Reachability Analysis and Revision of Dynamics of Biological Regulatory Networks

Xinwei Chai

École Centrale de Nantes
Le Laboratoire des Sciences du Numérique de Nantes
xinwei.chai@ls2n.fr

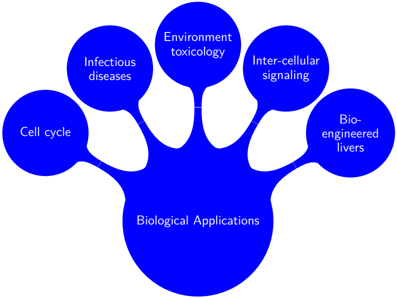
Rapporteurs : Gilles BERNOT, Professeur des universités, Université Côte d'Azur
Pascale LE GALL, Professeur des universités, Centrale Supélec

Examinateurs : Béatrice DUVAL, Professeur des universités, Université d'Angers
Loïc PAULEVÉ, Chargé de recherche, LaBRI, UMR CNRS

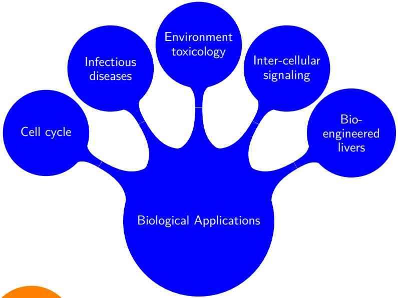
Directeur de thèse : Olivier ROUX, Professeur des universités, École Centrale de Nantes
Co-encadrant de thèse : Morgan MAGNIN, Professeur des universités, École Centrale de Nantes

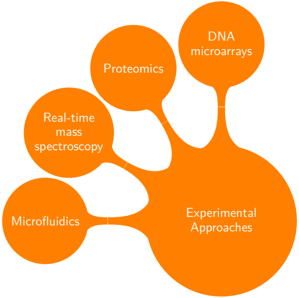
May 24, 2019

Positioning of Our Work

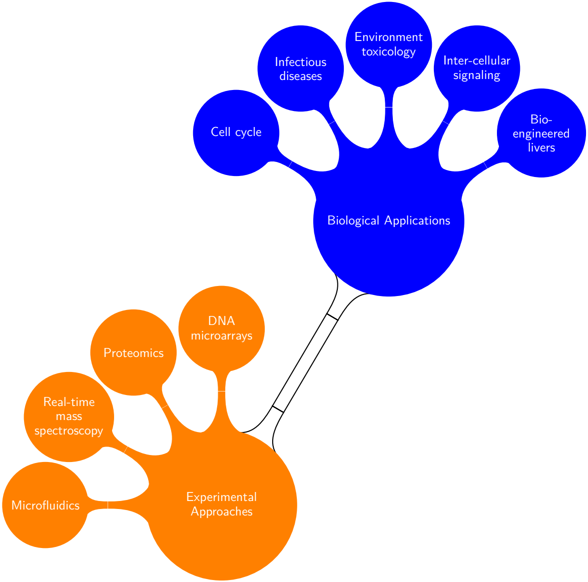


Positioning of Our Work

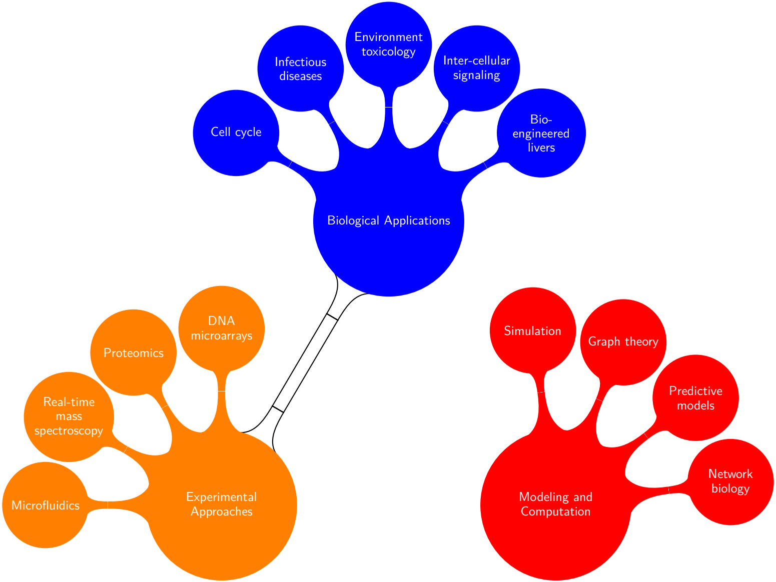




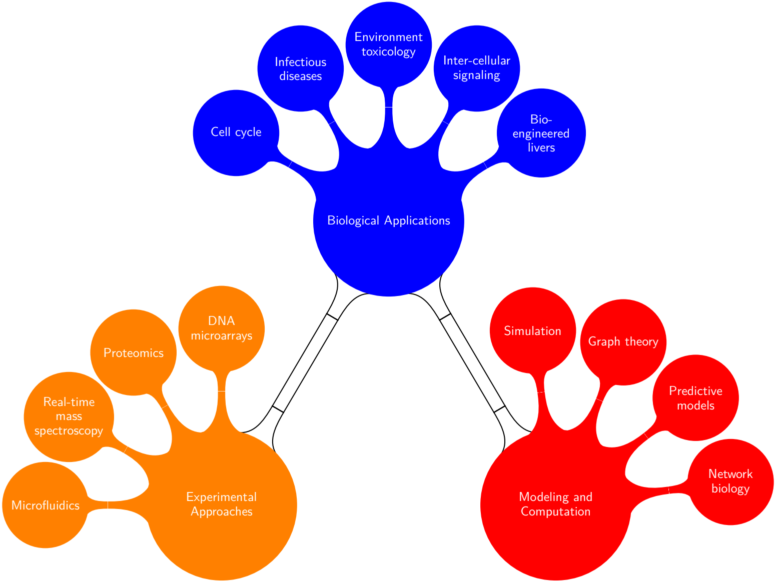
Positioning of Our Work



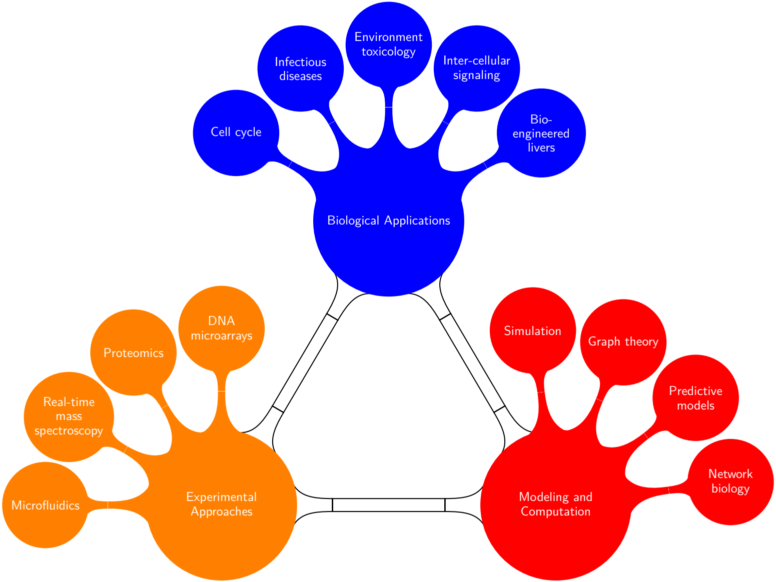
Positioning of Our Work



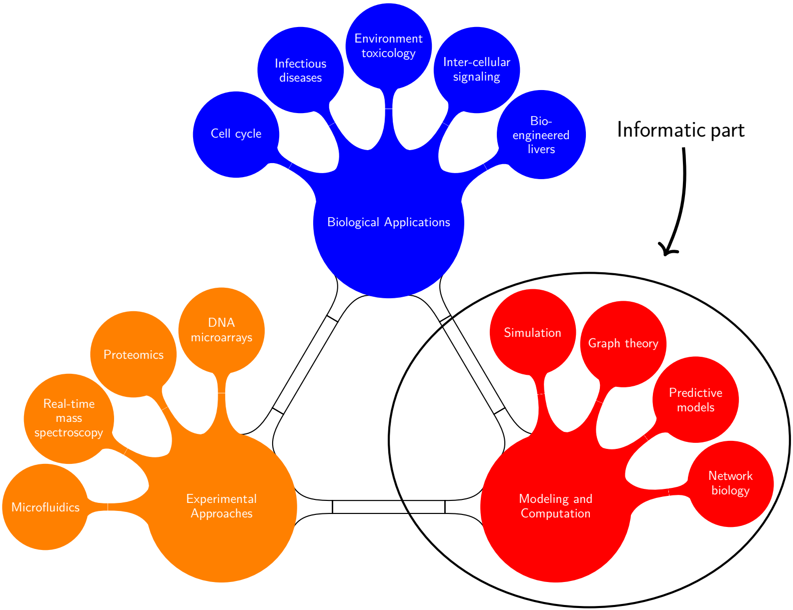
Positioning of Our Work



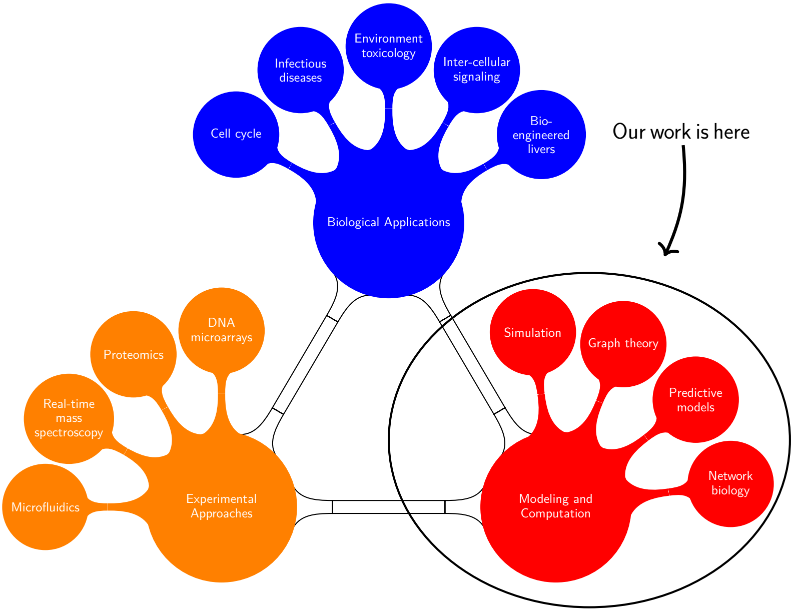
Positioning of Our Work



Positioning of Our Work

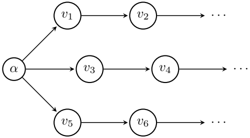


Positioning of Our Work



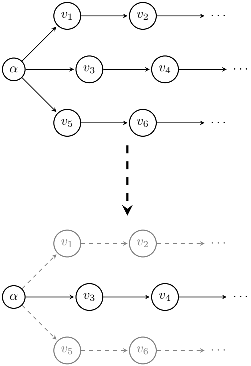
Problematics

Real system dynamics



Problematics

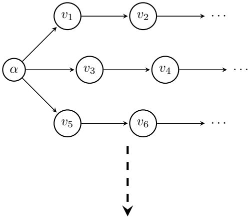
Real system dynamics

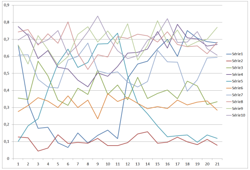


Partial observation

Problematics

Real system dynamics

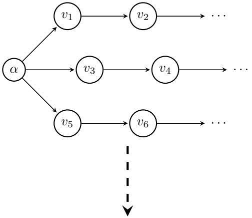


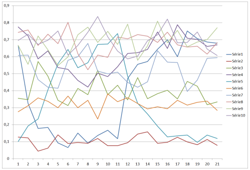


Time series data

Problematics

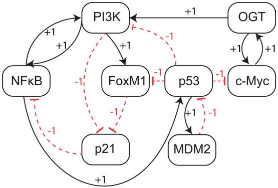
Real system dynamics





Time series data

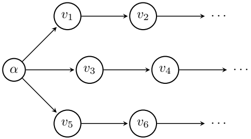
Learning



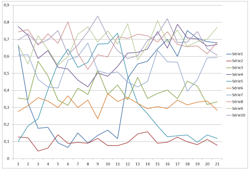
Biological network

Problematics

Real system dynamics

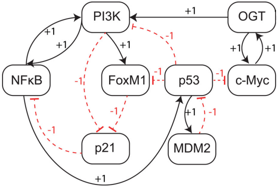


Reproducible?



Time series data

Learning

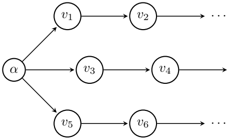


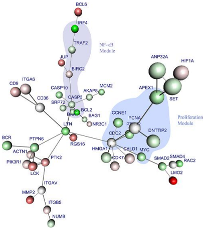
Biological network

Problematics

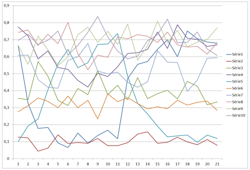
Biological *a priori* knowledge

Real system dynamics



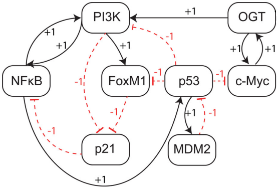


Reproducible?



Time series data

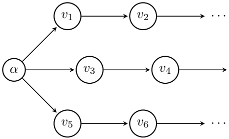
Learning



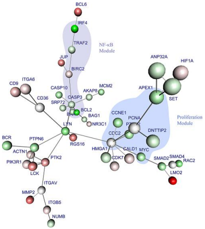
Biological network

Problematics

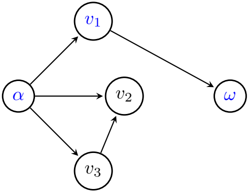
Real system dynamics



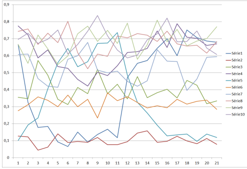
Biological *a priori* knowledge



Reachability properties

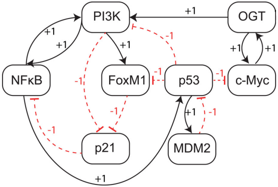


Reproducible?



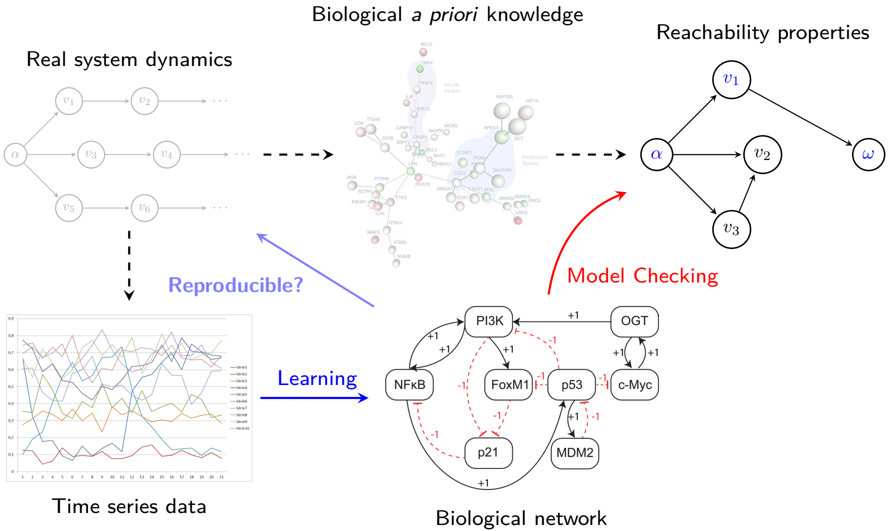
Time series data

Learning



Biological network

Problematics



Real system dynamics

Biological *a priori* knowledge

Reachability properties

Reproducible?

Learning

Model Checking

Time series data

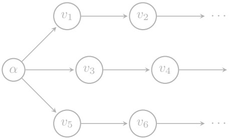
Biological network

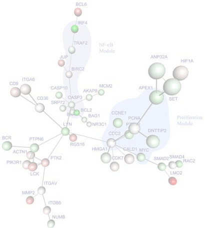
Problematics

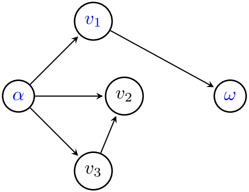
Biological *a priori* knowledge

Reachability properties

Real system dynamics



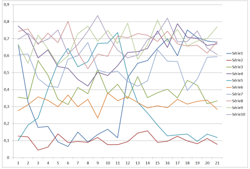




Reproducible?

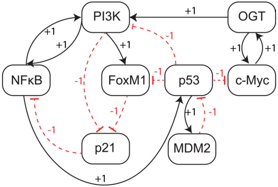
Model Checking

Revising



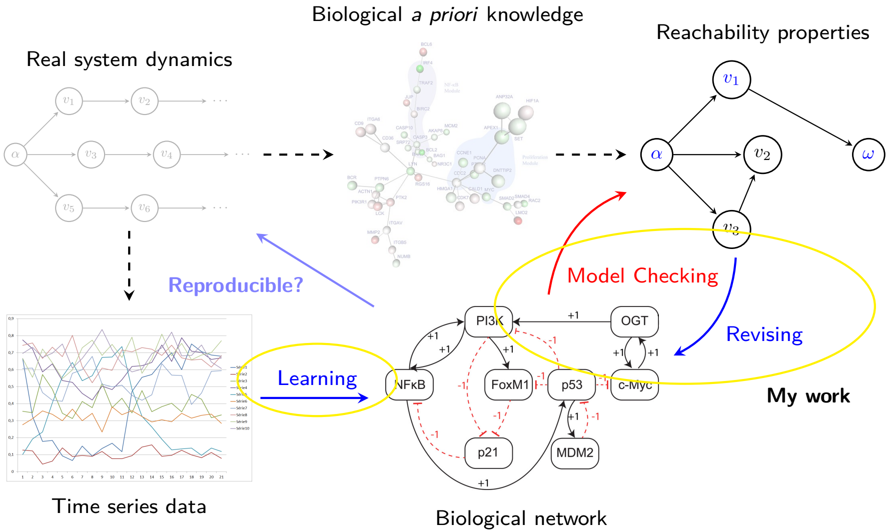
Time series data

Learning



Biological network

Problematics



Biological *a priori* knowledge

Reachability properties

Real system dynamics

Reproducible?

Model Checking

Revising

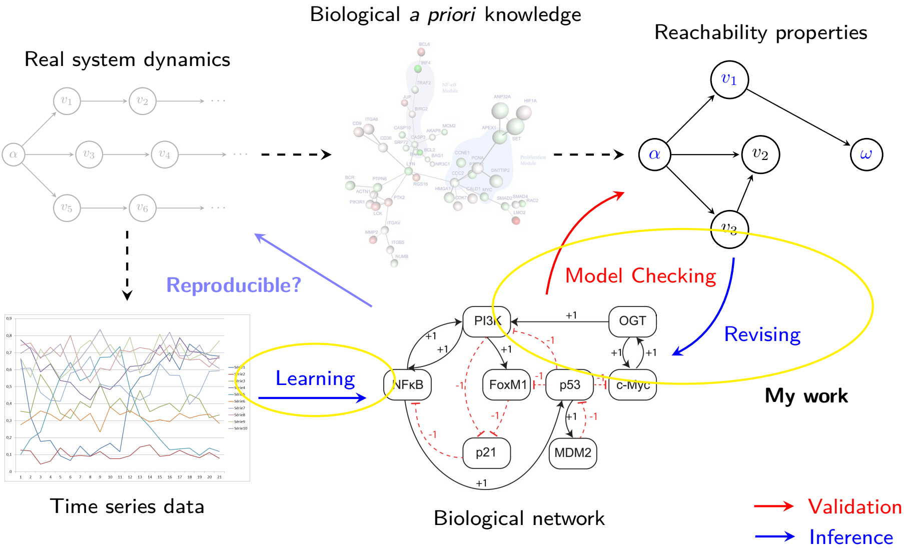
Learning

My work

Time series data

Biological network

Problematics



Biological *a priori* knowledge

Real system dynamics

Reachability properties

Reproducible?

