \omega_n \in \rho^\wedge(\omega_n, \mathbf{BS}^\wedge(\omega_n))$
$\gamma_\varsigma(\omega)$	$= \gamma_\varsigma(\widetilde{\rho}(\omega, \mathbf{BS}^\wedge))$

Abstract Structure of Process Hitting

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





Approximations of Successive Reachability

Over-approximations

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

No / Inconc

⇑

Successive Reachability

⇑

Under-approximations

- Un-ordered approximation.
- Ordered approximation.

Yes / Inconc

Approximations of Successive Reachability

Over-approximations

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

No / Inconc

⇑

Successive Reachability

⇑

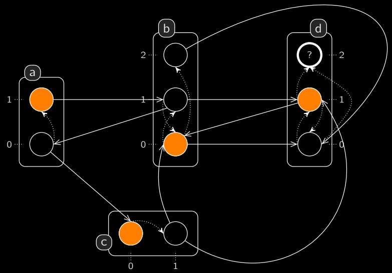
Under-approximations

- Un-ordered approximation.
- Ordered approximation.

Yes / Inconc

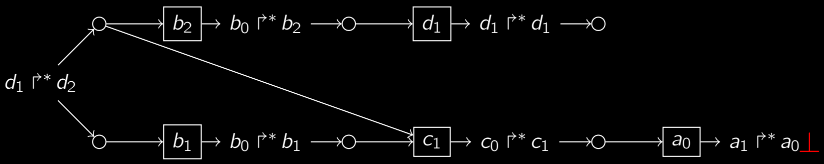
Un-ordered Over-approximation

Example



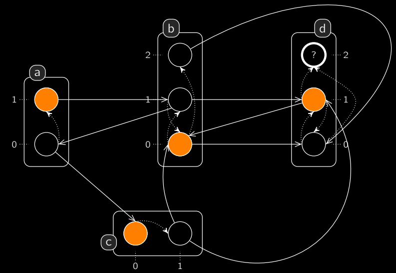
Necessary condition for $\gamma_\varsigma(\omega) \neq \emptyset$:
From each objective within ω, there exists a traversal of $\mathcal{A}_\varsigma^\omega$ such that:

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



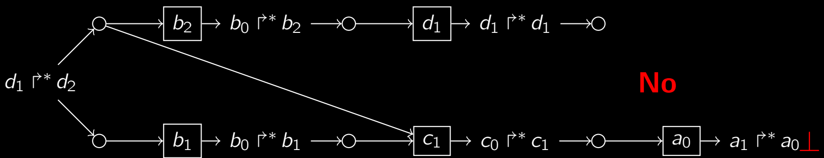
Un-ordered Over-approximation

Example



Necessary condition for $\gamma_\varsigma(\omega) \neq \emptyset$:
From each objective within ω, there exists a traversal of $\mathcal{A}_\varsigma^\omega$ such that:

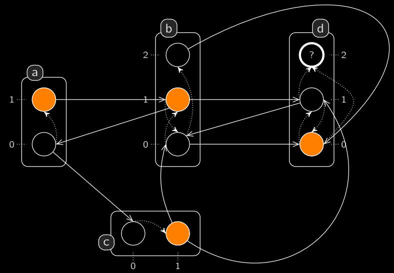
- objective $\rightarrow$ follow at least one solution;
- process $\rightarrow$ follow all objectives;
- no cycle.



No

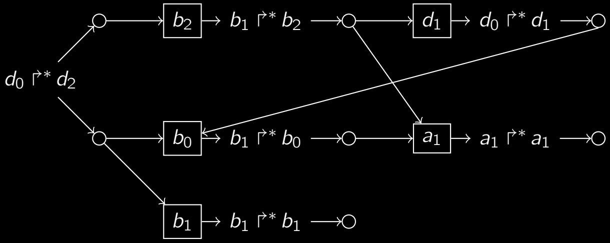
Un-ordered Over-approximation

Example



Necessary condition for $\gamma_\varsigma(\omega) \neq \emptyset$:
From each objective within ω, 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.



