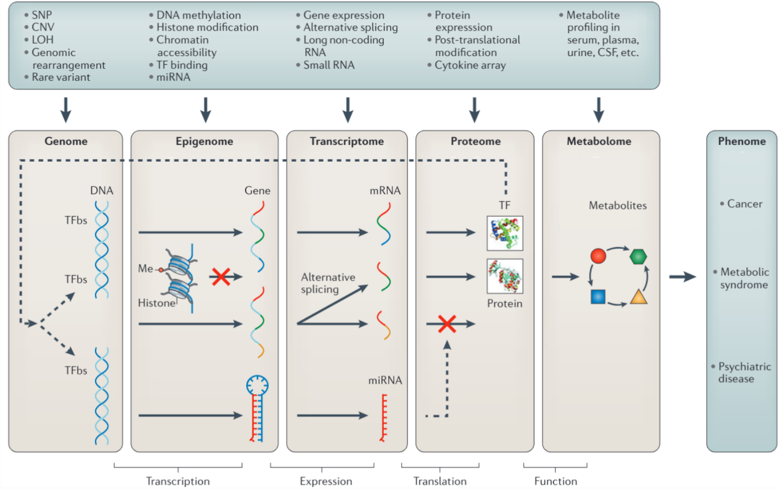
Accessible and Reproducible Analyses with the CoLoMoTo notebook

Pedro T. Monteiro

INESC-ID / IST - Universidade de Lisboa

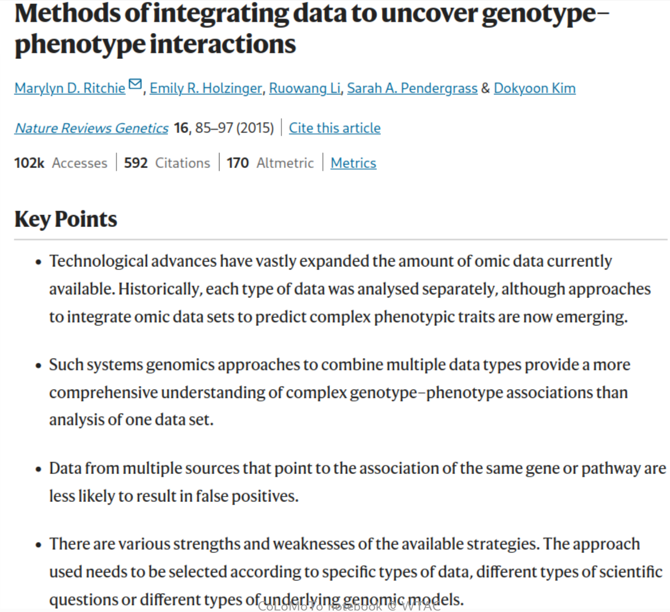
WTAC 2024 - Computational Systems Biology

Cell is tightly **regulated** at several levels



(Ritchie *et al.*, *Nat. Rev. Gen.* 2015)

Cellular interaction networks



Methods of integrating data to uncover genotype–phenotype interactions

Marylyn D. Ritchie ✉, Emily R. Holzinger, Ruowang Li, Sarah A. Pendergrass & Dokyoon Kim

Nature Reviews Genetics **16**, 85–97 (2015) | Cite this article

102k Accesses | **592** Citations | **170** Altmetric | Metrics

Key Points

- Technological advances have vastly expanded the amount of omic data currently available. Historically, each type of data was analysed separately, although approaches to integrate omic data sets to predict complex phenotypic traits are now emerging.

- Such systems genomics approaches to combine multiple data types provide a more comprehensive understanding of complex genotype–phenotype associations than analysis of one data set.

- Data from multiple sources that point to the association of the same gene or pathway are less likely to result in false positives.

- There are various strengths and weaknesses of the available strategies. The approach used needs to be selected according to specific types of data, different types of scientific questions or different types of underlying genomic models.

Networks to Models

Structural → Dynamical view

Network components need to be quantified:

- Values change over time (created / consumed)

Networks to Models

Structural → Dynamical view

Network components need to be quantified:

- Values change over time (created / consumed)
  ⇒ We need something to describe the system' evolution!