Model Checking

Revising

Learning

My work

Time series data

Biological network

Validation

Inference

Overview of My Contribution

- Model checkers

Overview of My Contribution

- Model checkers
    - Verify if the model meets a given specification

Overview of My Contribution

- Model checkers
  - Verify if the model meets a given specification
  - OBDD-based (Ordered Binary Decision Diagram) model checkers [BHSV^{+}96, CCGR00] are able to check over 10^{120} states but still not enough

Overview of My Contribution

- Model checkers
  - Verify if the model meets a given specification
  - OBDD-based (Ordered Binary Decision Diagram) model checkers [BHSV$^+$96, CCGR00] are able to check over 10^{120} states but still not enough
  - Static analyzers [PMR12] and bounded model checkers [CBRZ01] are computationally acceptable but have limits on completeness

Overview of My Contribution

- Model checkers
  - Verify if the model meets a given specification
  - OBDD-based (Ordered Binary Decision Diagram) model checkers [BHSV$^+$96, CCGR00] are able to check over 10^{120} states but still not enough
  - Static analyzers [PMR12] and bounded model checkers [CBRZ01] are computationally acceptable but have limits on completeness
  - → **High-performance reachability analyzers with both advantages**

Overview of My Contribution

- Model checkers
  - Verify if the model meets a given specification
  - OBDD-based (Ordered Binary Decision Diagram) model checkers [BHSV^{+}96, CCGR00] are able to check over 10^{120} states but still not enough
  - Static analyzers [PMR12] and bounded model checkers [CBRZ01] are computationally acceptable but have limits on completeness
  - → **High-performance reachability analyzers with both advantages**
- Model learning tools

Overview of My Contribution

- Model checkers
  - Verify if the model meets a given specification
  - OBDD-based (Ordered Binary Decision Diagram) model checkers [BHSV$^+$96, CCGR00] are able to check over 10^{120} states but still not enough
  - Static analyzers [PMR12] and bounded model checkers [CBRZ01] are computationally acceptable but have limits on completeness
  - → **High-performance reachability analyzers with both advantages**
- Model learning tools
  - Construct models from data

Overview of My Contribution

- Model checkers
  - Verify if the model meets a given specification
  - OBDD-based (Ordered Binary Decision Diagram) model checkers [BHSV^{+}96, CCGR00] are able to check over 10^{120} states but still not enough
  - Static analyzers [PMR12] and bounded model checkers [CBRZ01] are computationally acceptable but have limits on completeness
  - → **High-performance reachability analyzers with both advantages**
- Model learning tools
  - Construct models from data
  - Tools for learning static systems [MKIN18]

Overview of My Contribution

- Model checkers
  - Verify if the model meets a given specification
  - OBDD-based (Ordered Binary Decision Diagram) model checkers [BHSV^{+}96, CCGR00] are able to check over 10^{120} states but still not enough
  - Static analyzers [PMR12] and bounded model checkers [CBRZ01] are computationally acceptable but have limits on completeness
  - → **High-performance reachability analyzers with both advantages**
- Model learning tools
  - Construct models from data
  - Tools for learning static systems [MKIN18]
  - LFIT (Learning From Interpretation Transition) [RFM^{+}18], not tolerant to noise

Overview of My Contribution

- Model checkers
  - Verify if the model meets a given specification
  - OBDD-based (Ordered Binary Decision Diagram) model checkers [BHSV+96, CCGR00] are able to check over 10^{120} states but still not enough
  - Static analyzers [PMR12] and bounded model checkers [CBRZ01] are computationally acceptable but have limits on completeness
  - → **High-performance reachability analyzers with both advantages**
- Model learning tools
  - Construct models from data
  - Tools for learning static systems [MKIN18]
  - LFIT (Learning From Interpretation Transition) [RFM+18], not tolerant to noise
  - → **A fuzzy tool learning system dynamics, tolerant to noise**

Overview of My Contribution

- Model checkers
  - Verify if the model meets a given specification
  - OBDD-based (Ordered Binary Decision Diagram) model checkers [BHSV+96, CCGR00] are able to check over 10^{120} states but still not enough
  - Static analyzers [PMR12] and bounded model checkers [CBRZ01] are computationally acceptable but have limits on completeness
  - → **High-performance reachability analyzers with both advantages**
- Model learning tools
  - Construct models from data
  - Tools for learning static systems [MKIN18]
  - LFIT (Learning From Interpretation Transition) [RFM+18], not tolerant to noise
  - → **A fuzzy tool learning system dynamics, tolerant to noise**
- → **Model inference**

Overview of My Contribution

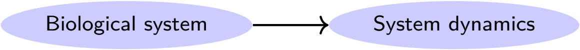
- Model checkers
  - Verify if the model meets a given specification
  - OBDD-based (Ordered Binary Decision Diagram) model checkers [BHSV+96, CCGR00] are able to check over 10^{120} states but still not enough
  - Static analyzers [PMR12] and bounded model checkers [CBRZ01] are computationally acceptable but have limits on completeness
  - → **High-performance reachability analyzers with both advantages**
- Model learning tools
  - Construct models from data
  - Tools for learning static systems [MKIN18]
  - LFIT (Learning From Interpretation Transition) [RFM+18], not tolerant to noise
  - → **A fuzzy tool learning system dynamics, tolerant to noise**
- → **Model inference**
  - Model learning + model revising, aiming at learning a model consistent with *a priori* knowledge, so that the model behaves closer to real system dynamics

Overview of My Contribution

- Model checkers
  - Verify if the model meets a given specification
  - OBDD-based (Ordered Binary Decision Diagram) model checkers [BHSV^{+}96, CCGR00] are able to check over 10^{120} states but still not enough
  - Static analyzers [PMR12] and bounded model checkers [CBRZ01] are computationally acceptable but have limits on completeness
  - → **High-performance reachability analyzers with both advantages**
- Model learning tools
  - Construct models from data
  - Tools for learning static systems [MKIN18]
  - LFIT (Learning From Interpretation Transition) [RFM^{+}18], not tolerant to noise
  - → **A fuzzy tool learning system dynamics, tolerant to noise**
- → **Model inference**
  - Model learning + model revising, aiming at learning a model consistent with *a priori* knowledge, so that the model behaves closer to real system dynamics

Outline: Reachability problem (model checking) → model learning → model revising based on reachability properties

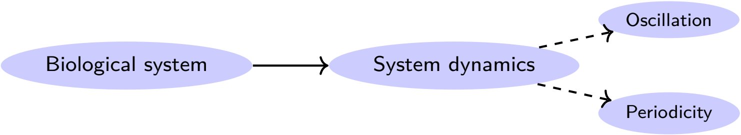
Problematic of Reachability Problem



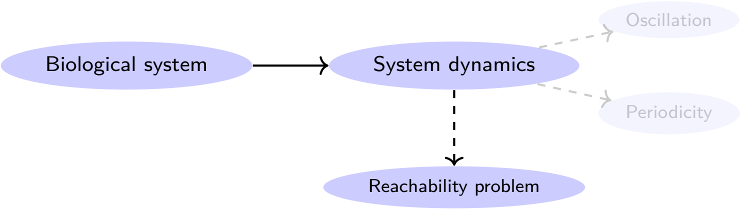
Biological system

System dynamics

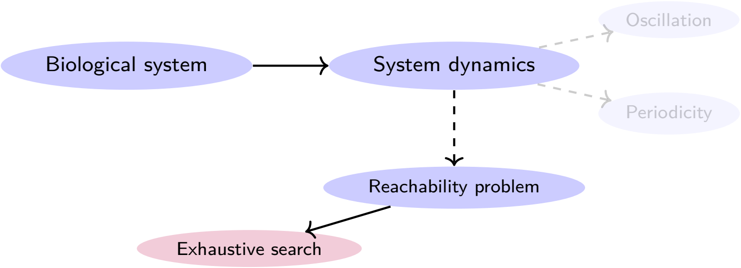
Problematic of Reachability Problem



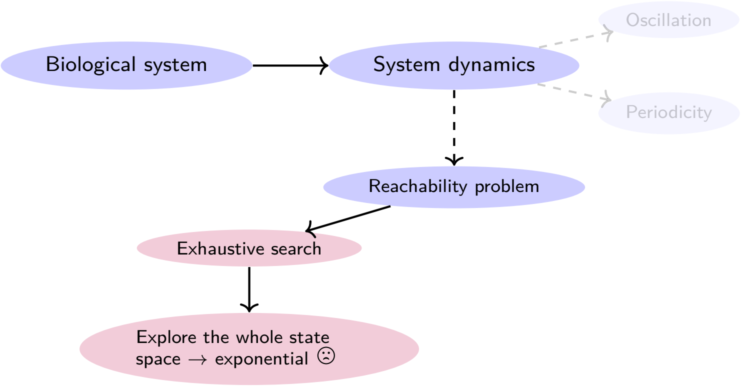
Problematic of Reachability Problem



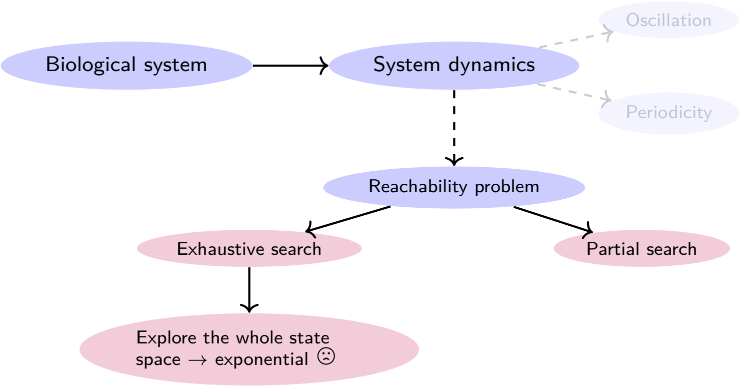
Problematic of Reachability Problem



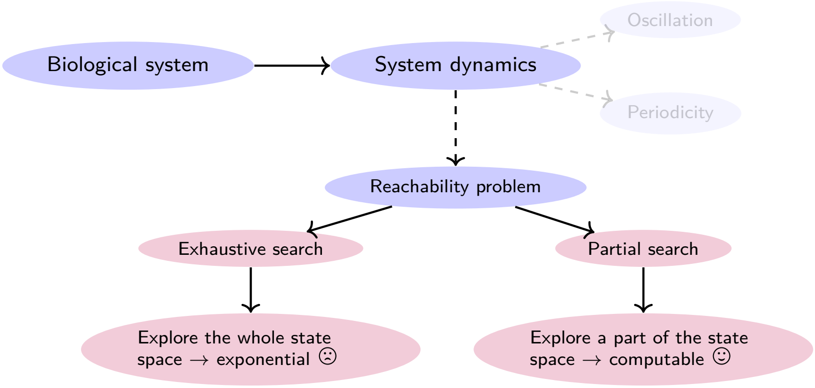
Problematic of Reachability Problem



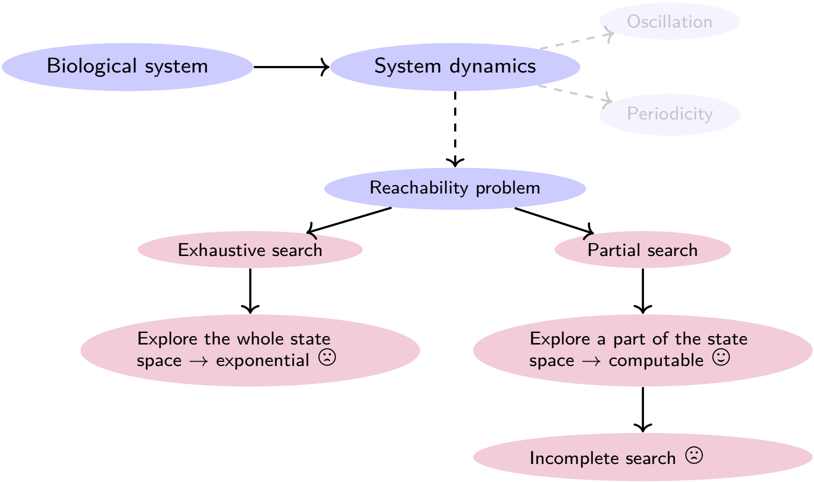
Problematic of Reachability Problem



Problematic of Reachability Problem



Problematic of Reachability Problem



Biological system

System dynamics

Oscillation

Periodicity

Reachability problem

Exhaustive search

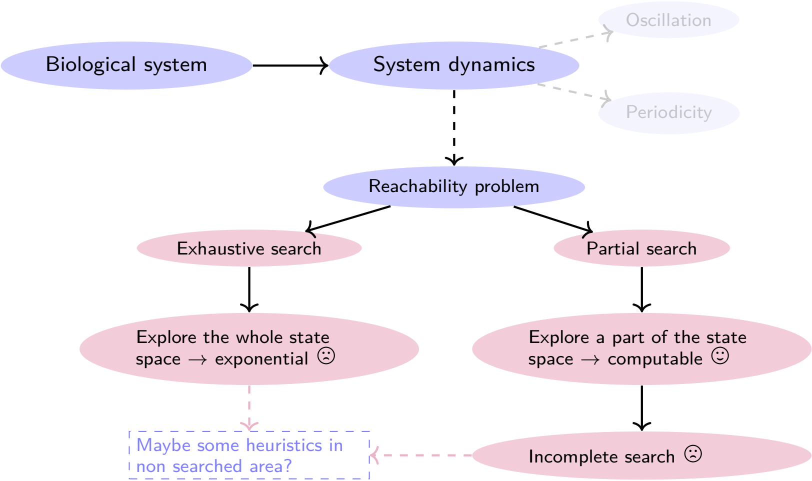
Partial search

Explore the whole state space → exponential ☹

Explore a part of the state space → computable ☺

Incomplete search ☹

Problematic of Reachability Problem



Biological system

System dynamics

Oscillation

Periodicity

Reachability problem

Exhaustive search

Partial search

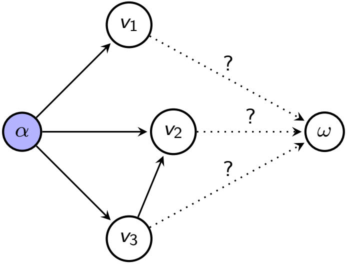
Explore the whole state space → exponential ☹

Explore a part of the state space → computable ☺

Incomplete search ☹

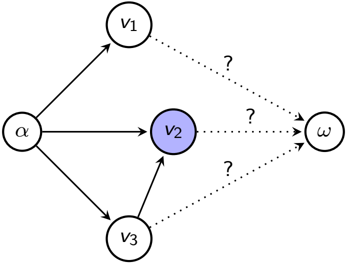
Maybe some heuristics in non searched area?

Reachability Problem Illustrated by Transition Graph (Exhaustive)



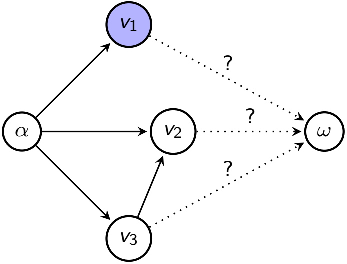
- A digraph representing state space
  - Nodes = system states
  - Edges = state transitions
  - α = initial state
  - ω = desired final state

Reachability Problem Illustrated by Transition Graph (Exhaustive)



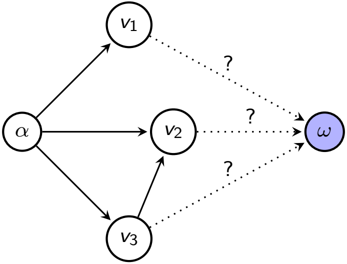
- A digraph representing state space
  - Nodes = system states
  - Edges = state transitions
  - α = initial state
  - ω = desired final state

Reachability Problem Illustrated by Transition Graph (Exhaustive)



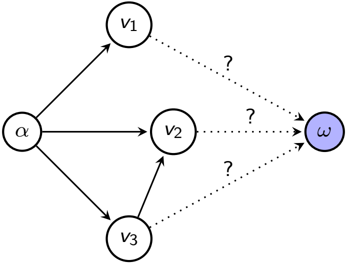
- A digraph representing state space
  - Nodes = system states
  - Edges = state transitions
  - α = initial state
  - ω = desired final state

Reachability Problem Illustrated by Transition Graph (Exhaustive)



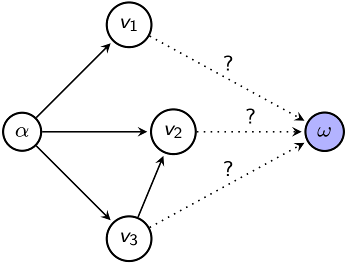
- A digraph representing state space
  - Nodes = system states
  - Edges = state transitions
  - α = initial state
  - ω = desired final state

Reachability Problem Illustrated by Transition Graph (Exhaustive)



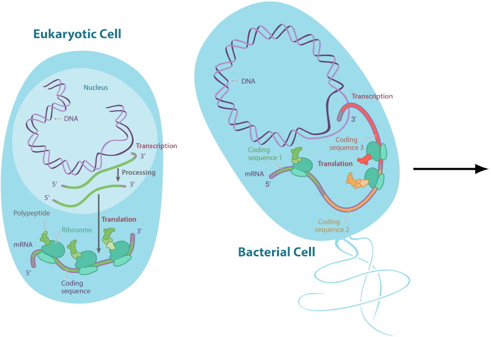
- A digraph representing state space
  - Nodes = system states
  - Edges = state transitions
  - α = initial state
  - ω = desired final state
- Solving reachability of digraphs needs at least polynomial time and space w.r.t $\#$*nodes* [HKV02], but $\#$*nodes* is exponential to $\#$*variables* in the system $\Longrightarrow$ exhaustive search is not acceptable when dealing with a large model

Reachability Problem Illustrated by Transition Graph (Exhaustive)

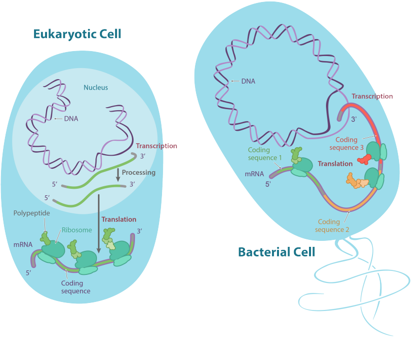


- A digraph representing state space
  - Nodes = system states
  - Edges = state transitions
  - α = initial state
  - ω = desired final state
- Solving reachability of digraphs needs at least polynomial time and space w.r.t $\#$*nodes* [HKV02], but $\#$*nodes* is exponential to $\#$*variables* in the system $\Longrightarrow$ exhaustive search is not acceptable when dealing with a large model
- A pertinent modeling framework is necessary to describe system dynamics

Biological Regulatory Networks (BRN)



Biological Regulatory Networks (BRN)

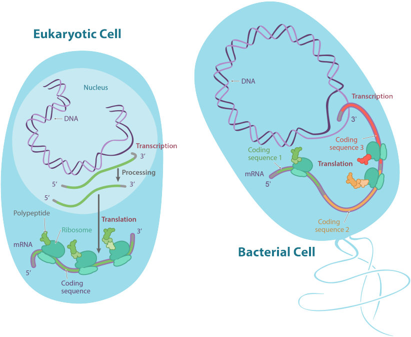


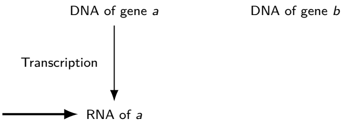
→

DNA of gene *a*

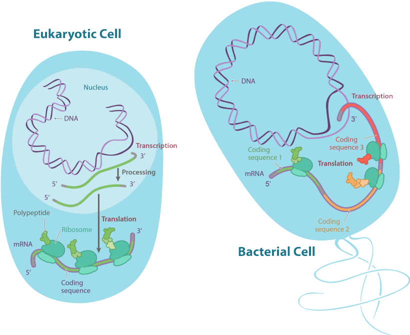
DNA of gene *b*

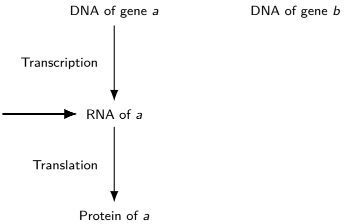
Biological Regulatory Networks (BRN)





Biological Regulatory Networks (BRN)




DNA of gene *a*

DNA of gene *b*

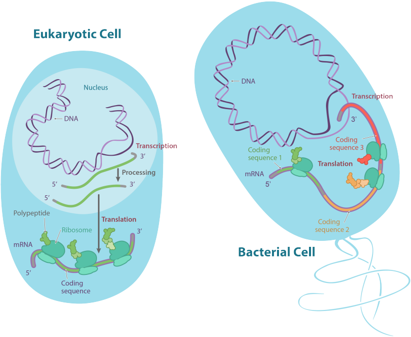
Transcription

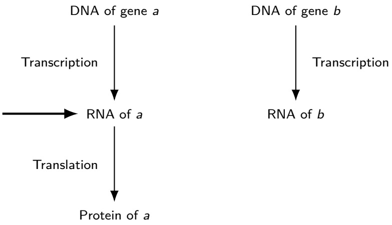
RNA of *a*

Translation

Protein of *a*

Biological Regulatory Networks (BRN)





DNA of gene *a*

Transcription

RNA of *a*

Translation

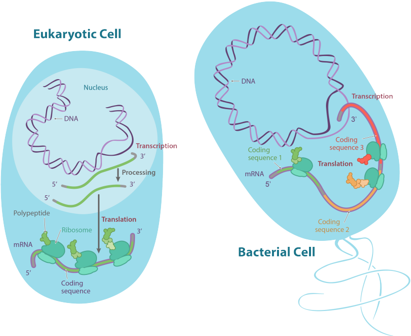
Protein of *a*

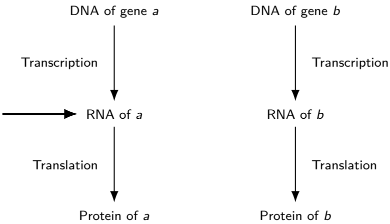
DNA of gene *b*

Transcription

RNA of *b*

Biological Regulatory Networks (BRN)