Inconc

Approximations of Successive Reachability

Over-approximations

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

No / Inconc

⇑

Successive Reachability

⇑

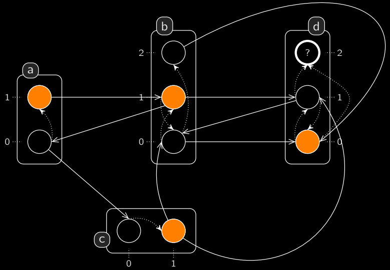
Under-approximations

- Un-ordered approximation.
- Ordered approximation.

Yes / Inconc

Un-ordered Under-approximation

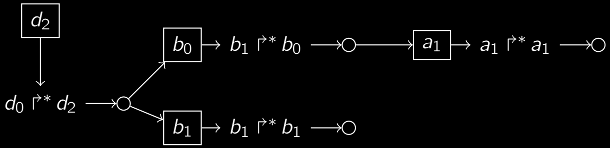
Example



Sufficient condition for $\gamma_\varsigma(\omega) \neq \emptyset$:

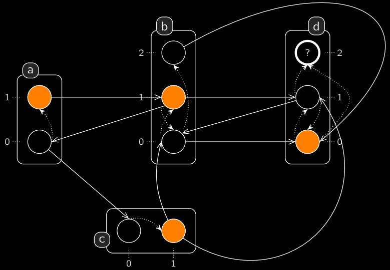
- $\lceil \mathcal{B}^\omega_\varsigma \rceil$ has no cycle;
- each objective has at least one solution.

$\lceil \mathcal{B}^\omega_\varsigma \rceil$: saturated $\mathcal{A}^\omega_\varsigma$.



Un-ordered Under-approximation

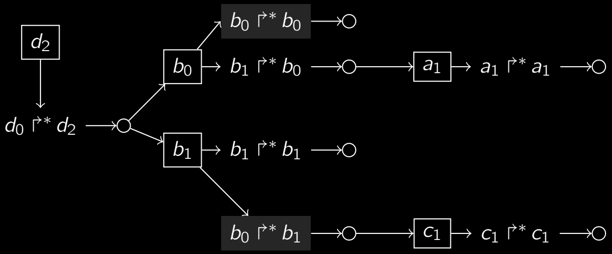
Example



Sufficient condition for $\gamma_\varsigma(\omega) \neq \emptyset$:

- $\lceil \mathcal{B}_\varsigma^\omega \rceil$ has no cycle;
- each objective has at least one solution.

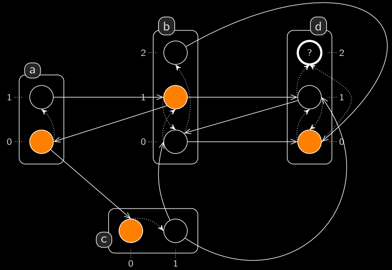
$\lceil \mathcal{B}_\varsigma^\omega \rceil$: saturated $\mathcal{A}_\varsigma^\omega$.



Yes

Un-ordered Under-approximation

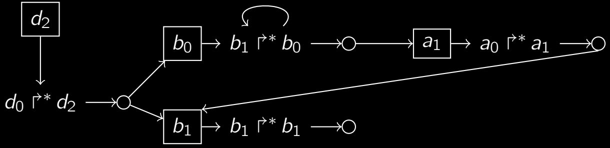
Example



Sufficient condition for $\gamma_\varsigma(\omega) \neq \emptyset$:

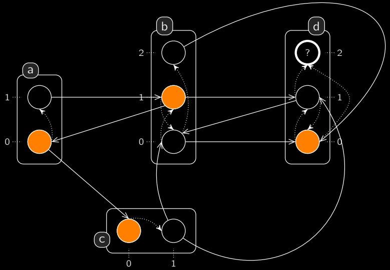
- $\lceil \mathcal{B}^\omega_\varsigma \rceil$ has no cycle;
- each objective has at least one solution.

$\lceil \mathcal{B}^\omega_\varsigma \rceil$: saturated $\mathcal{A}^\omega_\varsigma$.