(Kitano, *Science* 2002)

Networks to Models

Structural → Dynamical view

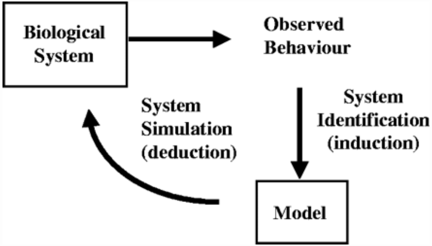
Network components need to be quantified:

- Values change over time (created / consumed)
  ⇒ We need something to describe the system' evolution!

(Kitano, *Science* 2002)

Rely on models to analyse networks

- Understand of the way in which particular molecular mechanisms control a cellular process
- Predict novel phenomena that can be confronted with experimental data


Biological System
Observed Behaviour
System Simulation (deduction)
System Identification (induction)
Model

(King, Garrett and Coghill, *Bioinformatics* 2005)

Modeling formalisms

Formalisms

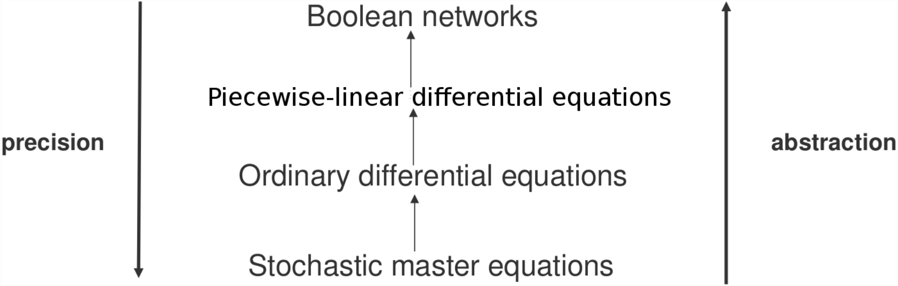
A variety of physical and mathematical approaches are used:

- Non-linear
- Piecewise-linear
- Logical (Boolean networks)
- Petri nets
- Statistical-mechanical
- Stochastic

Hierarchy of modeling formalisms

Describe the system at different levels of detail (e.g. continuous vs qualitative)

Modeling formalisms



precision

Boolean networks

Piecewise-linear differential equations

Ordinary differential equations

Stochastic master equations

abstraction

(de Jong, *J. Comput. Biol.* 2002) (Hasty *et al.*, *Nat. Rev. Genet.* 2001)

(Smolen *et al.*, *Bull. Mat. Biol.* 2000) (Szallassi *et al.*, *Systems Modeling in Cellular Biology* 2006)

Qualitative framework

Well established qualitative formalisms for modeling gene regulatory networks:

- Piecewise-linear differential equations (PLDEs)
  (Glass and Kauffman, *J. Theor. Biol.* 1973)
- **Logical formalism**
  (Boolean networks)
  (Thomas *et al.*, *Bull. Math. Biol.* 1995)
- Petri nets, ...

- Lack of quantitative data
  (ON/OFF mechanisms, thresholds)
- Discrete time

Qualitative framework

Well established qualitative formalisms for modeling gene regulatory networks:

- Piecewise-linear differential equations (PLDEs)
  (Glass and Kauffman, *J. Theor. Biol.* 1973)
- **Logical formalism**
  (Boolean networks)
  (Thomas *et al.*, *Bull. Math. Biol.* 1995)
- Petri nets, ...

- Lack of quantitative data
  (ON/OFF mechanisms, thresholds)
- Discrete time

Implicit assumptions

- Ignore intermediate gene products (mRNA)
- Ignore gene expression machinery (RNA polymerase, ribosome)
- Simplification of complex interactions of regulators with DNA to single response function

Gene regulatory networks

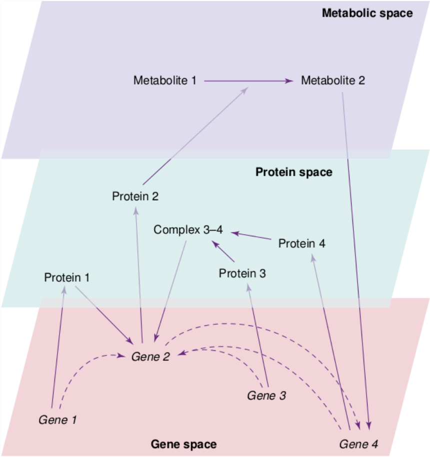
Gene regulatory networks

- consist of genes, gene products (RNA, proteins), and the regulatory effect of the latter on the expression of other genes