DNA of gene *a*

Transcription

RNA of *a*

Translation

Protein of *a*

DNA of gene *b*

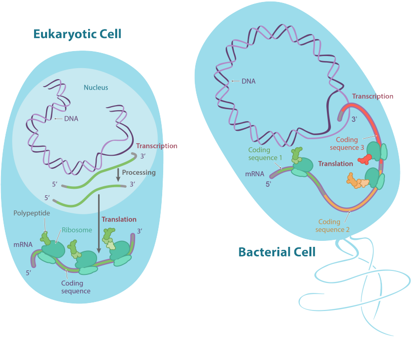
Transcription

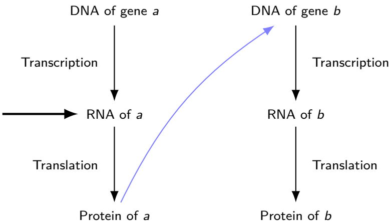
RNA of *b*

Translation

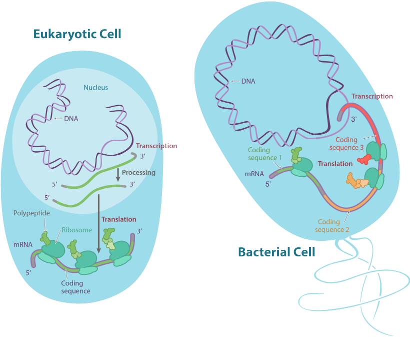
Protein of *b*

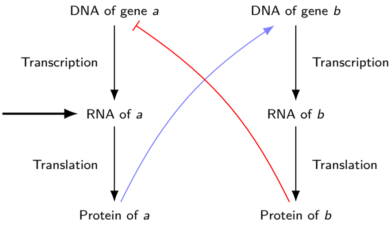
Biological Regulatory Networks (BRN)





Biological Regulatory Networks (BRN)





DNA of gene *a*

DNA of gene *b*

Transcription

Transcription

RNA of *a*

RNA of *b*

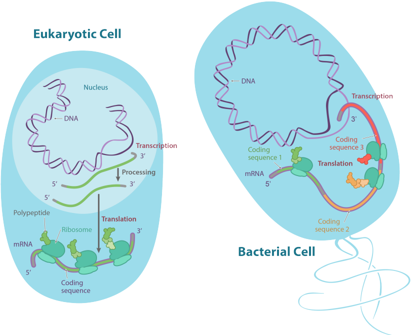
Translation

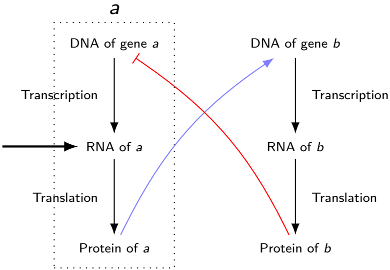
Translation

Protein of *a*

Protein of *b*

Biological Regulatory Networks (BRN)





a

DNA of gene *a*

Transcription

RNA of *a*

Translation

Protein of *a*

DNA of gene *b*

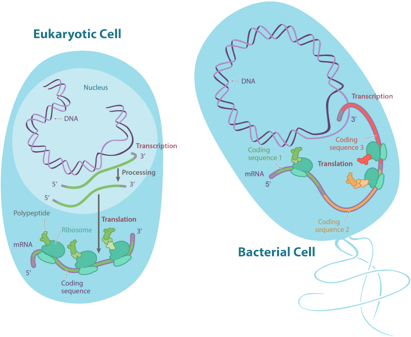
Transcription

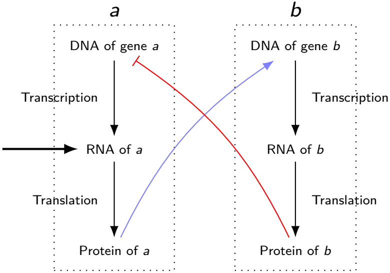
RNA of *b*

Translation

Protein of *b*

Biological Regulatory Networks (BRN)





a

DNA of gene *a*

Transcription

RNA of *a*

Translation

Protein of *a*

b

DNA of gene *b*

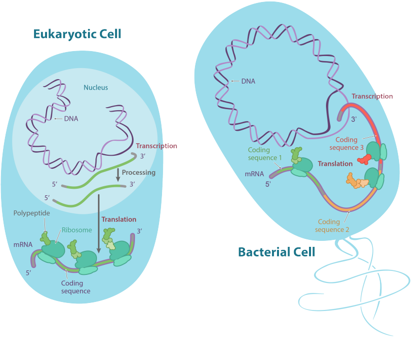
Transcription

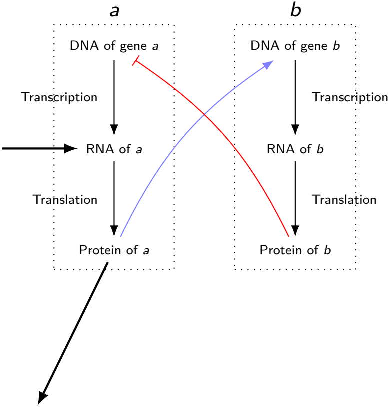
RNA of *b*

Translation

Protein of *b*

Biological Regulatory Networks (BRN)




a

DNA of gene *a*

Transcription

RNA of *a*

Translation

Protein of *a*

b

DNA of gene *b*

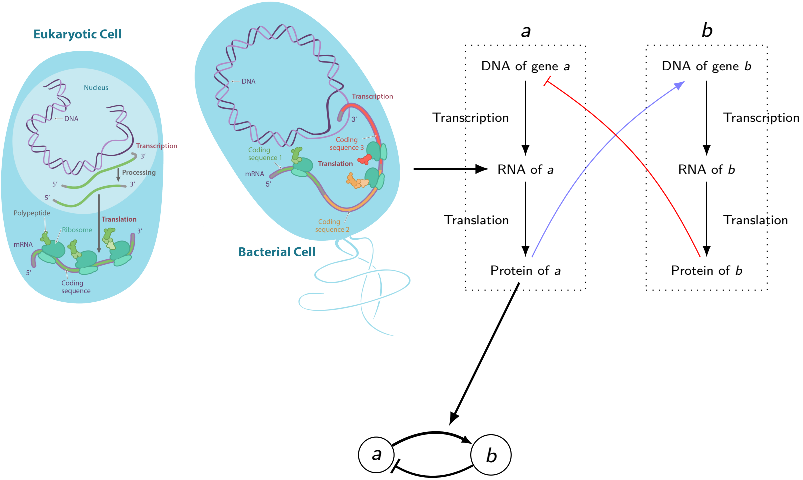
Transcription

RNA of *b*

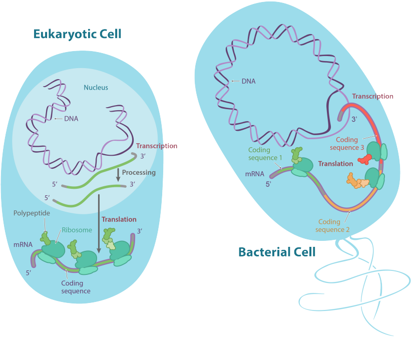
Translation

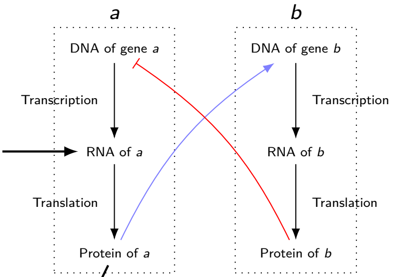
Protein of *b*

Biological Regulatory Networks (BRN)



Biological Regulatory Networks (BRN)





a

DNA of gene *a*

Transcription

RNA of *a*

Translation

Protein of *a*

b

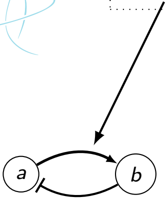
DNA of gene *b*

Transcription

RNA of *b*

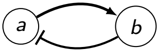
Translation

Protein of *b*



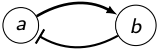
How to analyze system dynamics using BRN?

Discrete Modeling [BT09]



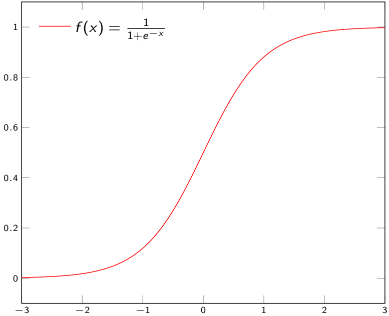
What are the possible values for a and b?

Discrete Modeling [BT09]

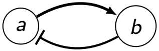


What are the possible values for *a* and *b*?

- Behaviors of elements in BRN can be approximated to sigmoid functions [BT09]

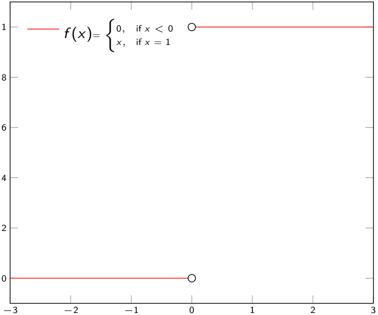


Discrete Modeling [BT09]

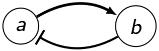


What are the possible values for a and b?

- Behaviors of elements in BRN can be approximated to sigmoid functions [BT09]
- Approximated to piecewise functions. When the concentration of an entity is below the regulation threshold, the regulation is activated, otherwise inactivated.
  The value of a and b is in $\{0, 1, 2, \ldots\}$



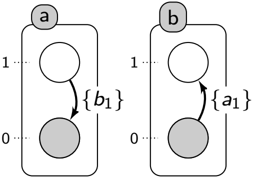
Discrete Modeling [BT09]



What are the possible values for a and b?

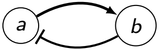
- Behaviors of elements in BRN can be approximated to sigmoid functions [BT09]
- Approximated to piecewise functions. When the concentration of an entity is below the regulation threshold, the regulation is activated, otherwise inactivated.
  The value of a and b is in $\{0, 1, 2, \ldots\}$
- A model containing dynamic information

Automata Network (AN)
A modeling framework representing state transitions and using $O(\#nodes)$ memory



$\{b_1\} \rightarrow a_0$
$\{a_1\} \rightarrow b_1$

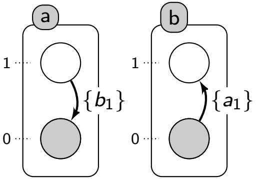
Discrete Modeling [BT09]



What are the possible values for a and b?

- Behaviors of elements in BRN can be approximated to sigmoid functions [BT09]
- Approximated to piecewise functions. When the concentration of an entity is below the regulation threshold, the regulation is activated, otherwise inactivated.
  The value of a and b is in $\{0, 1, 2, \ldots\}$
- A model containing dynamic information

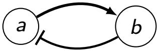
Automata Network (AN)
A modeling framework representing state transitions and using $O(\#nodes)$ memory



$\{b_1\} \to a_0$
$\{a_1\} \to b_1$

The global reachability of a system state $(a, b) = (1, 1)$ is hard to compute

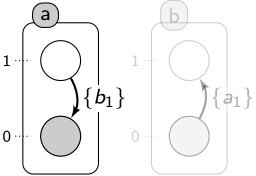
Discrete Modeling [BT09]



What are the possible values for a and b?

- Behaviors of elements in BRN can be approximated to sigmoid functions [BT09]
- Approximated to piecewise functions. When the concentration of an entity is below the regulation threshold, the regulation is activated, otherwise inactivated.
  The value of a and b is in $\{0, 1, 2, \ldots\}$
- A model containing dynamic information

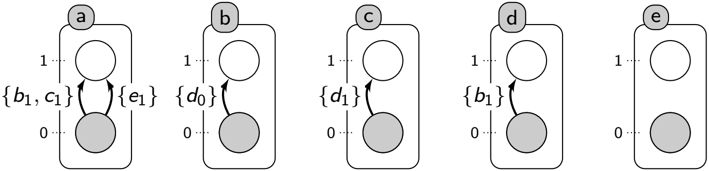
Automata Network (AN)
A modeling framework representing state transitions and using $O(\#nodes)$ memory



$\{b_1\} \rightarrow a_0$
$\{a_1\} \rightarrow b_1$

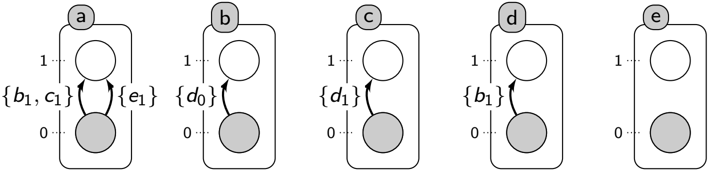
The global reachability of a system state $(a, b) = (1, 1)$ is hard to compute
$\Longrightarrow$ The complexity of the reachability of the value of one variable $a = 1$ can be smaller?

Automata Network (AN)



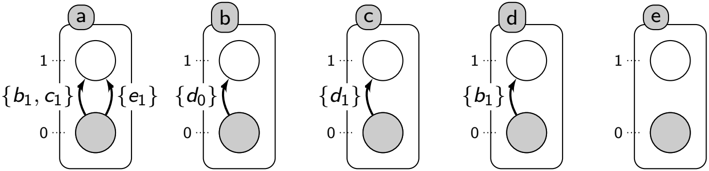
- $\Sigma = \{a, b, c, d, e\}$ set of automata

Automata Network (AN)



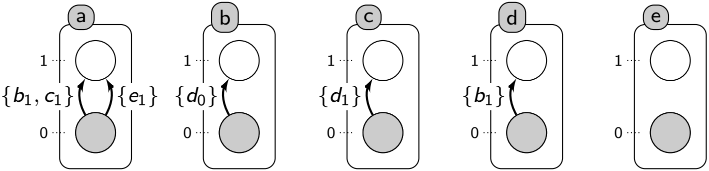
- $\Sigma = \{a, b, c, d, e\}$ set of automata
- States of automata:

Automata Network (AN)



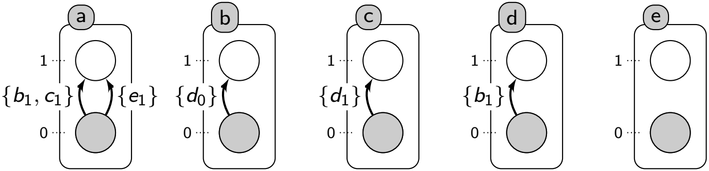
- $\Sigma = \{a, b, c, d, e\}$ set of automata
- States of automata:
  - Local state: a_0: automaton a is at state 0

Automata Network (AN)



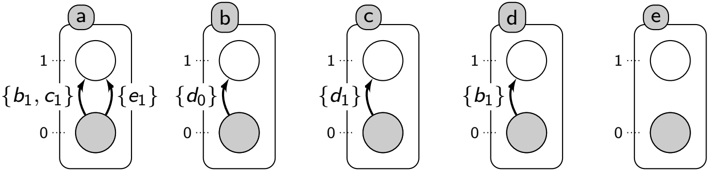
- $\Sigma = \{a, b, c, d, e\}$ set of automata
- States of automata:
  - Local state: a_0: automaton a is at state 0
  - Global state: $\langle a_0, b_0, c_0, d_0, e_0 \rangle$, the state of the whole system

Automata Network (AN)



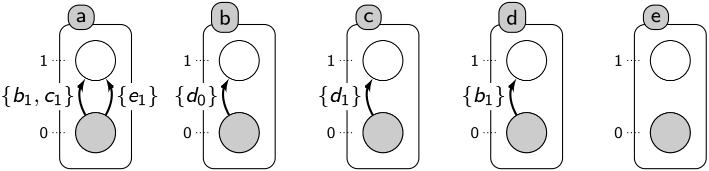
- $\Sigma = \{a, b, c, d, e\}$ set of automata
- States of automata:
  - Local state: a_0: automaton a is at state 0
  - Global state: $\langle a_0, b_0, c_0, d_0, e_0 \rangle$, the state of the whole system
  - Joint state: $\{a_0, b_0\}$ a part of the system state

Automata Network (AN)



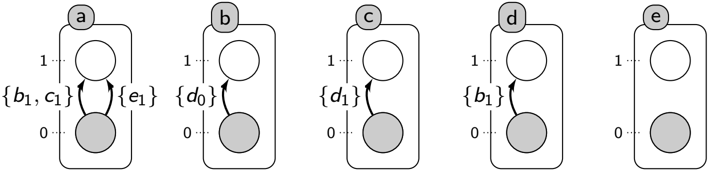
- $\Sigma = \{a, b, c, d, e\}$ set of automata
- States of automata:
  - Local state: a_0: automaton a is at state 0
  - Global state: $\langle a_0, b_0, c_0, d_0, e_0 \rangle$, the state of the whole system
  - Joint state: $\{a_0, b_0\}$ a part of the system state
- Transition:

Automata Network (AN)



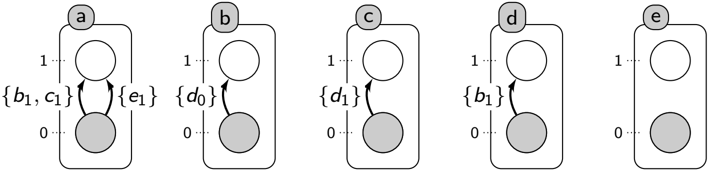
- $\Sigma = \{a, b, c, d, e\}$ set of automata
- States of automata:
  - Local state: a_0: automaton a is at state 0
  - Global state: $\langle a_0, b_0, c_0, d_0, e_0 \rangle$, the state of the whole system
  - Joint state: $\{a_0, b_0\}$ a part of the system state
- Transition:
  - $\{b_1, c_1\} \rightarrow a_0 \upharpoonright a_1$: a can transit from state 0 to state 1 when joint state $\{b_1, c_1\}$ is present

Automata Network (AN)



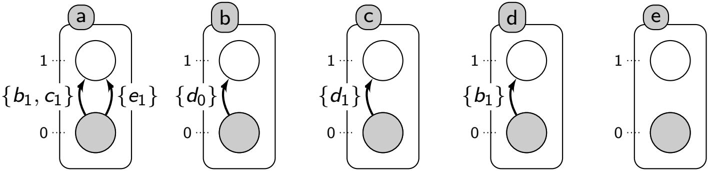
- $\Sigma = \{a, b, c, d, e\}$ set of automata
- States of automata:
  - Local state: a_0: automaton a is at state 0
  - Global state: $\langle a_0, b_0, c_0, d_0, e_0 \rangle$, the state of the whole system
  - Joint state: $\{a_0, b_0\}$ a part of the system state
- Transition:
  - $\{b_1, c_1\} \rightarrow a_0 \upharpoonright a_1$: a can transit from state 0 to state 1 when joint state $\{b_1, c_1\}$ is present
- Update scheme

Automata Network (AN)



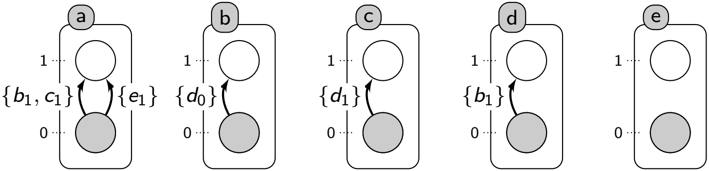
- $\Sigma = \{a, b, c, d, e\}$ set of automata
- States of automata:
  - Local state: a_0: automaton a is at state 0
  - Global state: $\langle a_0, b_0, c_0, d_0, e_0 \rangle$, the state of the whole system
  - Joint state: $\{a_0, b_0\}$ a part of the system state
- Transition:
  - $\{b_1, c_1\} \rightarrow a_0 \Rsh a_1$: a can transit from state 0 to state 1 when joint state $\{b_1, c_1\}$ is present
- Update scheme
  - Asynchronous Automata Network (AAN) [FPMR15]: at most one transition can be fired at each time point which conform to biological non-deterministic dynamics

Automata Network (AN)



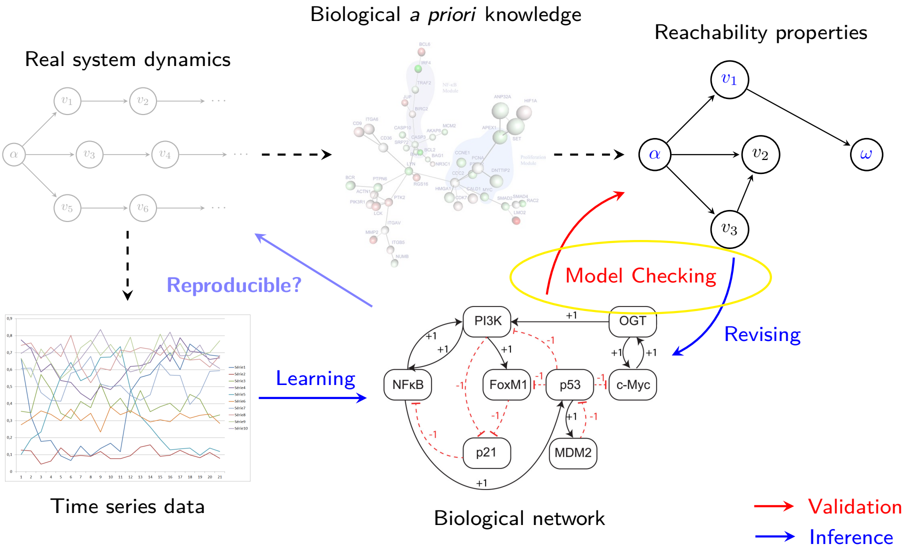
- $\Sigma = \{a, b, c, d, e\}$ set of automata with every automaton having a Boolean state
- States of automata:
  - Local state: a_0: automaton a is at state 0
  - Global state: $\langle a_0, b_0, c_0, d_0, e_0 \rangle$, the state of the whole system
  - Joint state: $\{a_0, b_0\}$ a part of the system state
- Transition:
  - $\{b_1, c_1\} \rightarrow a_0 \Rsh a_1$: a can transit from state 0 to state 1 when joint state $\{b_1, c_1\}$ is present
- Update scheme
  - Asynchronous Automata Network (AAN) [FPMR15]: at most one transition can be fired at each time point which conform to biological non-deterministic dynamics
- To simplify further reachability analysis, we limit the AAN to Boolean values
  $\rightarrow$ Asynchronous Binary Automata Network (ABAN)
  The notation of transitions is simplified to $\{b_1, c_1\} \rightarrow a_1$

Automata Network (AN)



- $\Sigma = \{a, b, c, d, e\}$ set of automata with every automaton having a Boolean state
- States of automata:
  - Local state: a_0: automaton a is at state 0
  - Global state: $\langle a_0, b_0, c_0, d_0, e_0 \rangle$, the state of the whole system
  - Joint state: $\{a_0, b_0\}$ a part of the system state
- Transition:
  - $\{b_1, c_1\} \rightarrow a_0 \Rsh a_1$: a can transit from state 0 to state 1 when joint state $\{b_1, c_1\}$ is present
- Update scheme
  - Asynchronous Automata Network (AAN) [FPMR15]: at most one transition can be fired at each time point which conform to biological non-deterministic dynamics
- To simplify further reachability analysis, we limit the AAN to Boolean values
  → Asynchronous Binary Automata Network (ABAN)
  The notation of transitions is simplified to $\{b_1, c_1\} \rightarrow a_1$
- To study reachability problem → Simplified Local Causality Graph (SLCG) based on Local Causality Graph (LCG) [PMR12]

Problematics



Biological *a priori* knowledge

Reachability properties

Real system dynamics

Reproducible?