Un-ordered Under-approximation

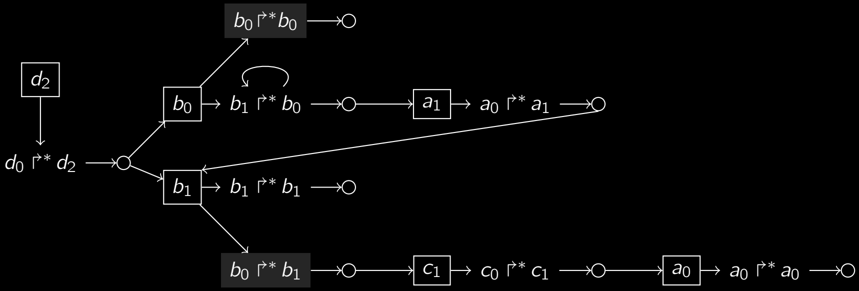
Example



Sufficient condition for $\gamma_\varsigma(\omega) \neq \emptyset$:

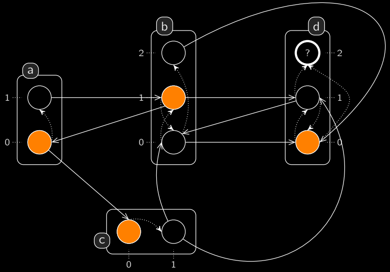
- $\lceil \mathcal{B}^\omega_\varsigma \rceil$ has no cycle;
- each objective has at least one solution.

$\lceil \mathcal{B}^\omega_\varsigma \rceil$: saturated $\mathcal{A}^\omega_\varsigma$.



Un-ordered Under-approximation

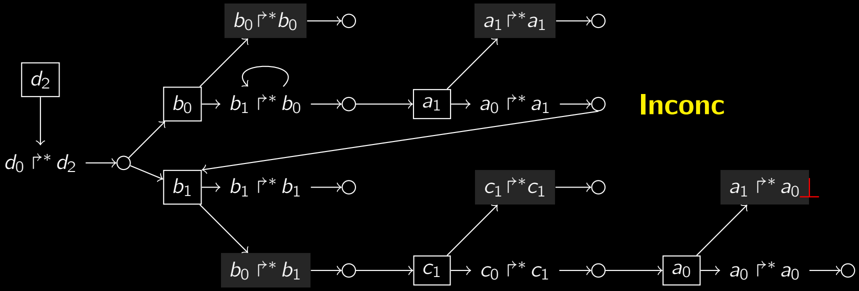
Example



Sufficient condition for $\gamma_\varsigma(\omega) \neq \emptyset$:

- $\lceil \mathcal{B}^\omega_\varsigma \rceil$ has no cycle;
- each objective has at least one solution.

$\lceil \mathcal{B}^\omega_\varsigma \rceil$: saturated $\mathcal{A}^\omega_\varsigma$.



Static Analysis of Successive Reachability

Over-approximations

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

No / Inconc

⇑

Successive Reachability

⇑

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

$\Uparrow$

Successive Reachability

$\Uparrow$

Under-approximations

- Un-ordered approximation.
- Ordered approximation.

Yes / Inconc

Still inconclusive?

- Require new analyses of the abstract structure
- $\Rightarrow$ drive refinements of ω.

Complexity

Abstract Strutures $\mathcal{A}^{\omega}_{\varsigma}$, $\lceil\mathcal{B}^{\omega}_{\varsigma}\rceil$

- $\mathbf{BS}^{\wedge}$ computation: exponential in the number of processes within a single sort.
- Size of $\mathbf{BS}^{\wedge}$: combinaisons of $|\mathbf{Proc}_a|$ processes $\binom{|\mathbf{Proc}|}{|\mathbf{Proc}_a|}$.
- Size of $\mathcal{A}^{\omega}_{\varsigma}$ (and $\lceil\mathcal{B}^{\omega}_{\varsigma}\rceil$): polynomial in processes number $\times$ size of $\mathbf{BS}^{\wedge}$.

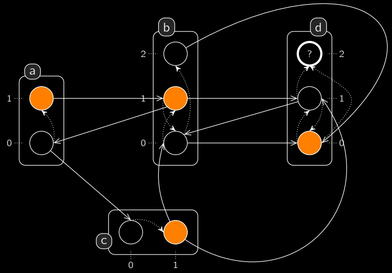
Analyses

- Over-approximations: polynomial in the size of $\mathcal{A}^{\omega}_{\varsigma}$.
- Different strategies of under-approximation:
  - global: polynomial in the size of $\lceil\mathcal{B}^{\omega}_{\varsigma}\rceil$;
  - per solution: $\times$ exponential in the size of $\mathbf{BS}^{\wedge}$.