Model Checking

Revising

Learning

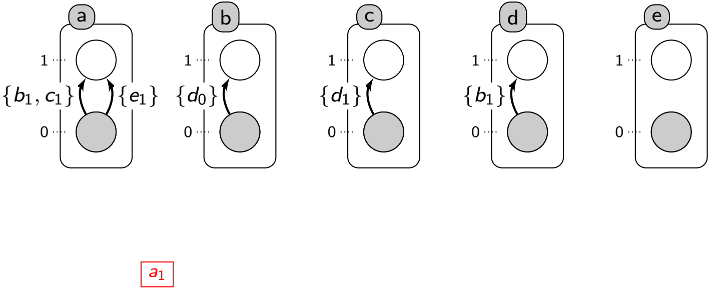
Time series data

Biological network

Validation

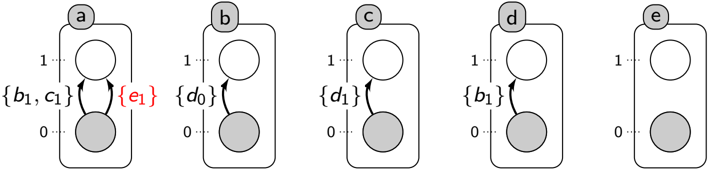
Inference

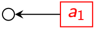
Simplified Local Causality Graph (SLCG)



Small circles stand for transition nodes, squares for state nodes

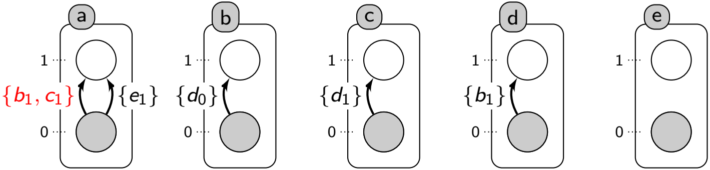
Simplified Local Causality Graph (SLCG)

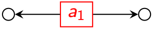




Small circles stand for transition nodes, squares for state nodes

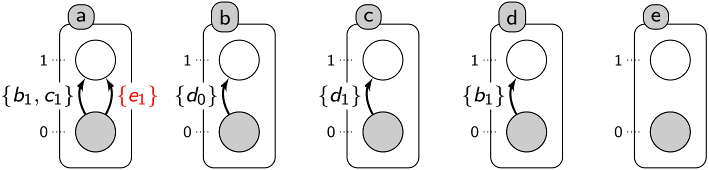
Simplified Local Causality Graph (SLCG)

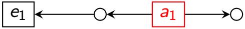




Small circles stand for transition nodes, squares for state nodes

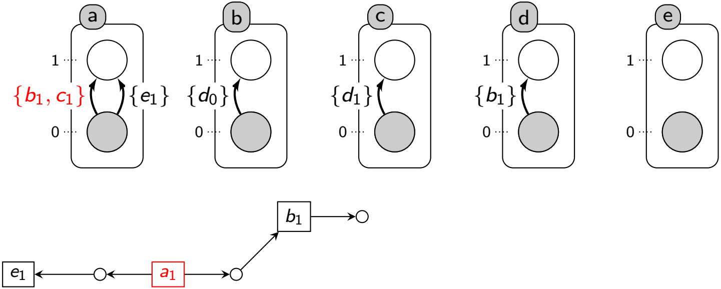
Simplified Local Causality Graph (SLCG)





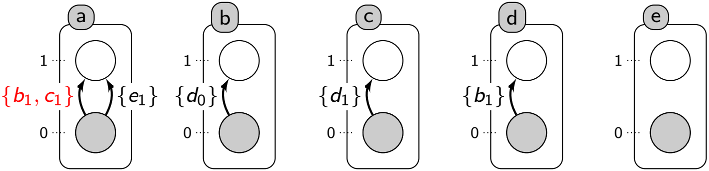
Small circles stand for transition nodes, squares for state nodes

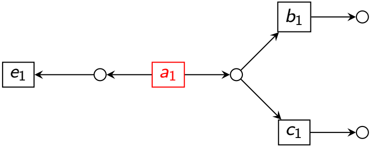
Simplified Local Causality Graph (SLCG)



Small circles stand for transition nodes, squares for state nodes

Simplified Local Causality Graph (SLCG)

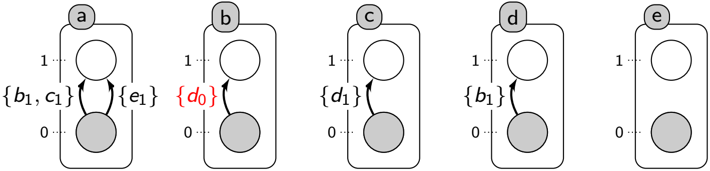


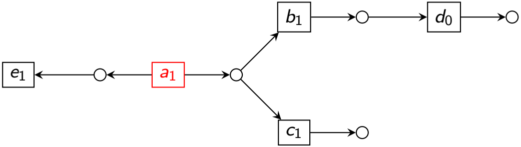


Pseudo-reachability r' is obtained *via* pure recursive causality reasoning

$$r'(a_1) = r'(e_1) \vee (r'(b_1) \wedge r'(c_1))$$

Simplified Local Causality Graph (SLCG)

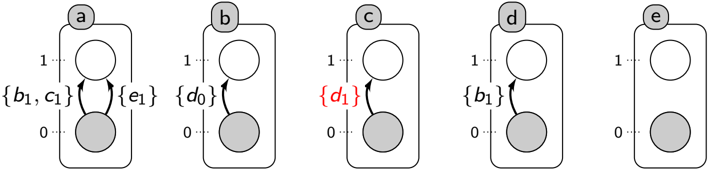


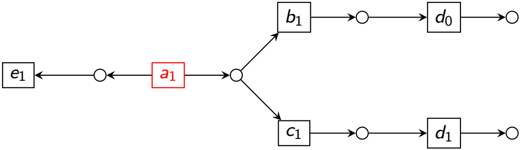


Pseudo-reachability r' is obtained *via* pure recursive causality reasoning

$$\begin{aligned} r'(a_1) &= r'(e_1) \lor (r'(b_1) \land r'(c_1)) \\ &= r'(d_0) \land r'(c_1) \end{aligned}$$

Simplified Local Causality Graph (SLCG)

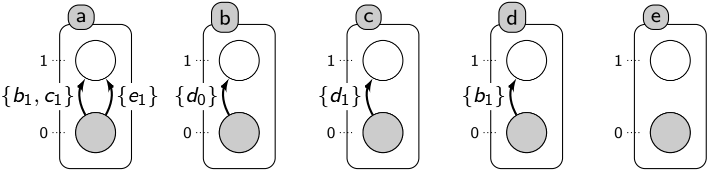


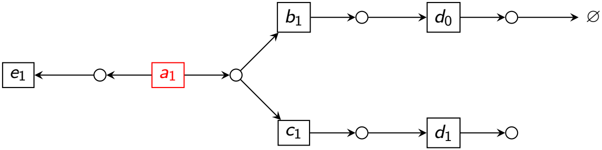


Pseudo-reachability r' is obtained *via* pure recursive causality reasoning

$$\begin{aligned} r'(a_1) &= r'(e_1) \vee (r'(b_1) \wedge r'(c_1)) \\ &= r'(d_0) \wedge r'(c_1) \\ &= r'(d_0) \wedge r'(d_1) \end{aligned}$$

Simplified Local Causality Graph (SLCG)

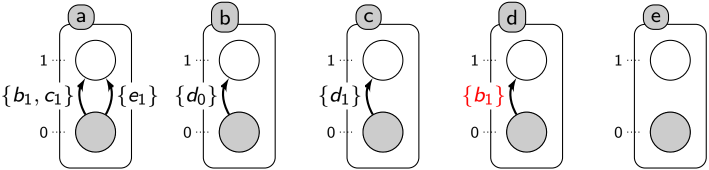


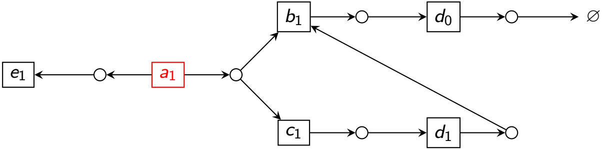


Pseudo-reachability r' is obtained *via* pure recursive causality reasoning

$$\begin{aligned} r'(a_1) &= r'(e_1) \vee (r'(b_1) \wedge r'(c_1)) \\ &= r'(d_0) \wedge r'(c_1) \\ &= r'(d_0) \wedge r'(d_1) \\ &= r'(d_1) \end{aligned}$$

Simplified Local Causality Graph (SLCG)



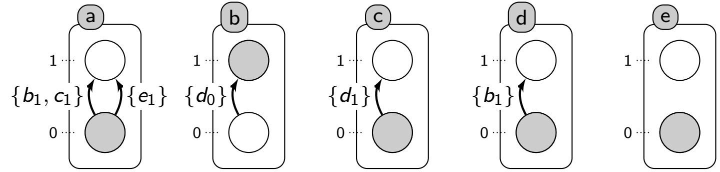


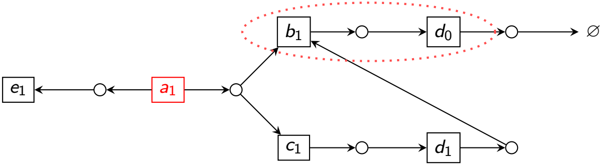
Pseudo-reachability r' is obtained *via* pure recursive causality reasoning

$$
\begin{aligned}
r'(a_1) &= r'(e_1) \vee (r'(b_1) \wedge r'(c_1)) \\
&= r'(d_0) \wedge r'(c_1) \\
&= r'(d_0) \wedge r'(d_1) \\
&= r'(d_1) \\
&= r'(b_1) = r'(d_0) = \textbf{True}
\end{aligned}
$$

$r'(d_0) = $ **True** because d_0 is at initial state

Simplified Local Causality Graph (SLCG)



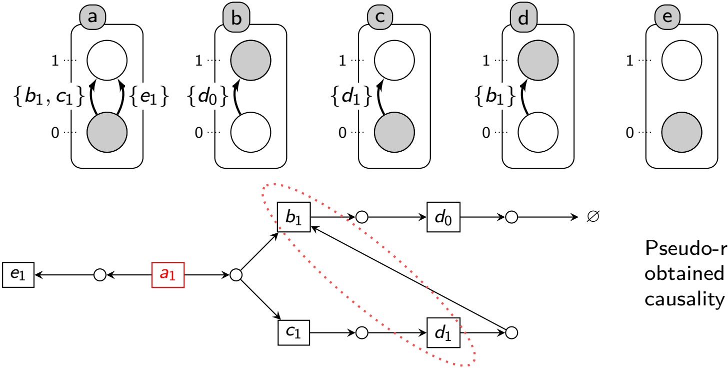


Pseudo-reachability r' is obtained *via* pure recursive causality reasoning

$$\begin{aligned} r'(a_1) &= r'(e_1) \vee (r'(b_1) \wedge r'(c_1)) \\ &= r'(d_0) \wedge r'(c_1) \\ &= r'(d_0) \wedge r'(d_1) \\ &= r'(d_1) \\ &= r'(b_1) = r'(d_0) = \textbf{True} \end{aligned}$$

$r'(d_0) = \textbf{True}$ because d_0 is at initial state

Simplified Local Causality Graph (SLCG)

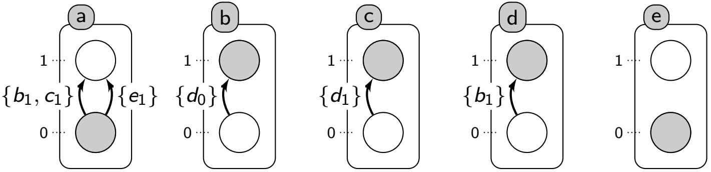


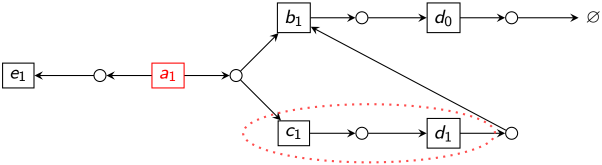
Pseudo-reachability r' is obtained *via* pure recursive causality reasoning

$$\begin{aligned} r'(a_1) &= r'(e_1) \vee (r'(b_1) \wedge r'(c_1)) \\ &= r'(d_0) \wedge r'(c_1) \\ &= r'(d_0) \wedge r'(d_1) \\ &= r'(d_1) \\ &= r'(b_1) = r'(d_0) = \textbf{True} \end{aligned}$$

$r'(d_0) = \textbf{True}$ because d_0 is at initial state

Simplified Local Causality Graph (SLCG)



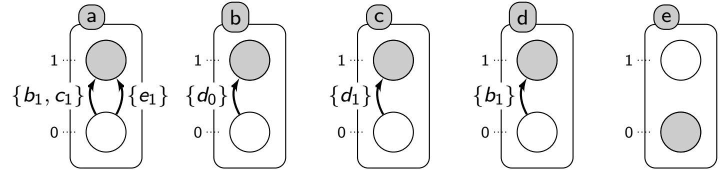


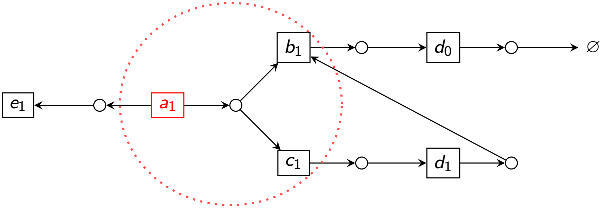
Pseudo-reachability r' is obtained *via* pure recursive causality reasoning

$$
\begin{aligned}
r'(a_1) &= r'(e_1) \vee (r'(b_1) \wedge r'(c_1)) \\
&= r'(d_0) \wedge r'(c_1) \\
&= r'(d_0) \wedge r'(d_1) \\
&= r'(d_1) \\
&= r'(b_1) = r'(d_0) = \textbf{True}
\end{aligned}
$$

$r'(d_0) =$ **True** because d_0 is at initial state

Simplified Local Causality Graph (SLCG)



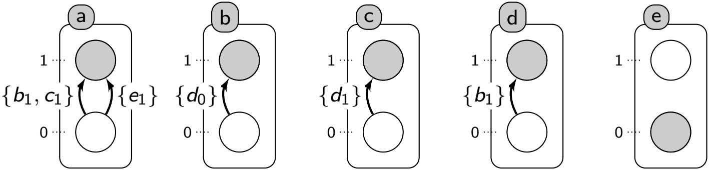


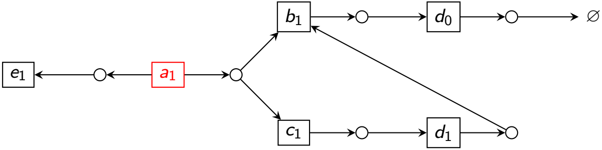
Pseudo-reachability r' is obtained *via* pure recursive causality reasoning

$$
\begin{aligned}
r'(a_1) &= r'(e_1) \vee (r'(b_1) \wedge r'(c_1)) \\
&= r'(d_0) \wedge r'(c_1) \\
&= r'(d_0) \wedge r'(d_1) \\
&= r'(d_1) \\
&= r'(b_1) = r'(d_0) = \textbf{True}
\end{aligned}
$$

$r'(d_0) = $ **True** because d_0 is at initial state

Simplified Local Causality Graph (SLCG)





Pseudo-reachability r' is obtained *via* pure recursive causality reasoning

$$\begin{aligned} r'(a_1) &= r'(e_1) \vee (r'(b_1) \wedge r'(c_1)) \\ &= r'(d_0) \wedge r'(c_1) \\ &= r'(d_0) \wedge r'(d_1) \\ &= r'(d_1) \\ &= r'(b_1) = r'(d_0) = \textbf{True} \end{aligned}$$

$r'(d_0) = \textbf{True}$ because d_0 is at initial state

$r'(a_i)$ is **not equivalent** to the reachability of a_1 because the state space is not fully explored

Counterexample of SLCG

- SLCG is **exact** for unreachability

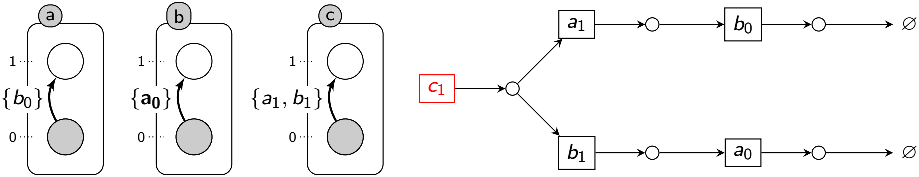
Counterexample of SLCG

- SLCG is exact for unreachability
- SLCG is exact for reachability when it does not contain self-dependent structure:
  - different state nodes of the same automaton in different branches

Counterexample of SLCG

- SLCG is exact for unreachability
- SLCG is exact for reachability when it does not contain self-dependent structure:
  - different state nodes of the same automaton in different branches

Given initial state: $\langle a_0, b_0, c_0 \rangle$, consider the reachability of c_1:

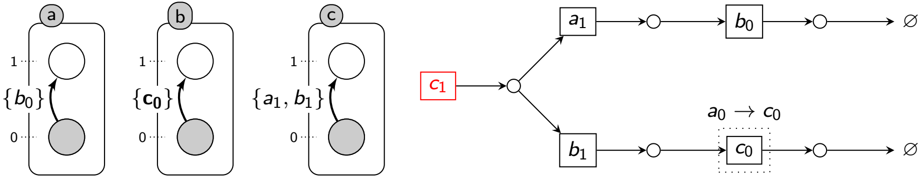


When a reaches level 1, b_1 is no longer reachable as the condition of $\{a_0\} \rightarrow b_1$ is not satisfied. $\rightarrow$ Reaching a_1 disables the reachability of b_1, *vice versa*, $\{a_1, b_1\}$ is not reachable, *i.e.* c_1 is unreachable

Counterexample of SLCG

- SLCG is exact for unreachability
- SLCG is exact for reachability when it does not contain self-dependent structure:
    - different state nodes of the same automaton in different branches

Given initial state: $\langle a_0, b_0, c_0 \rangle$, consider the reachability of c_1:

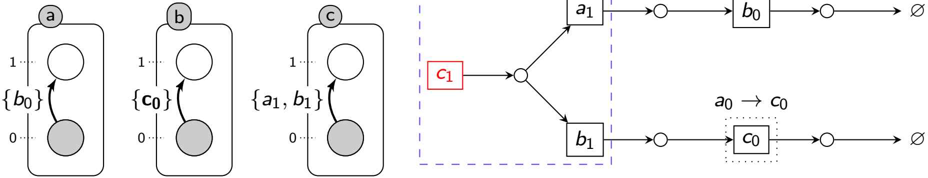


b_1 is not blocked by a, c_1 is reachable *via* $a_1 :: b_1 :: c_1$

Counterexample of SLCG

- SLCG is exact for unreachability
- SLCG is exact for reachability when it does not contain self-dependent structure:
  - different state nodes of the same automaton in different branches

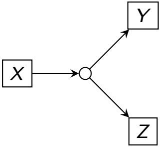
Given initial state: $\langle a_0, b_0, c_0\rangle$, consider the reachability of c_1:



b_1 is not blocked by a, c_1 is reachable *via* $a_1 :: b_1 :: c_1$

SLCG does not show orders of local states

$\Longrightarrow$ How to deal with this structure?



Solution of Inconclusive Structure

Complete search on global states

Solution of Inconclusive Structure

Complete search on global states → State space explosion problem ☹

Solution of Inconclusive Structure

Complete search on global states → State space explosion problem ☹

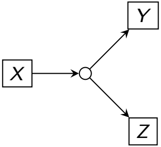
$\Longrightarrow$ Partial search

Solution of Inconclusive Structure

Complete search on global states → State space explosion problem ☹

$\Longrightarrow$ Partial search

$\Longrightarrow$ Apply heuristic search on branches to find a **sequence** of local states in the form $Z :: Y :: X$ in order to avoid the inconclusiveness of the reachability of X

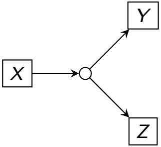


Solution of Inconclusive Structure

Complete search on global states → State space explosion problem ☹

$\Longrightarrow$ Partial search

$\Longrightarrow$ Apply heuristic search on branches to find a **sequence** of local states in the form $Z :: Y :: X$ in order to avoid the inconclusiveness of the reachability of X



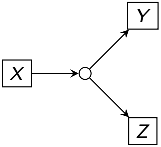
- PermReach: searching all permutations in branches

Solution of Inconclusive Structure

Complete search on global states → State space explosion problem ☹

$\Longrightarrow$ Partial search

$\Longrightarrow$ Apply heuristic search on branches to find a **sequence** of local states in the form $Z :: Y :: X$ in order to avoid the inconclusiveness of the reachability of X



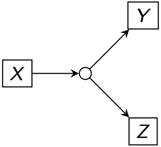
- PermReach: searching all permutations in branches
- ASPReach: searching all possible sequences using Answer Set Programming [Bar03]

Solution of Inconclusive Structure

Complete search on global states → State space explosion problem ☹

$\Longrightarrow$ Partial search

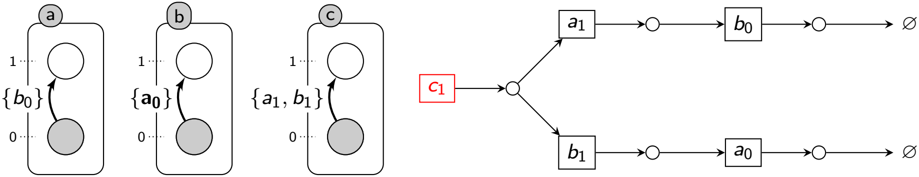
$\Longrightarrow$ Apply heuristic search on branches to find a **sequence** of local states in the form $Z :: Y :: X$ in order to avoid the inconclusiveness of the reachability of X



- PermReach: searching all permutations in branches
- ASPReach: searching all possible sequences using Answer Set Programming [Bar03]

	PermReach	ASPReach [CRM^{+}18]
Method	Search all the permutations of branches	Search all the possible order of branches
Runtime	$+$	$-$
Conclusiveness	$-$	$+$

Solve the Counterexample Using ASPReach



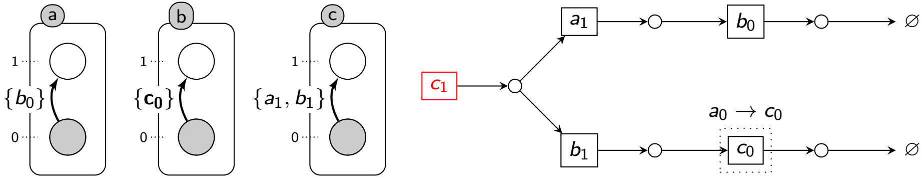
Notation: $a \rhd b$ means a appears in the sequence before b

Order constraints in SLCG ⇒ $b_0 \rhd a_1 \rhd c_1$ and $a_0 \rhd b_1 \rhd c_1$

Additional constraint ⇒ $a_1 \rhd b_1$ and $b_1 \rhd a_1$

Contradiction in order, c_1 unreachable

Solve the Counterexample Using ASPReach



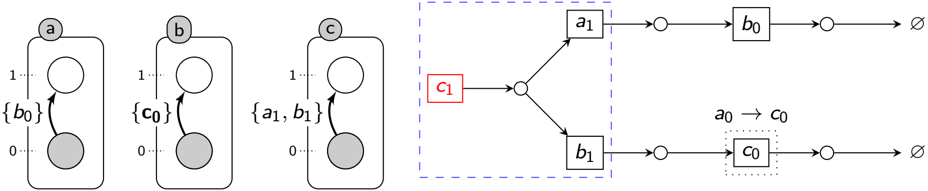
Notation: $a \rhd b$ means a appears in the sequence before b

Order constraints in SLCG $\Rightarrow$ $b_0 \rhd a_1 \rhd c_1$ and $c_0 \rhd b_1 \rhd c_1$

Additional constraint $\Rightarrow$ $a_1 \rhd b_1$

The only admissible order is $a_1 :: b_1 :: c_1$

Solve the Counterexample Using ASPReach



Notation: $a \triangleright b$ means a appears in the sequence before b

Order constraints in SLCG $\Rightarrow$ $b_0 \triangleright a_1 \triangleright c_1$ and $c_0 \triangleright b_1 \triangleright c_1$

Additional constraint $\Rightarrow$ $a_1 \triangleright b_1$

The only admissible order is $a_1 :: b_1 :: c_1$

$\Longrightarrow$ Problematic structure solved, reachable if sequences found, unreachable if not found

Benchmarks: on Biological Examples

- Traditional model checkers: Mole, NuSMV → **memory-out**, not listed in the benchmarks

Benchmarks: on Biological Examples

- Traditional model checkers: Mole, NuSMV → **memory-out**, not listed in the benchmarks
- Pure static analyzer: Pint [FPMR15]

Benchmarks: on Biological Examples

- Traditional model checkers: Mole, NuSMV → **memory-out**, not listed in the benchmarks
- Pure static analyzer: Pint [FPMR15]
- Small example: λ-phage, 4 components

Model	λ-phage		
Inputs	4	Outputs	4
Total tests	$2^4 \times 4 = 64$		
Analyzer	Pint	**PermReach**	**ASPReach**
Reachable	36(56%)	38(59%)	
Unreachable	26(41%)		
Inconclusive	**2(3%)**	**0(0%)**	
Total time	< 1s		

Benchmarks: on Biological Examples

- Traditional model checkers: Mole, NuSMV → **memory-out**, not listed in the benchmarks
- Pure static analyzer: Pint [FPMR15]
- Small example: λ-phage, 4 components
- Big examples: TCR (T-Cell Receptor, 95 components)

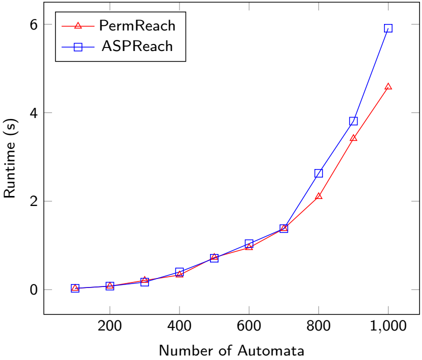
Model	λ-phage		
Inputs	4	Outputs	4
Total tests	$2^4 \times 4 = 64$		
Analyzer	Pint	**PermReach**	**ASPReach**
Reachable	36(56%)	38(59%)	
Unreachable	26(41%)		
Inconclusive	**2(3%)**	**0(0%)**	
Total time	< 1s		
Model	TCR		
Inputs	3	Outputs	5
Total tests	$2^3 \times 5 = 40$		
Analyzer	Pint	**PermReach**	**ASPReach**
Reachable	16(40%)		
Unreachable	24(60%)		
Inconclusive	**0(0%)**		
Total time	7s	0.85s	40s

Benchmarks: on Biological Examples

- Traditional model checkers: Mole, NuSMV → **memory-out**, not listed in the benchmarks
- Pure static analyzer: Pint [FPMR15]
- Small example: λ-phage, 4 components
- Big examples: TCR (T-Cell Receptor, 95 components)
- EGFR (Epidermal Growth Factor Receptor, 104 components)

Model	λ-phage		
Inputs	4	Outputs	4
Total tests	$2^4 \times 4 = 64$		
Analyzer	Pint	**PermReach**	**ASPReach**
Reachable	36(56%)	38(59%)	
Unreachable	26(41%)		
Inconclusive	**2(3%)**	**0(0%)**	
Total time	< 1s		
Model	TCR		
Inputs	3	Outputs	5
Total tests	$2^3 \times 5 = 40$		
Analyzer	Pint	**PermReach**	**ASPReach**
Reachable	16(40%)		
Unreachable	24(60%)		
Inconclusive	**0(0%)**		
Total time	7s	0.85s	40s
Model	EGFR		
Inputs	13	Outputs	12
Total tests	$2^{13} \times 12 = 98,304$		
Analyzer	Pint	**PermReach**	**ASPReach**
Reachable	64,282(65.4%)	74,268(75.5%)	
Unreachable	24,036(24.5%)		
Inconclusive	**9,986(10.1%)**	**0(0%)**	
Total time	**9h50min**	**15min31s**	**3h46min**

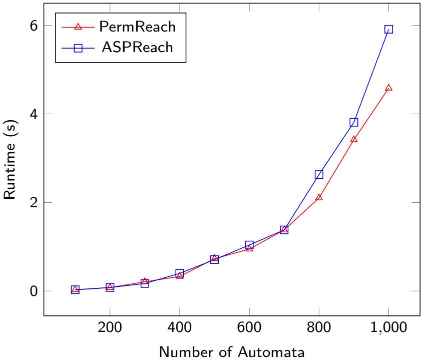
Benchmarks: on Random Examples

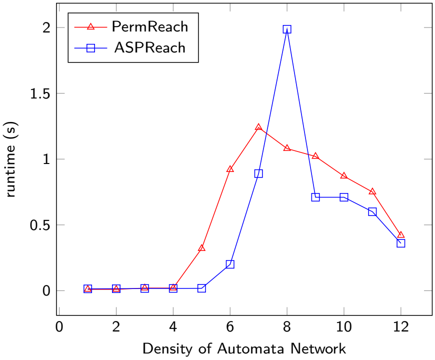


$$\text{density} = \frac{\#\text{transitions}}{\#\text{automata}}$$

Fixing $\text{density} = 3$

Benchmarks: on Random Examples



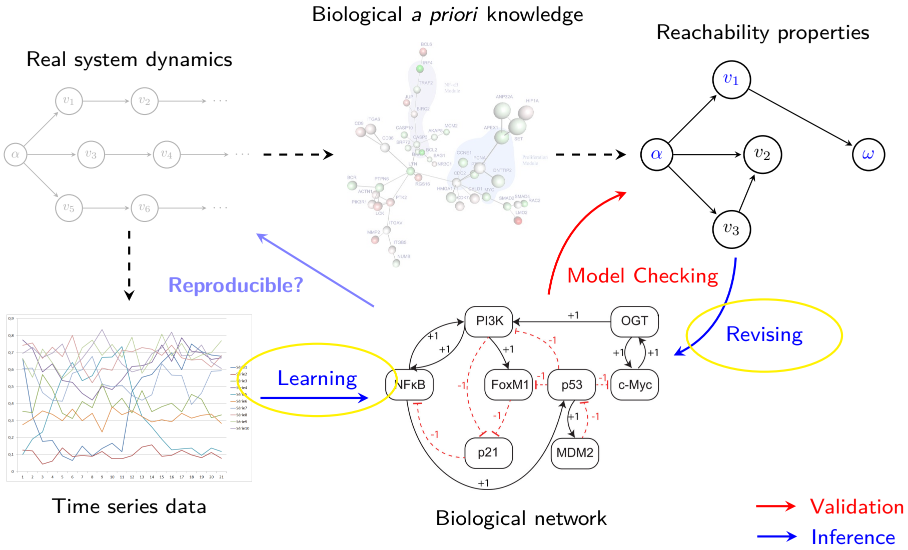


$$\text{density} = \frac{\#\text{transitions}}{\#\text{automata}}$$

Fixing $\text{density} = 3$

Fixing $\#\text{automata} = 20$

Problematics



Real system dynamics

Biological *a priori* knowledge

Reachability properties

Reproducible?

Model Checking

Learning

Revising

Time series data

Biological network

→ Validation

→ Inference

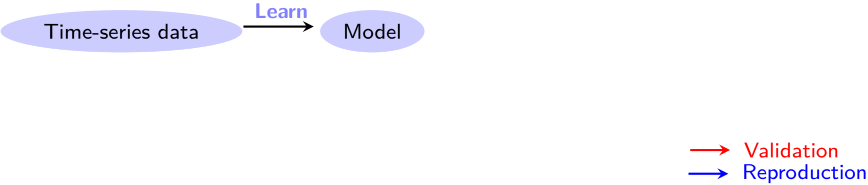
Learning and Revising Models

Time-series data

→ Validation

→ Reproduction

Learning and Revising Models



Time-series data

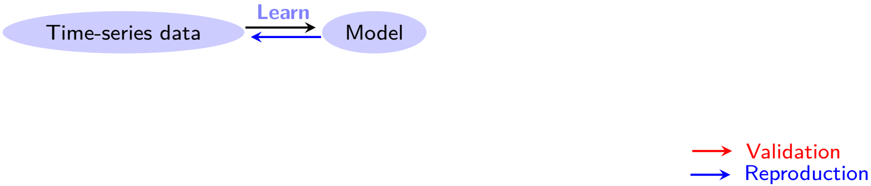
Learn

Model

→ Validation

→ Reproduction

Learning and Revising Models



Time-series data

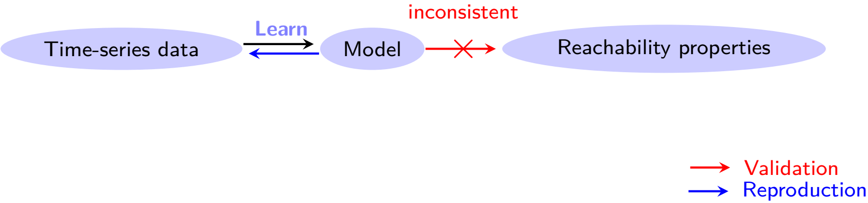
Learn

Model

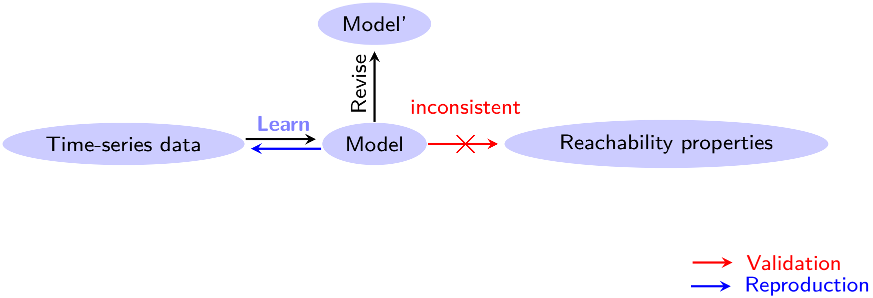
Validation

Reproduction

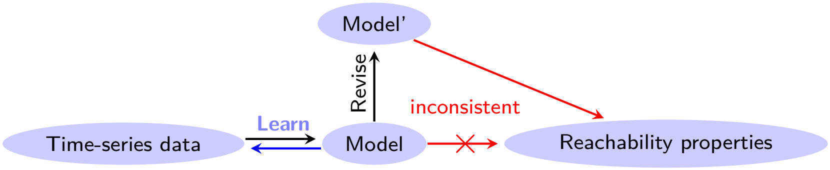
Learning and Revising Models

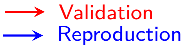


Learning and Revising Models

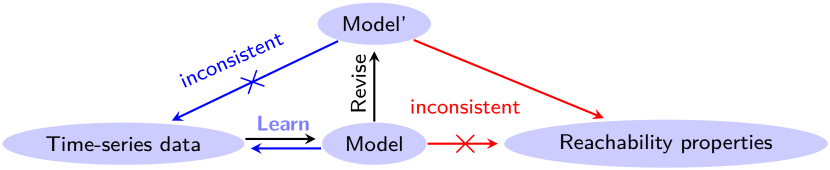


Learning and Revising Models





Learning and Revising Models



Model'

inconsistent

Revise

inconsistent

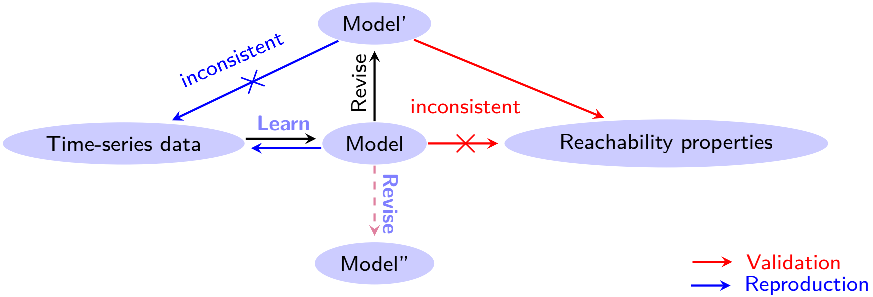
Learn

Time-series data

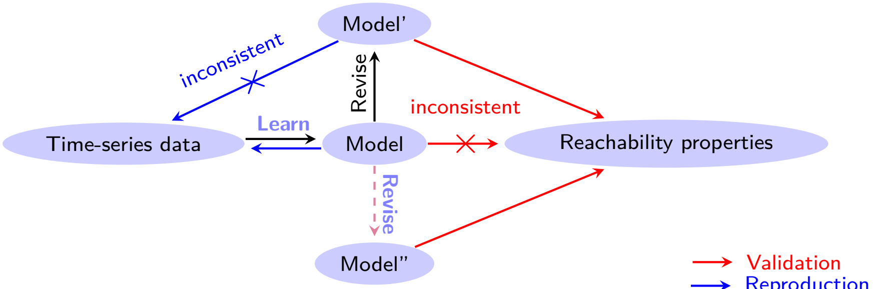
Model

Reachability properties

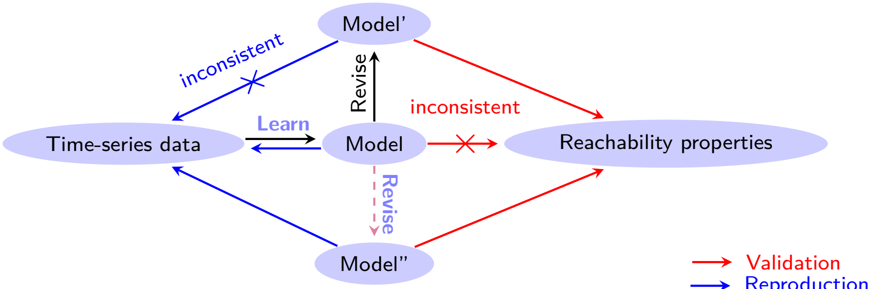
Learning and Revising Models



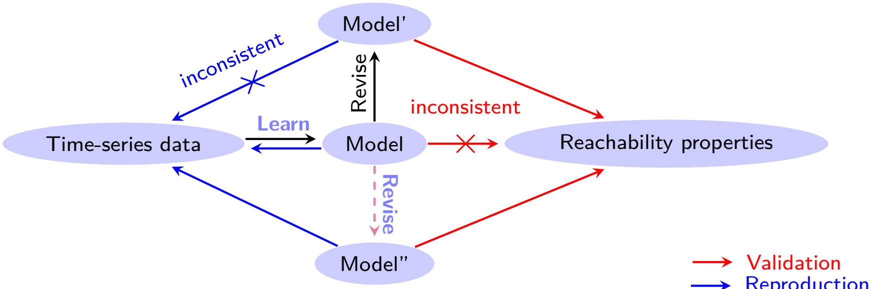
Learning and Revising Models



Learning and Revising Models

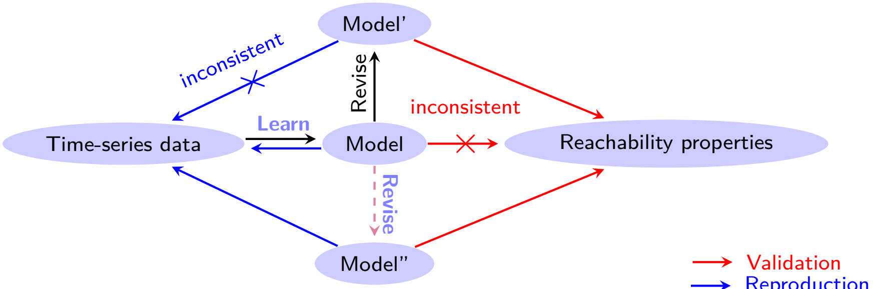


Learning and Revising Models



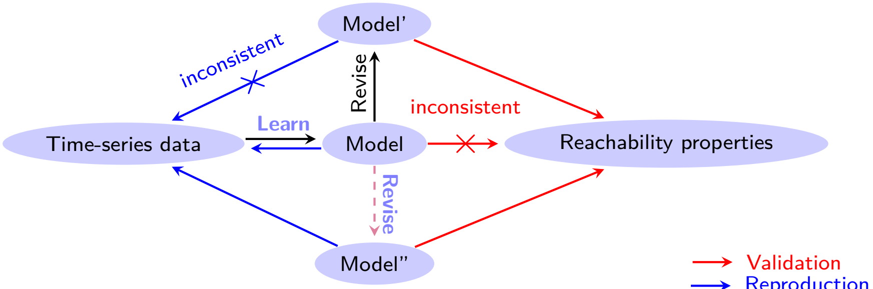
- CRAC: Completion *via* Reachability And Correlations