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

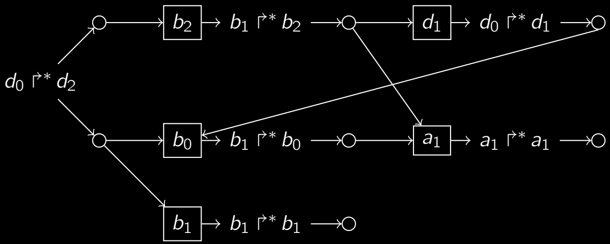
Outline

Extraction of Key Processes



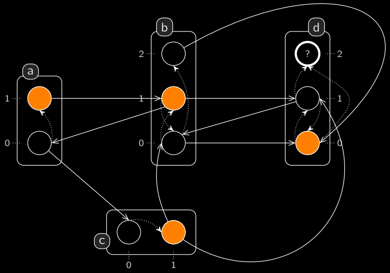
Necessary condition for $\gamma_\varsigma(\omega) \neq \emptyset$:
From each objective within ω, there exists a traversal of $\mathcal{A}_\varsigma^\omega$ such that:

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



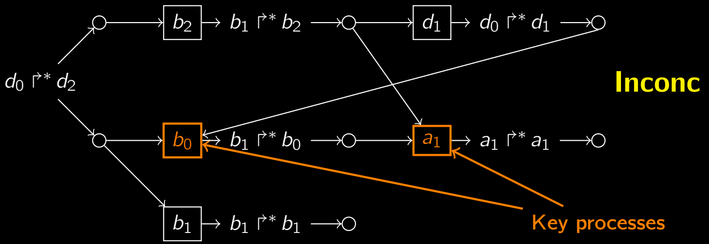
Inconc

Extraction of Key Processes



Necessary condition for $\gamma_\varsigma(\omega) \neq \emptyset$:
From each objective within ω, there exists a traversal of $\mathcal{A}^{\omega}_{\varsigma}$ such that:

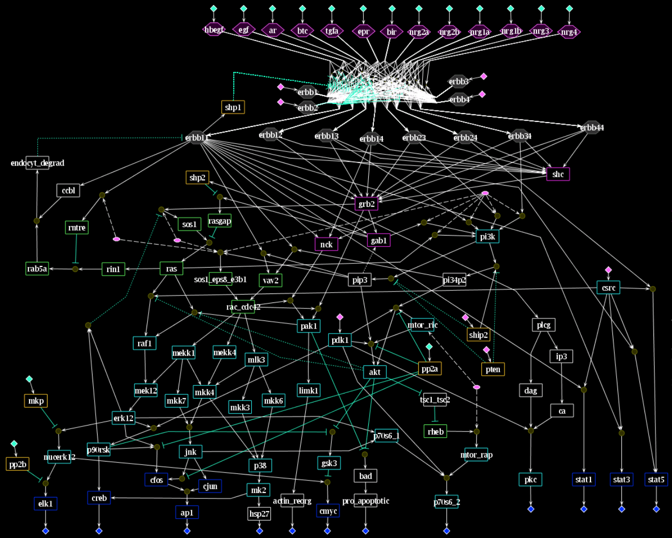
- objective $\rightarrow$ follow at least one solution;
- process $\rightarrow$ follow all objectives;
- no cycle.



Outline

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	PINT[2]
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 [Inria Paris-Rocquencourt/Contraintes] using NuSMV2

2 `http://process.hitting.free.fr`

Outline

Conclusion

Static Analysis of Process Hittings

- Static listing of fixed points.
- Very efficient approximations of successive reachability properties.
- Key processes uncovering (necessary to a given reachability) (towards control).
- Make tractable the formal analysis of large Biological Regulatory Networks.

Future work

- Formal link with event structures (such as Petri Nets unfoldings);
- Improve the analysis with libddd?
- Extension to the Process Hitting with Priorities
  (allows the weak bisimulation of CFSM).

Thank you for your attention.