Learning and Revising Models



- CRAC: Completion *via* Reachability And Correlations
- M2RIT: Model Revision *via* Reachability and Interpretation Transitions

Learning and Revising Models



- CRAC: Completion *via* Reachability And Correlations
- M2RIT: Model Revision *via* Reachability and Interpretation Transitions

	CRAC	M2RIT
Learning phase	Correlation Coefficients	Learning from Interpretation Transitions (LFIT) [RFM$^+$18]
Revising phase	Reachability+ candidate transitions	Reachability+ time-series data

Learning Phase

CRAC: using correlation coefficient $r_{x,y} = \frac{\mathrm{cov}(x,y)}{\sigma_x \sigma_y}$

Learning Phase

CRAC: using correlation coefficient $r_{x,y} = \frac{\mathrm{cov}(x,y)}{\sigma_x \sigma_y} = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{N}(y_i - \bar{y})^2}}$

Learning Phase

CRAC: using correlation coefficient $r_{x,y} = \frac{\mathrm{cov}(x,y)}{\sigma_x \sigma_y} = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i-\bar{x})^2}\sqrt{\sum_{i=1}^{N}(y_i-\bar{y})^2}}$

1.

t	0	1	2	3
a	2.01	2.51	1.97	1.17
b	0.74	0.87	0.78	0.33

Learning Phase

CRAC: using correlation coefficient $r_{x,y} = \frac{\mathrm{cov}(x,y)}{\sigma_x \sigma_y} = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{N}(y_i - \bar{y})^2}}$

1.

t	0	1	2	3
a	2.01	2.51	1.97	1.17
b	0.74	0.87	0.78	0.33

2.

t	1	2	3
Δa	0.5	-0.54	-0.8
Δb	0.13	-0.09	-0.45

Learning Phase

CRAC: using correlation coefficient $r_{x,y} = \frac{\mathrm{cov}(x,y)}{\sigma_x \sigma_y} = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{N}(y_i - \bar{y})^2}}$

1.

t	0	1	2	3
a	2.01	2.51	1.97	1.17
b	0.74	0.87	0.78	0.33

2.

t	1	2	3
Δa	0.5	-0.54	-0.8
Δb	0.13	-0.09	-0.45

3. $$r' = \begin{array}{c c} & \begin{array}{cc} \Delta a & \Delta b \end{array} \\ \begin{array}{c} a \\ b \end{array} & \left[\begin{array}{cc} \text{N/A} & 0.09 \\ 0.65 & \text{N/A} \end{array} \right] \end{array}$$

Learning Phase

CRAC: using correlation coefficient $r_{x,y} = \frac{\text{cov}(x,y)}{\sigma_x \sigma_y} = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{N}(y_i - \bar{y})^2}}$

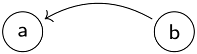
1.

t	0	1	2	3
a	2.01	2.51	1.97	1.17
b	0.74	0.87	0.78	0.33

2.

t	1	2	3
Δa	0.5	-0.54	-0.8
Δb	0.13	-0.09	-0.45

3. $$r' = \begin{array}{ccc} & \Delta a & \Delta b \\ a & \text{N/A} & 0.09 \\ b & 0.65 & \text{N/A} \end{array}$$

4. 

Learning Phase

CRAC: using correlation coefficient $r_{x,y} = \frac{\mathrm{cov}(x,y)}{\sigma_x \sigma_y} = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{N}(y_i - \bar{y})^2}}$

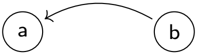
1.

t	0	1	2	3
a	2.01	2.51	1.97	1.17
b	0.74	0.87	0.78	0.33

2.

t	1	2	3
Δa	0.5	-0.54	-0.8
Δb	0.13	-0.09	-0.45

3. $$r' = \begin{array}{c|cc} & \Delta a & \Delta b \\ \hline a & \text{N/A} & 0.09 \\ b & 0.65 & \text{N/A} \end{array}$$

4. 

M2RIT: using LFIT algorithm on discretized time-series data

Global transitions obtained from time-series data

$$\langle a_1, b_0, c_1\rangle(t = T_1) \rightarrow \langle a_1, b_1, c_1\rangle(t = T_1 + 1)$$
$$\langle a_1, b_0, c_0\rangle(t = T_2) \rightarrow \langle a_1, b_1, c_0\rangle(t = T_2 + 1)$$
$$\langle a_0, b_0, c_0\rangle(t = T_3) \rightarrow \langle a_0, b_0, c_1\rangle(t = T_3 + 1)$$

classified into

$$\Longrightarrow$$

partial transitions

$$\{a_1\} \rightarrow b_1$$

Learning Phase

CRAC: using correlation coefficient $r_{x,y} = \frac{\mathrm{cov}(x,y)}{\sigma_x \sigma_y} = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{N}(y_i - \bar{y})^2}}$

1.

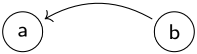
t	0	1	2	3
a	2.01	2.51	1.97	1.17
b	0.74	0.87	0.78	0.33

2.

t	1	2	3
Δa	0.5	-0.54	-0.8
Δb	0.13	-0.09	-0.45

3. $r' = \begin{array}{cc} & \begin{array}{cc} \Delta a & \Delta b \end{array} \\ \begin{array}{c} a \\ b \end{array} & \left[\begin{array}{cc} \mathrm{N/A} & 0.09 \\ 0.65 & \mathrm{N/A} \end{array}\right] \end{array}$

4.



M2RIT: using LFIT algorithm on discretized time-series data

Global transitions obtained from time-series data classified into partial transitions

$\langle a_1, b_0, c_1 \rangle (t = T_1) \rightarrow \langle a_1, b_1, c_1 \rangle (t = T_1 + 1)$

$\langle a_1, b_0, c_0 \rangle (t = T_2) \rightarrow \langle a_1, b_1, c_0 \rangle (t = T_2 + 1) \quad \Longrightarrow \quad \{a_1\} \rightarrow b_1$

$\langle a_0, b_0, c_0 \rangle (t = T_3) \rightarrow \langle a_0, b_0, c_1 \rangle (t = T_3 + 1)$

The classification is sensitive to input → revise the result

Revising Phase

Methodology:

	Reachable	Unreachable
Knowledge	R_K	U_K

Revising Phase

Methodology:

	Reachable	Unreachable
Knowledge	R_K	U_K
Inferred model	R_I	U_I

Revising Phase

Methodology:

	Reachable	Unreachable
Knowledge	R_K	U_K
Inferred model	R_I	U_I
Inconsistency	$R'_K = R_K \cap U_I$	$U'_K = R_I \cap U_K$

Revising Phase

Methodology:

	Reachable	Unreachable
Knowledge	R_K	U_K
Inferred model	R_I	U_I
Inconsistency	$R'_K = R_K \cap U_I$	$U'_K = R_I \cap U_K$
Keep consistent with	U_K	R_K

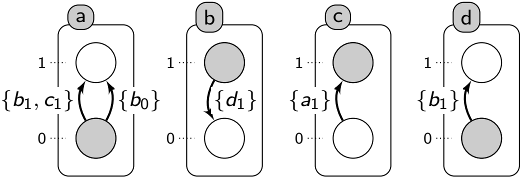
Revising Phase

Methodology:

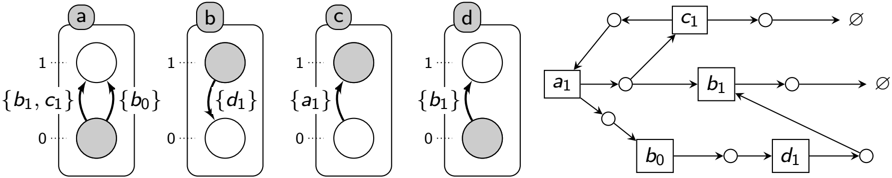
	Reachable	Unreachable
Knowledge	R_K	U_K
Inferred model	R_I	U_I
Inconsistency	$R'_K = R_K \cap U_I$	$U'_K = R_I \cap U_K$
Keep consistent with	U_K	R_K

- Generic solution: computes all the revisions → complete but costly
  - Ensure unreachability by inhibiting elements in corresponding **cut set**
  - Ensure reachability by guaranteeing elements in corresponding **completion set**
- Heuristics: explores the saturated SLCG, aiming at revising transitions one by one → memory-saving, fast but not complete

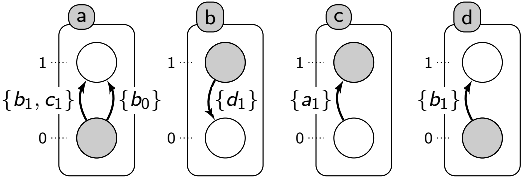
Cut Set

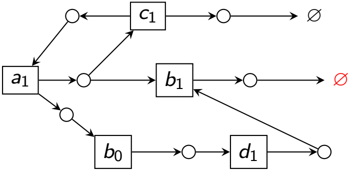


Cut Set



Cut Set

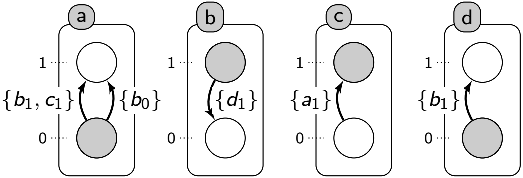


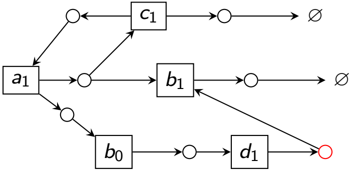


Node	Rank	$\mathbb{V}$
$\varnothing$ (of b_1)	1	$\varnothing$
b_1	2	$\{\{b_1\}\}$
$\{b_1\} \to d_1$	3	$\{\{b_1\}\}$
d_1	4	$\{\{b_1\}, \{d_1\}\}$
$\{d_1\} \to b_0$	5	$\{\{b_1\}, \{d_1\}\}$
b_0	6	$\{\{b_0\}, \{b_1\}, \{d_1\}\}$
$\{b_0\} \to a_1$	7	$\{\{b_0\}, \{b_1\}, \{d_1\}\}$
$\varnothing$ (of c_1)	8	$\varnothing$
c_1	9	$\{\{c_1\}\}$
$\{b_1, c_1\} \to a_1$	9	$\{\{b_1\}, \{c_1\}\}$
a_1	9	$\{\{a_1\}, \{b_1\}, \{b_0, c_1\}, \{c_1, d_1\}\}$
$\{a_1\} \to c_1$	9	$\{\{a_1\}, \{b_1\}, \{b_0, c_1\}, \{c_1, d_1\}\}$

Rank: topological numbering of the nodes in SLCG, nodes with higher rank cannot be successor of nodes with lower rank.

Cut Set

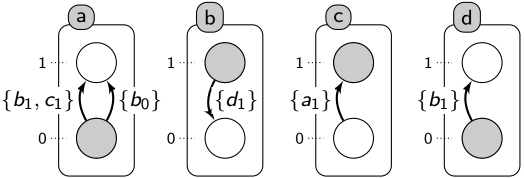


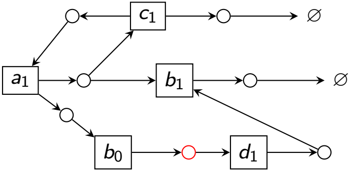


Node	Rank	$\mathbb{V}$
$\varnothing$ (of b_1)	1	$\varnothing$
b_1	2	$\{\{b_1\}\}$
$\{b_1\} \to d_1$	3	$\{\{b_1\}\}$
d_1	4	$\{\{b_1\}, \{d_1\}\}$
$\{d_1\} \to b_0$	5	$\{\{b_1\}, \{d_1\}\}$
b_0	6	$\{\{b_0\}, \{b_1\}, \{d_1\}\}$
$\{b_0\} \to a_1$	7	$\{\{b_0\}, \{b_1\}, \{d_1\}\}$
$\varnothing$ (of c_1)	8	$\varnothing$
c_1	9	$\{\{c_1\}\}$
$\{b_1, c_1\} \to a_1$	9	$\{\{b_1\}, \{c_1\}\}$
a_1	9	$\{\{a_1\}, \{b_1\}, \{b_0, c_1\}, \{c_1, d_1\}\}$
$\{a_1\} \to c_1$	9	$\{\{a_1\}, \{b_1\}, \{b_0, c_1\}, \{c_1, d_1\}\}$

Rank: topological numbering of the nodes in SLCG, nodes with higher rank cannot be successor of nodes with lower rank.

Cut Set

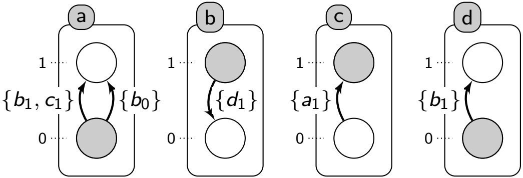


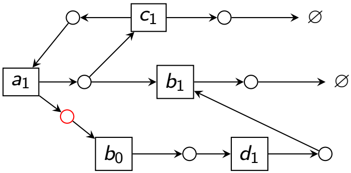


Node	Rank	$\mathbb{V}$
$\varnothing$ (of b_1)	1	$\varnothing$
b_1	2	$\{\{b_1\}\}$
$\{b_1\} \rightarrow d_1$	3	$\{\{b_1\}\}$
d_1	4	$\{\{b_1\}, \{d_1\}\}$
$\{d_1\} \rightarrow b_0$	5	$\{\{b_1\}, \{d_1\}\}$
b_0	6	$\{\{b_0\}, \{b_1\}, \{d_1\}\}$
$\{b_0\} \rightarrow a_1$	7	$\{\{b_0\}, \{b_1\}, \{d_1\}\}$
$\varnothing$ (of c_1)	8	$\varnothing$
c_1	9	$\{\{c_1\}\}$
$\{b_1, c_1\} \rightarrow a_1$	9	$\{\{b_1\}, \{c_1\}\}$
a_1	9	$\{\{a_1\}, \{b_1\}, \{b_0, c_1\}, \{c_1, d_1\}\}$
$\{a_1\} \rightarrow c_1$	9	$\{\{a_1\}, \{b_1\}, \{b_0, c_1\}, \{c_1, d_1\}\}$

Rank: topological numbering of the nodes in SLCG, nodes with higher rank cannot be successor of nodes with lower rank.

Cut Set

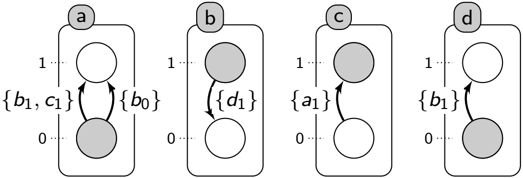


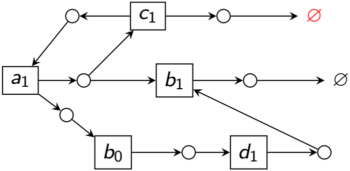


Node	Rank	$\mathbb{V}$
$\emptyset$ (of b_1)	1	$\emptyset$
b_1	2	$\{\{b_1\}\}$
$\{b_1\} \rightarrow d_1$	3	$\{\{b_1\}\}$
d_1	4	$\{\{b_1\}, \{d_1\}\}$
$\{d_1\} \rightarrow b_0$	5	$\{\{b_1\}, \{d_1\}\}$
b_0	6	$\{\{b_0\}, \{b_1\}, \{d_1\}\}$
$\{b_0\} \rightarrow a_1$	7	$\{\{b_0\}, \{b_1\}, \{d_1\}\}$
$\emptyset$ (of c_1)	8	$\emptyset$
c_1	9	$\{\{c_1\}\}$
$\{b_1, c_1\} \rightarrow a_1$	9	$\{\{b_1\}, \{c_1\}\}$
a_1	9	$\{\{a_1\}, \{b_1\}, \{b_0, c_1\}, \{c_1, d_1\}\}$
$\{a_1\} \rightarrow c_1$	9	$\{\{a_1\}, \{b_1\}, \{b_0, c_1\}, \{c_1, d_1\}\}$

Rank: topological numbering of the nodes in SLCG, nodes with higher rank cannot be successor of nodes with lower rank.

Cut Set

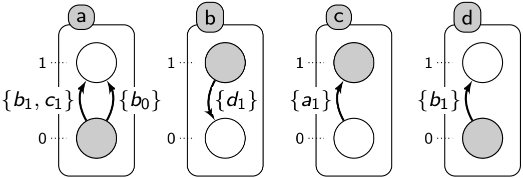


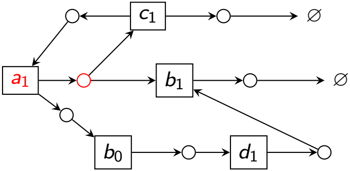


Node	Rank	$\mathbb{V}$
$\varnothing$ (of b_1)	1	$\varnothing$
b_1	2	$\{\{b_1\}\}$
$\{b_1\} \to d_1$	3	$\{\{b_1\}\}$
d_1	4	$\{\{b_1\}, \{d_1\}\}$
$\{d_1\} \to b_0$	5	$\{\{b_1\}, \{d_1\}\}$
b_0	6	$\{\{b_0\}, \{b_1\}, \{d_1\}\}$
$\{b_0\} \to a_1$	7	$\{\{b_0\}, \{b_1\}, \{d_1\}\}$
$\varnothing$ (of c_1)	8	$\varnothing$
c_1	9	$\{\{c_1\}\}$
$\{b_1, c_1\} \to a_1$	9	$\{\{b_1\}, \{c_1\}\}$
a_1	9	$\{\{a_1\}, \{b_1\}, \{b_0, c_1\}, \{c_1, d_1\}\}$
$\{a_1\} \to c_1$	9	$\{\{a_1\}, \{b_1\}, \{b_0, c_1\}, \{c_1, d_1\}\}$

Rank: topological numbering of the nodes in SLCG, nodes with higher rank cannot be successor of nodes with lower rank.

Cut Set

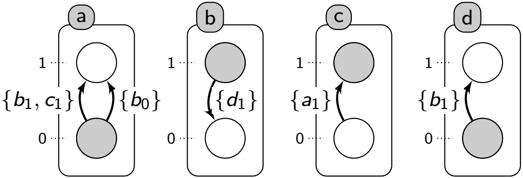


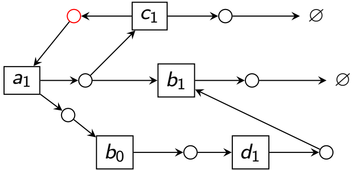


Node	Rank	$\mathbb{V}$
$\varnothing$ (of b_1)	1	$\varnothing$
b_1	2	$\{\{b_1\}\}$
$\{b_1\} \rightarrow d_1$	3	$\{\{b_1\}\}$
d_1	4	$\{\{b_1\}, \{d_1\}\}$
$\{d_1\} \rightarrow b_0$	5	$\{\{b_1\}, \{d_1\}\}$
b_0	6	$\{\{b_0\}, \{b_1\}, \{d_1\}\}$
$\{b_0\} \rightarrow a_1$	7	$\{\{b_0\}, \{b_1\}, \{d_1\}\}$
$\varnothing$ (of c_1)	8	$\varnothing$
c_1	9	$\{\{c_1\}\}$
$\{b_1, c_1\} \rightarrow a_1$	9	$\{\{b_1\}, \{c_1\}\}$
a_1	9	$\{\{a_1\}, \{b_1\}, \{b_0, c_1\}, \{c_1, d_1\}\}$
$\{a_1\} \rightarrow c_1$	9	$\{\{a_1\}, \{b_1\}, \{b_0, c_1\}, \{c_1, d_1\}\}$

Rank: topological numbering of the nodes in SLCG, nodes with higher rank cannot be successor of nodes with lower rank.

Cut Set

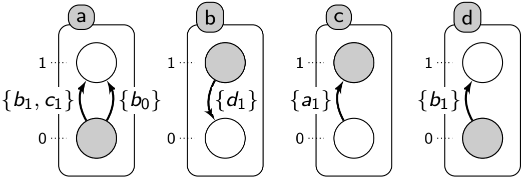


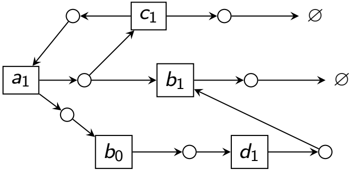


Node	Rank	$\mathbb{V}$
$\varnothing$ (of b_1)	1	$\varnothing$
b_1	2	$\{\{b_1\}\}$
$\{b_1\} \rightarrow d_1$	3	$\{\{b_1\}\}$
d_1	4	$\{\{b_1\}, \{d_1\}\}$
$\{d_1\} \rightarrow b_0$	5	$\{\{b_1\}, \{d_1\}\}$
b_0	6	$\{\{b_0\}, \{b_1\}, \{d_1\}\}$
$\{b_0\} \rightarrow a_1$	7	$\{\{b_0\}, \{b_1\}, \{d_1\}\}$
$\varnothing$ (of c_1)	8	$\varnothing$
c_1	9	$\{\{c_1\}\}$
$\{b_1, c_1\} \rightarrow a_1$	9	$\{\{b_1\}, \{c_1\}\}$
a_1	9	$\{\{a_1\}, \{b_1\}, \{b_0, c_1\}, \{c_1, d_1\}\}$
$\{a_1\} \rightarrow c_1$	9	$\{\{a_1\}, \{b_1\}, \{b_0, c_1\}, \{c_1, d_1\}\}$

Rank: topological numbering of the nodes in SLCG, nodes with higher rank cannot be successor of nodes with lower rank.

Cut Set



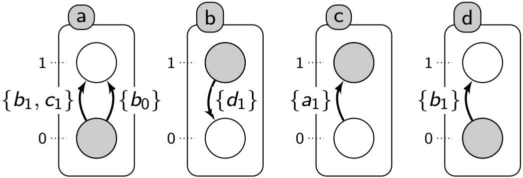


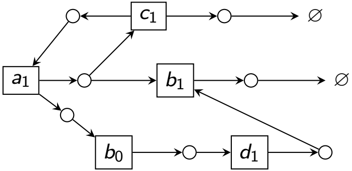
Node	Rank	$\mathbb{V}$
$\varnothing$ (of b_1)	1	$\varnothing$
b_1	2	$\{\{b_1\}\}$
$\{b_1\} \to d_1$	3	$\{\{b_1\}\}$
d_1	4	$\{\{b_1\}, \{d_1\}\}$
$\{d_1\} \to b_0$	5	$\{\{b_1\}, \{d_1\}\}$
b_0	6	$\{\{b_0\}, \{b_1\}, \{d_1\}\}$
$\{b_0\} \to a_1$	7	$\{\{b_0\}, \{b_1\}, \{d_1\}\}$
$\varnothing$ (of c_1)	8	$\varnothing$
c_1	9	$\{\{c_1\}\}$
$\{b_1, c_1\} \to a_1$	9	$\{\{b_1\}, \{c_1\}\}$
a_1	9	$\{\{a_1\}, \{b_1\}, \{b_0, c_1\}, \{c_1, d_1\}\}$
$\{a_1\} \to c_1$	9	$\{\{a_1\}, \{b_1\}, \{b_0, c_1\}, \{c_1, d_1\}\}$

Rank: topological numbering of the nodes in SLCG, nodes with higher rank cannot be successor of nodes with lower rank.

By inhibiting the reachability of one of the sets in cut set $\{\{a_1\}, \{b_1\}, \{b_0, c_1\}, \{c_1, d_1\}\}$, a_1 shall not be reachable.
Corresponding transition sets can be deduced.

Cut Set





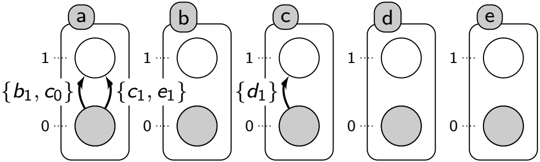
Node	Rank	$\mathbb{V}$
$\varnothing$ (of b_1)	1	$\varnothing$
b_1	2	$\{\{b_1\}\}$
$\{b_1\} \rightarrow d_1$	3	$\{\{b_1\}\}$
d_1	4	$\{\{b_1\}, \{d_1\}\}$
$\{d_1\} \rightarrow b_0$	5	$\{\{b_1\}, \{d_1\}\}$
b_0	6	$\{\{b_0\}, \{b_1\}, \{d_1\}\}$
$\{b_0\} \rightarrow a_1$	7	$\{\{b_0\}, \{b_1\}, \{d_1\}\}$
$\varnothing$ (of c_1)	8	$\varnothing$
c_1	9	$\{\{c_1\}\}$
$\{b_1, c_1\} \rightarrow a_1$	9	$\{\{b_1\}, \{c_1\}\}$
a_1	9	$\{\{a_1\}, \{b_1\}, \{b_0, c_1\}, \{c_1, d_1\}\}$
$\{a_1\} \rightarrow c_1$	9	$\{\{a_1\}, \{b_1\}, \{b_0, c_1\}, \{c_1, d_1\}\}$

Rank: topological numbering of the nodes in SLCG, nodes with higher rank cannot be successor of nodes with lower rank.

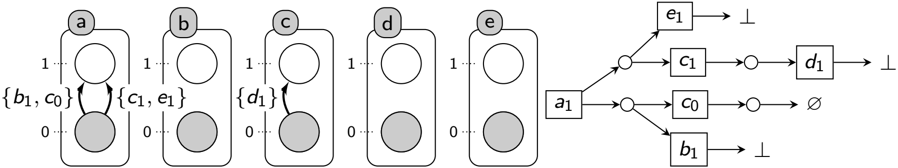
By inhibiting the reachability of one of the sets in cut set $\{\{a_1\}, \{b_1\}, \{b_0, c_1\}, \{c_1, d_1\}\}$, a_1 shall not be reachable.
Corresponding transition sets can be deduced.

Modification depends on the consistency with the result of learning phase.

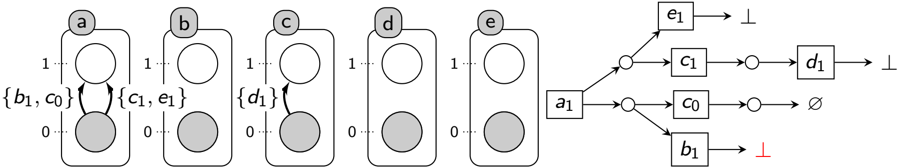
Completion Set



Completion Set



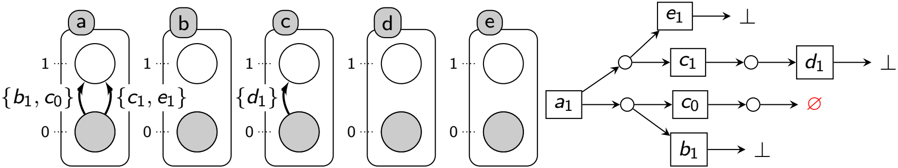
Completion Set



Node	Rank	$\mathbb{V}$
$\bot$ (of b_1)	1	$\varnothing$
b_1	2	$\{\{b_1\}\}$
$\varnothing$ (of c_0)	3	$\{\varnothing\}$
c_1	4	$\{\varnothing\}$
$\{b_1, c_0\} \to a_1$	5	$\{\{b_1\}\}$
$\bot$ (of d_1)	6	$\varnothing$
d_1	7	$\{\{d_1\}\}$
$\{d_1\} \to c_1$	8	$\{\{d_1\}\}$
c_1	9	$\{\{c_1, d_1\}\}$
$\bot$ (of e_1)	10	$\varnothing$
e_1	11	$\{\{e_1\}\}$
$\{c_1, e_1\} \to a_1$	12	$\{\{c_1, e_1\}, \{d_1, e_1\}\}$
a_1	13	$\{\{a_1\}, \{b_1\}, \{c_1, e_1\}, \{d_1, e_1\}\}$

Rank: topological numbering of the nodes in SLCG, nodes with higher rank cannot be successor of nodes with lower rank

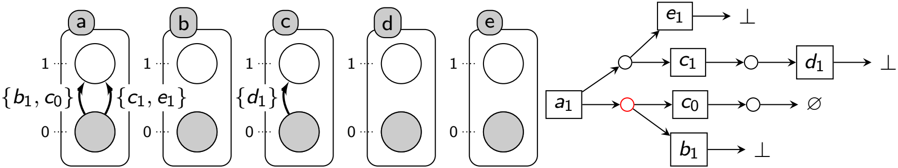
Completion Set



Node	Rank	$\mathbb{V}$
$\bot$ (of b_1)	1	$\varnothing$
b_1	2	$\{\{b_1\}\}$
$\varnothing$ (of c_0)	3	$\{\varnothing\}$
c_1	4	$\{\varnothing\}$
$\{b_1, c_0\} \rightarrow a_1$	5	$\{\{b_1\}\}$
$\bot$ (of d_1)	6	$\varnothing$
d_1	7	$\{\{d_1\}\}$
$\{d_1\} \rightarrow c_1$	8	$\{\{d_1\}\}$
c_1	9	$\{\{c_1, d_1\}\}$
$\bot$ (of e_1)	10	$\varnothing$
e_1	11	$\{\{e_1\}\}$
$\{c_1, e_1\} \rightarrow a_1$	12	$\{\{c_1, e_1\}, \{d_1, e_1\}\}$
a_1	13	$\{\{a_1\}, \{b_1\}, \{c_1, e_1\}, \{d_1, e_1\}\}$

Rank: topological numbering of the nodes in SLCG, nodes with higher rank cannot be successor of nodes with lower rank

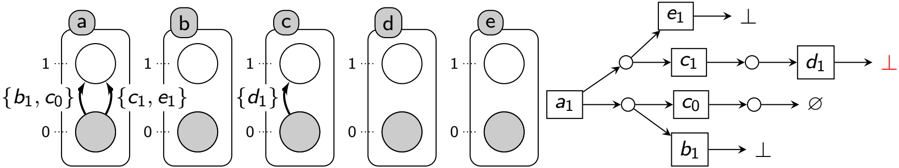
Completion Set



Node	Rank	$\mathbb{V}$
$\perp$ (of b_1)	1	$\varnothing$
b_1	2	$\{\{b_1\}\}$
$\varnothing$ (of c_0)	3	$\{\varnothing\}$
c_1	4	$\{\varnothing\}$
$\{b_1, c_0\} \rightarrow a_1$	5	$\{\{b_1\}\}$
$\perp$ (of d_1)	6	$\varnothing$
d_1	7	$\{\{d_1\}\}$
$\{d_1\} \rightarrow c_1$	8	$\{\{d_1\}\}$
c_1	9	$\{\{c_1, d_1\}\}$
$\perp$ (of e_1)	10	$\varnothing$
e_1	11	$\{\{e_1\}\}$
$\{c_1, e_1\} \rightarrow a_1$	12	$\{\{c_1, e_1\}, \{d_1, e_1\}\}$
a_1	13	$\{\{a_1\}, \{b_1\}, \{c_1, e_1\}, \{d_1, e_1\}\}$

Rank: topological numbering of the nodes in SLCG, nodes with higher rank cannot be successor of nodes with lower rank

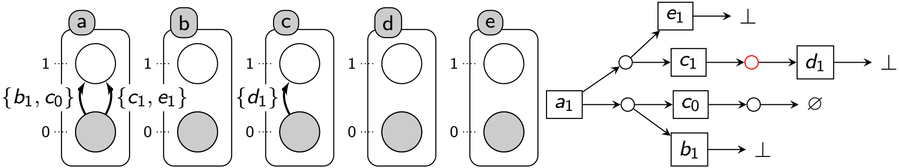
Completion Set



Node	Rank	$\mathbb{V}$
$\perp$ (of b_1)	1	$\varnothing$
b_1	2	$\{\{b_1\}\}$
$\varnothing$ (of c_0)	3	$\{\varnothing\}$
c_1	4	$\{\varnothing\}$
$\{b_1, c_0\} \rightarrow a_1$	5	$\{\{b_1\}\}$
$\perp$ (of d_1)	6	$\varnothing$
d_1	7	$\{\{d_1\}\}$
$\{d_1\} \rightarrow c_1$	8	$\{\{d_1\}\}$
c_1	9	$\{\{c_1, d_1\}\}$
$\perp$ (of e_1)	10	$\varnothing$
e_1	11	$\{\{e_1\}\}$
$\{c_1, e_1\} \rightarrow a_1$	12	$\{\{c_1, e_1\}, \{d_1, e_1\}\}$
a_1	13	$\{\{a_1\}, \{b_1\}, \{c_1, e_1\}, \{d_1, e_1\}\}$

Rank: topological numbering of the nodes in SLCG, nodes with higher rank cannot be successor of nodes with lower rank

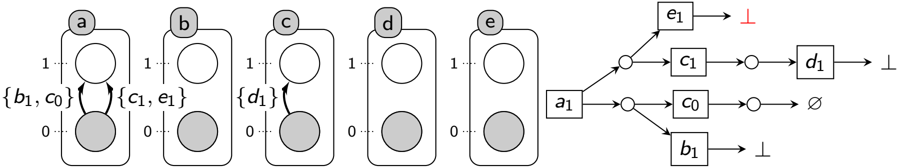
Completion Set



Node	Rank	$\mathbb{V}$
$\perp$ (of b_1)	1	$\varnothing$
b_1	2	$\{\{b_1\}\}$
$\varnothing$ (of c_0)	3	$\{\varnothing\}$
c_1	4	$\{\varnothing\}$
$\{b_1, c_0\} \to a_1$	5	$\{\{b_1\}\}$
$\perp$ (of d_1)	6	$\varnothing$
d_1	7	$\{\{d_1\}\}$
$\{d_1\} \to c_1$	8	$\{\{d_1\}\}$
c_1	9	$\{\{c_1, d_1\}\}$
$\perp$ (of e_1)	10	$\varnothing$
e_1	11	$\{\{e_1\}\}$
$\{c_1, e_1\} \to a_1$	12	$\{\{c_1, e_1\}, \{d_1, e_1\}\}$
a_1	13	$\{\{a_1\}, \{b_1\}, \{c_1, e_1\}, \{d_1, e_1\}\}$

Rank: topological numbering of the nodes in SLCG, nodes with higher rank cannot be successor of nodes with lower rank

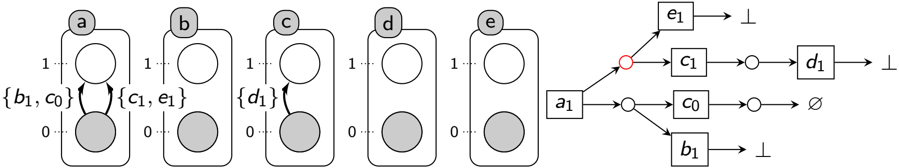
Completion Set



Node	Rank	$\mathbb{V}$
$\perp$ (of b_1)	1	$\varnothing$
b_1	2	$\{\{b_1\}\}$
$\varnothing$ (of c_0)	3	$\{\varnothing\}$
c_1	4	$\{\varnothing\}$
$\{b_1, c_0\} \rightarrow a_1$	5	$\{\{b_1\}\}$
$\perp$ (of d_1)	6	$\varnothing$
d_1	7	$\{\{d_1\}\}$
$\{d_1\} \rightarrow c_1$	8	$\{\{d_1\}\}$
c_1	9	$\{\{c_1, d_1\}\}$
$\perp$ (of e_1)	10	$\varnothing$
e_1	11	$\{\{e_1\}\}$
$\{c_1, e_1\} \rightarrow a_1$	12	$\{\{c_1, e_1\}, \{d_1, e_1\}\}$
a_1	13	$\{\{a_1\}, \{b_1\}, \{c_1, e_1\}, \{d_1, e_1\}\}$

Rank: topological numbering of the nodes in SLCG, nodes with higher rank cannot be successor of nodes with lower rank

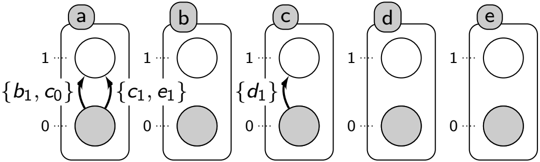
Completion Set

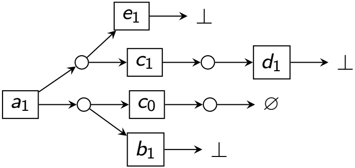


Node	Rank	$\mathbb{V}$
$\perp$ (of b_1)	1	$\varnothing$
b_1	2	$\{\{b_1\}\}$
$\varnothing$ (of c_0)	3	$\{\varnothing\}$
c_1	4	$\{\varnothing\}$
$\{b_1, c_0\} \to a_1$	5	$\{\{b_1\}\}$
$\perp$ (of d_1)	6	$\varnothing$
d_1	7	$\{\{d_1\}\}$
$\{d_1\} \to c_1$	8	$\{\{d_1\}\}$
c_1	9	$\{\{c_1, d_1\}\}$
$\perp$ (of e_1)	10	$\varnothing$
e_1	11	$\{\{e_1\}\}$
$\{c_1, e_1\} \to a_1$	12	$\{\{c_1, e_1\}, \{d_1, e_1\}\}$
a_1	13	$\{\{a_1\}, \{b_1\}, \{c_1, e_1\}, \{d_1, e_1\}\}$

Rank: topological numbering of the nodes in SLCG, nodes with higher rank cannot be successor of nodes with lower rank

Completion Set



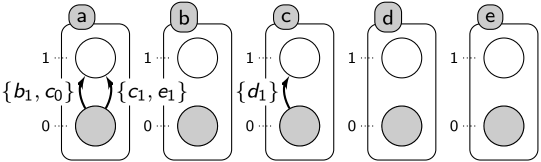


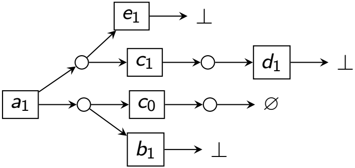
Node	Rank	$\mathbb{V}$
$\perp$ (of b_1)	1	$\varnothing$
b_1	2	$\{\{b_1\}\}$
$\varnothing$ (of c_0)	3	$\{\varnothing\}$
c_1	4	$\{\varnothing\}$
$\{b_1, c_0\} \rightarrow a_1$	5	$\{\{b_1\}\}$
$\perp$ (of d_1)	6	$\varnothing$
d_1	7	$\{\{d_1\}\}$
$\{d_1\} \rightarrow c_1$	8	$\{\{d_1\}\}$
c_1	9	$\{\{c_1, d_1\}\}$
$\perp$ (of e_1)	10	$\varnothing$
e_1	11	$\{\{e_1\}\}$
$\{c_1, e_1\} \rightarrow a_1$	12	$\{\{c_1, e_1\}, \{d_1, e_1\}\}$
a_1	13	$\{\{a_1\}, \{b_1\}, \{c_1, e_1\}, \{d_1, e_1\}\}$

Rank: topological numbering of the nodes in SLCG, nodes with higher rank cannot be successor of nodes with lower rank

By assuring the reachability of one of the sets in completion set $\{\{a_1\}, \{b_1\}, \{b_0, c_1\}, \{c_1, d_1\}\}$, a_1 shall be reachable. Corresponding transition sets can be deduced.

Completion Set





Node	Rank	$\mathbb{V}$
$\perp$ (of b_1)	1	$\varnothing$
b_1	2	$\{\{b_1\}\}$
$\varnothing$ (of c_0)	3	$\{\varnothing\}$
c_1	4	$\{\varnothing\}$
$\{b_1, c_0\} \to a_1$	5	$\{\{b_1\}\}$
$\perp$ (of d_1)	6	$\varnothing$
d_1	7	$\{\{d_1\}\}$
$\{d_1\} \to c_1$	8	$\{\{d_1\}\}$
c_1	9	$\{\{c_1, d_1\}\}$
$\perp$ (of e_1)	10	$\varnothing$
e_1	11	$\{\{e_1\}\}$
$\{c_1, e_1\} \to a_1$	12	$\{\{c_1, e_1\}, \{d_1, e_1\}\}$
a_1	13	$\{\{a_1\}, \{b_1\}, \{c_1, e_1\}, \{d_1, e_1\}\}$

Rank: topological numbering of the nodes in SLCG, nodes with higher rank cannot be successor of nodes with lower rank

By assuring the reachability of one of the sets in completion set $\{\{a_1\}, \{b_1\}, \{b_0, c_1\}, \{c_1, d_1\}\}$, a_1 shall be reachable. Corresponding transition sets can be deduced.

Modification depends on the consistency with the result of learning phase.

Recap of Cut/Completion Set

- Cut/completion set compute all the possible modifications

Recap of Cut/Completion Set

- Cut/completion set compute all the possible modifications
- The complexity comes mainly from the cartesian product of sets

Recap of Cut/Completion Set

- Cut/completion set compute all the possible modifications
- The complexity comes mainly from the cartesian product of sets
- Consider n sets containing k elements

$$\underbrace{E_1 = \overbrace{(e_{1,1}, e_{1,2}, \dots, e_{1,k})}^{k \text{ elements}}, E_2 = (e_{2,1}, e_{2,2}, \dots, e_{2,k}), \dots, E_n = (e_{n,1}, e_{n,2}, \dots, e_{n,k})}_{n \text{ sets}}$$

Recap of Cut/Completion Set

- Cut/completion set compute all the possible modifications
- The complexity comes mainly from the cartesian product of sets
- Consider n sets containing k elements

$$\underbrace{E_1 = \overbrace{(e_{1,1}, e_{1,2}, \ldots, e_{1,k})}^{k \text{ elements}}, E_2 = (e_{2,1}, e_{2,2}, \ldots, e_{2,k}), \ldots, E_n = (e_{n,1}, e_{n,2}, \ldots, e_{n,k})}_{n \text{ sets}}$$

- The cardinality of the cartesian product $\prod_{i=1}^{n} E_i$ is k^n, exponential

Recap of Cut/Completion Set

- Cut/completion set compute all the possible modifications
- The complexity comes mainly from the cartesian product of sets
- Consider n sets containing k elements

$$\underbrace{E_1 = \overbrace{(e_{1,1}, e_{1,2}, \ldots, e_{1,k})}^{k \text{ elements}}, E_2 = (e_{2,1}, e_{2,2}, \ldots, e_{2,k}), \ldots, E_n = (e_{n,1}, e_{n,2}, \ldots, e_{n,k})}_{n \text{ sets}}$$

- The cardinality of the cartesian product $\prod_{i=1}^{n} E_i$ is k^n, exponential
- $\Longrightarrow$ Heuristics aiming at finding **one** solution instead of **all** the solutions

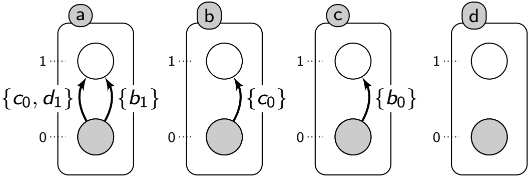
Recap of Cut/Completion Set

- Cut/completion set compute all the possible modifications
- The complexity comes mainly from the cartesian product of sets
- Consider n sets containing k elements

$$\underbrace{E_1 = \overbrace{(e_{1,1}, e_{1,2}, \ldots, e_{1,k})}^{k \text{ elements}}, E_2 = (e_{2,1}, e_{2,2}, \ldots, e_{2,k}), \ldots, E_n = (e_{n,1}, e_{n,2}, \ldots, e_{n,k})}_{n \text{ sets}}$$

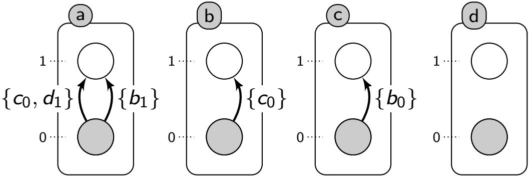
- The cardinality of the cartesian product $\prod_{i=1}^{n} E_i$ is k^n, exponential
- $\Longrightarrow$ Heuristics aiming at finding **one** solution instead of **all** the solutions
  - Compute with the SLCG involving all the wanted states in *a priori* knowledge

Heuristic Revision

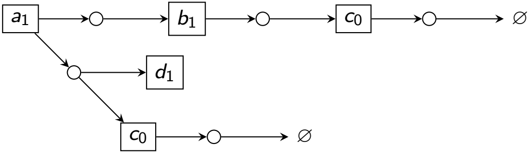


$\alpha = \langle a_0, b_0, c_0, d_0 \rangle$, $\omega = a_1$
$U_K = \{(\alpha, b_1), (\alpha, d_1)\}$
$R_K = \{(\alpha, a_1)\}$

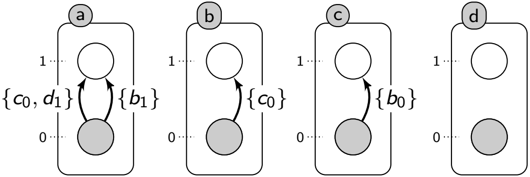
Heuristic Revision



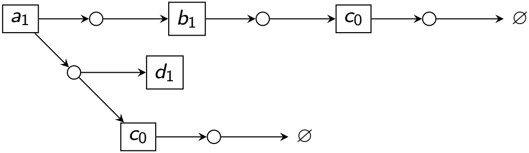
$$\alpha = \langle a_0, b_0, c_0, d_0 \rangle,\ \omega = a_1$$
$$U_K = \{(\alpha, b_1), (\alpha, d_1)\}$$
$$R_K = \{(\alpha, a_1)\}$$



Heuristic Revision

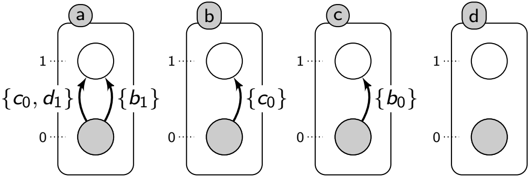


$\alpha = \langle a_0, b_0, c_0, d_0 \rangle,\ \omega = a_1$
$U_K = \{(\alpha, b_1), (\alpha, d_1)\}$
$R_K = \{(\alpha, a_1)\}$

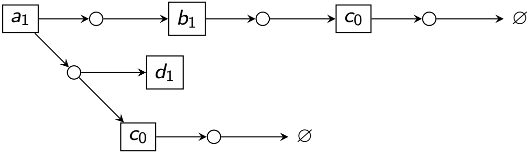


- $L = \{(\alpha, a_1) : \{(\alpha, b_1), (\alpha, d_1)\},\ (\alpha, b_1) : \varnothing,\ (\alpha, d_1) : \varnothing\}$

Heuristic Revision

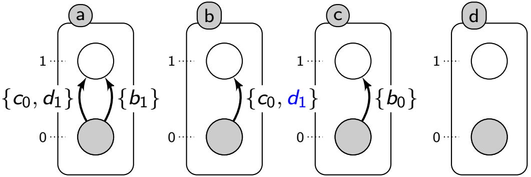


$\alpha = \langle a_0, b_0, c_0, d_0 \rangle,\ \omega = a_1$
$U_K = \{(\alpha, b_1), (\alpha, d_1)\}$
$R_K = \{(\alpha, a_1)\}$

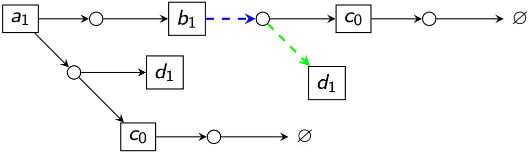


- $L = \{(\alpha, a_1) : \{(\alpha, b_1), (\alpha, d_1)\},\ (\alpha, b_1) : \varnothing,\ (\alpha, d_1) : \varnothing\}$
- Start from (α, b_1) and (α, d_1), choose (α, b_1) as (α, d_1) is already satisfied

Heuristic Revision

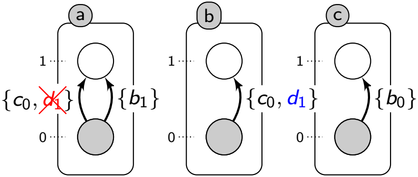


$\alpha = \langle a_0, b_0, c_0, d_0 \rangle$, $\omega = a_1$
$U_K = \{(\alpha, b_1), (\alpha, d_1)\}$
$R_K = \{(\alpha, a_1)\}$

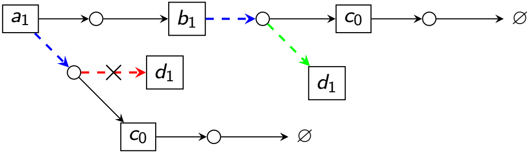


- $L = \{(\alpha, a_1) : \{(\alpha, b_1), (\alpha, d_1)\},\ (\alpha, b_1) : \varnothing,\ (\alpha, d_1) : \varnothing\}$
- Start from (α, b_1) and (α, d_1), choose (α, b_1) as (α, d_1) is already satisfied
- $\{c_0\} \rightarrow b_1$ can be specialized to $\{c_0, d_1\} \rightarrow b_1$ to make b_1 unreachable

Heuristic Revision

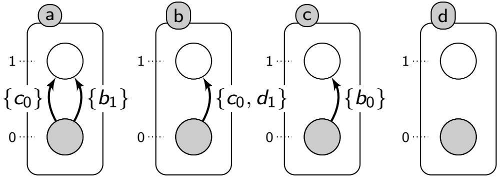


$$\alpha = \langle a_0, b_0, c_0, d_0 \rangle,\ \omega = a_1$$
$$U_K = \{(\alpha, b_1), (\alpha, d_1)\}$$
$$R_K = \{(\alpha, a_1)\}$$

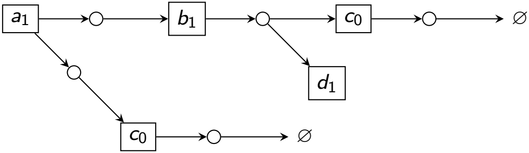


- $L = \{(\alpha, a_1) : \{(\alpha, b_1), (\alpha, d_1)\},\ (\alpha, b_1) : \varnothing,\ (\alpha, d_1) : \varnothing\}$
- Start from (α, b_1) and (α, d_1), choose (α, b_1) as (α, d_1) is already satisfied
- $\{c_0\} \to b_1$ can be specialized to $\{c_0, d_1\} \to b_1$ to make b_1 unreachable
- $\{d_1, c_0\} \to a_1$ can only be generalized to $\{c_0\} \to a_1$ as $(\alpha, d_1) \in U_K$

Heuristic Revision



$\alpha = \langle a_0, b_0, c_0, d_0 \rangle,\ \omega = a_1$
$U_K = \{(\alpha, b_1), (\alpha, d_1)\}$
$R_K = \{(\alpha, a_1)\}$



- $L = \{(\alpha, a_1) : \{(\alpha, b_1), (\alpha, d_1)\},\ (\alpha, b_1) : \varnothing,\ (\alpha, d_1) : \varnothing\}$
- Start from (α, b_1) and (α, d_1), choose (α, b_1) as (α, d_1) is already satisfied
- $\{c_0\} \rightarrow b_1$ can be specialized to $\{c_0, d_1\} \rightarrow b_1$ to make b_1 unreachable
- $\{d_1, c_0\} \rightarrow a_1$ can only be generalized to $\{c_0\} \rightarrow a_1$ as $(\alpha, d_1) \in U_K$
- Check the reachability of (α, a_1): reachable, finish

Summary of Model Inference

- Aim: use time-series data + reachability properties to infer the real system

Summary of Model Inference

- Aim: use time-series data + reachability properties to infer the real system
- Inference = Learning + Revising

Summary of Model Inference

- Aim: use time-series data + reachability properties to infer the real system
- Inference = Learning + Revising

	CRAC	M2RIT
Learning phase	Correlation Coefficients	Learning from Interpretation Transitions (LFIT)

Summary of Model Inference

- Aim: use time-series data + reachability properties to infer the real system
- Inference = Learning + Revising

	CRAC	M2RIT
Learning phase	Correlation Coefficients	Learning from Interpretation Transitions (LFIT)
Input data	Continuous	Discrete

Summary of Model Inference

- Aim: use time-series data + reachability properties to infer the real system
- Inference = Learning + Revising

	CRAC	M2RIT
Learning phase	Correlation Coefficients	Learning from Interpretation Transitions (LFIT)
Input data	Continuous	Discrete
Revising phase	Cut set + completion set	Heuristic revision+ time-series data

Summary of Model Inference

- Aim: use time-series data + reachability properties to infer the real system
- Inference = Learning + Revising

	CRAC	M2RIT
Learning phase	Correlation Coefficients	Learning from Interpretation Transitions (LFIT)
Input data	Continuous	Discrete
Revising phase	Cut set + completion set	Heuristic revision+ time-series data
Tolerance to noise	Yes	No

Summary of Model Inference

- Aim: use time-series data + reachability properties to infer the real system
- Inference = Learning + Revising

	CRAC	M2RIT
Learning phase	Correlation Coefficients	Learning from Interpretation Transitions (LFIT)
Input data	Continuous	Discrete
Revising phase	Cut set + completion set	Heuristic revision+ time-series data
Tolerance to noise	Yes	No

- Test result: we obtain models reproducing original time-series data and satisfying given reachability properties, but not identical to real system

Summary of Model Inference

- Aim: use time-series data + reachability properties to infer the real system
- Inference = Learning + Revising

	CRAC	M2RIT
Learning phase	Correlation Coefficients	Learning from Interpretation Transitions (LFIT)
Input data	Continuous	Discrete
Revising phase	Cut set + completion set	Heuristic revision+ time-series data
Tolerance to noise	Yes	No

- Test result: we obtain models reproducing original time-series data and satisfying given reachability properties, but not identical to real system
  - By picking only reachability as revision criteria, inferred models are bisimilar to original systems in the sense of reachability

Conclusion & Future Work

- Bridge connecting biological regulatory networks and dynamical properties

Conclusion & Future Work

- Bridge connecting biological regulatory networks and dynamical properties
- Reachability analysis

Conclusion & Future Work

- Bridge connecting biological regulatory networks and dynamical properties
- Reachability analysis
    - PermReach and ASPReach: faster than traditional model checkers and more conclusive than pure static analysis → balance between complexity and completeness

Conclusion & Future Work

- Bridge connecting biological regulatory networks and dynamical properties
- Reachability analysis
  - PermReach and ASPReach: faster than traditional model checkers and more conclusive than pure static analysis → balance between complexity and completeness
  - Deal with systems with up to 10^3 automata

Conclusion & Future Work

- Bridge connecting biological regulatory networks and dynamical properties
- Reachability analysis
  - PermReach and ASPReach: faster than traditional model checkers and more conclusive than pure static analysis → balance between complexity and completeness
  - Deal with systems with up to 10^3 automata
  - $\overset{?}{\Rightarrow}$ **Hybrid analyzer is more performing in reachability analysis**

Conclusion & Future Work

- Bridge connecting biological regulatory networks and dynamical properties
- Reachability analysis
  - PermReach and ASPReach: faster than traditional model checkers and more conclusive than pure static analysis → balance between complexity and completeness
  - Deal with systems with up to 10^3 automata
  - $\overset{?}{\Rightarrow}$ **Hybrid analyzer is more performing in reachability analysis**
- Model learning and revision

Conclusion & Future Work

- Bridge connecting biological regulatory networks and dynamical properties
- Reachability analysis
  - PermReach and ASPReach: faster than traditional model checkers and more conclusive than pure static analysis → balance between complexity and completeness
  - Deal with systems with up to 10^3 automata
  - $\overset{?}{\Rightarrow}$ **Hybrid analyzer is more performing in reachability analysis**
- Model learning and revision
  - A comparatively new domain

Conclusion & Future Work

- Bridge connecting biological regulatory networks and dynamical properties
- Reachability analysis
  - PermReach and ASPReach: faster than traditional model checkers and more conclusive than pure static analysis → balance between complexity and completeness
  - Deal with systems with up to 10^3 automata
  - $\overset{?}{\Rightarrow}$ **Hybrid analyzer is more performing in reachability analysis**
- Model learning and revision
  - A comparatively new domain
  - Generic solution vs. heuristics → balance between complexity and completeness

Conclusion & Future Work

- Bridge connecting biological regulatory networks and dynamical properties
- Reachability analysis
  - PermReach and ASPReach: faster than traditional model checkers and more conclusive than pure static analysis → balance between complexity and completeness
  - Deal with systems with up to 10^3 automata
  - $\overset{?}{\Rightarrow}$ **Hybrid analyzer is more performing in reachability analysis**
- Model learning and revision
  - A comparatively new domain
  - Generic solution vs. heuristics → balance between complexity and completeness
  - **More metrics other than reachability are needed to obtain a more precise model**

Conclusion & Future Work

- Bridge connecting biological regulatory networks and dynamical properties
- Reachability analysis
  - PermReach and ASPReach: faster than traditional model checkers and more conclusive than pure static analysis → balance between complexity and completeness
  - Deal with systems with up to 10^3 automata
  - $\overset{?}{\Rightarrow}$ **Hybrid analyzer is more performing in reachability analysis**
- Model learning and revision
  - A comparatively new domain
  - Generic solution vs. heuristics → balance between complexity and completeness
  - **More metrics other than reachability are needed to obtain a more precise model**
- **More biological applications and discover more applicable domains**

Reachability Analysis and Revision of Dynamics of Biological Regulatory Networks

Xinwei Chai

École Centrale de Nantes
Le Laboratoire des Sciences du Numérique de Nantes
xinwei.chai@ls2n.fr

Rapporteurs :	Gilles BERNOT, Professeur des universités, Université Côte d'Azur
	Pascale LE GALL, Professeur des universités, Centrale Supélec
Examinateurs :	Béatrice DUVAL, Professeur des universités, Université d'Angers
	Loïc PAULEVÉ, Chargé de recherche, LaBRI, UMR CNRS
Directeur de thèse :	Olivier ROUX, Professeur des universités, École Centrale de Nantes
Co-encadrant de thèse :	Morgan MAGNIN, Professeur des universités, École Centrale de Nantes

May 24, 2019

Bibliography I

Chitta Baral.
Knowledge representation, reasoning and declarative problem solving.
Cambridge university press, 2003.

Robert K Brayton, Gary D Hachtel, Alberto Sangiovanni-Vincentelli, Fabio Somenzi, Adnan Aziz, Szu-Tsung Cheng, Stephen Edwards, Sunil Khatri, Yuji Kukimoto, Abelardo Pardo, et al.
VIS: A system for verification and synthesis.
In *International conference on computer aided verification*, pages 428–432. Springer, 1996.

Gilles Bernot and Fariza Tahi.
Behaviour preservation of a biological regulatory network when embedded into a larger network.
Fundamenta Informaticae, 91(3-4):463–485, 2009.

Edmund Clarke, Armin Biere, Richard Raimi, and Yunshan Zhu.
Bounded model checking using satisfiability solving.
Formal methods in system design, 19(1):7–34, 2001.

Alessandro Cimatti, Edmund Clarke, Fausto Giunchiglia, and Marco Roveri.
NuSMV: a new symbolic model checker.
International Journal on Software Tools for Technology Transfer, 2(4):410–425, 2000.

Xinwei Chai, Tony Ribeiro, Morgan Magnin, Olivier Roux, and Katsumi Inoue.
Static analysis and stochastic search for reachability problem.
In *9th Static Analysis in Systems Biology, affiliated with Static Analysis Symposium*, 2018.

Bibliography II

Maxime Folschette, Loïc Paulevé, Morgan Magnin, and Olivier Roux.
Sufficient conditions for reachability in automata networks with priorities.
Theoretical Computer Science, 608:66–83, 2015.

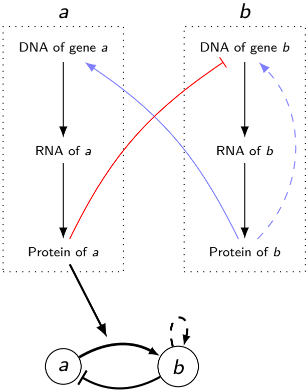
David Harel, Orna Kupferman, and Moshe Y Vardi.
On the complexity of verifying concurrent transition systems.
Information and Computation, 173(2):143–161, 2002.

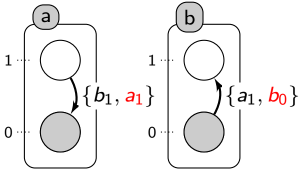
Keiichi Mochida, Satoru Koda, Komaki Inoue, and Ryuei Nishii.
Statistical and machine learning approaches to predict gene regulatory networks from transcriptome datasets.
Frontiers in Plant Science, 9:1770, 2018.

Loïc Paulevé, Morgan Magnin, and Olivier Roux.
Static analysis of biological regulatory networks dynamics using abstract interpretation.
Mathematical Structures in Computer Science, 22(04):651–685, 2012.

Tony Ribeiro, Maxime Folschette, Morgan Magnin, Olivier Roux, and Katsumi Inoue.
Learning dynamics with synchronous, asynchronous and general semantics.
In *the 28th International Conference on Inductive Logic Programming*, 2018.

Self Regulation

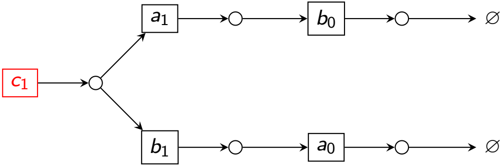




- The self regulation of b can contain two elements: self activation/inhibition and intrinsic degradation.
- However they are not necessary in ABAN case. As the transitions are between Boolean states, the state changes necessarily from one to another → there is no need to add self regulation related conditions in the transitions.

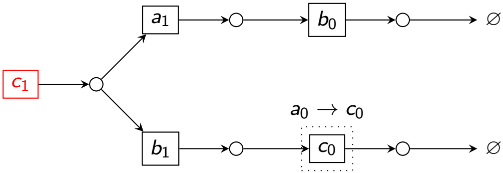
Test on Pint up to Date

- The last release of Pint is version 2018-11-30
- It returns inconclusive instances in the counterexample given in the presentation and also in λ-phage model



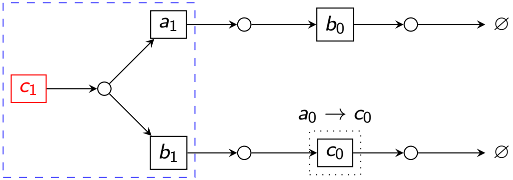
Test on Pint up to Date

- The last release of Pint is version 2018-11-30
- It returns inconclusive instances in the counterexample given in the presentation and also in λ-phage model



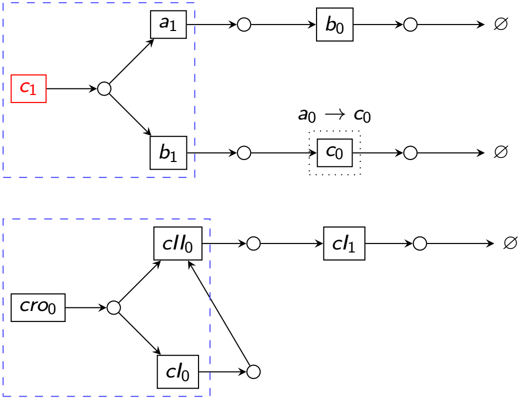
Test on Pint up to Date

- The last release of Pint is version 2018-11-30
- It returns inconclusive instances in the counterexample given in the presentation and also in λ-phage model



Test on Pint up to Date

- The last release of Pint is version 2018-11-30
- It returns inconclusive instances in the counterexample given in the presentation and also in λ-phage model


c_1
a_1
b_1
b_0
c_0
$a_0 \rightarrow c_0$
$\varnothing$
$\varnothing$
cro_0
cII_0
cI_0
cI_1
$\varnothing$