  (Bolouri *et al.*, *Computational Modeling of Gene Regulatory Networks* 2008)

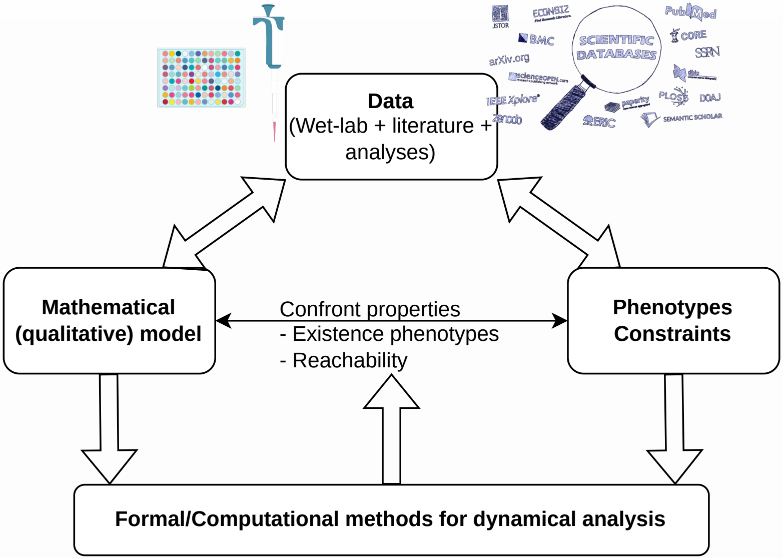
- cannot be reduced to direct interactions (transcriptional regulation), but also include indirect interactions (mediated by metabolism)

  (Brazhnik *et al.*, *Trends Biotechnol* 2002)



Different levels

Qualitative Dynamical Models in Biology



Data
(Wet-lab + literature + analyses)

Mathematical (qualitative) model

Confront properties
- Existence phenotypes
- Reachability

Phenotypes Constraints

Formal/Computational methods for dynamical analysis

Reproducibility in wet and dry labs

	Wet lab Lab book	**Dry lab** Program
Material	Biological material: - Cell type - Genetic background	Dataset: - Source - Format
Tools	Reagents & Equip.: - Provider, brand, . . .	Software: - Version
Protocol	- Steps - Conditions	- Commands - Parameters

Reproducibility in wet and dry labs

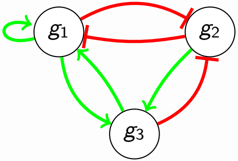
	Wet lab Lab book	**Dry lab** Program
Material	Biological material: - Cell type - Genetic background	Dataset: - Source - Format
Tools	Reagents & Equip.: - Provider, brand, . . .	Software: - Version
Protocol	- Steps - Conditions	- Commands - Parameters

The (reproducibility) devil is in the details!

Gene regulatory networks (topology)

Regulatory graph

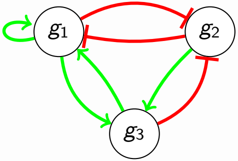
- Is a tuple (G, E) where G denotes a set of nodes/vertices (eg. gene expression) and E a set of edges
- An edge $(i, j) \in E$ indicates that i regulates the expression of j
- Edges can have sign information about interactions:
  - $(i, j, +)$ for "i activates j"
  - $(i, j, -)$ for "i inhibits j"



Boolean model (regulatory graph + logical rules) $\mathcal{M} = (G, K)$

(Thomas and d'Ari, *Biological Feedback* 1989)

- $G = \{g_i\}_{i=1,\ldots,n}$ is a set of n regulatory components
- $\prod_{g_i \in G} \{0,1\}$ defines the state space $\mathcal{S}$
- K is the set of regulatory functions $K_{g_i} \in K : \mathcal{S} \to \{0,1\}$



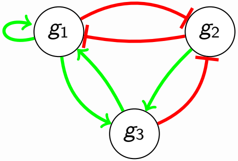
$$
\begin{aligned}
K_{g_1}(x) &= x_1 \lor (\neg x_1 \land \neg x_2) \lor (\neg x_1 \land x_3)\\
K_{g_2}(x) &= \neg x_1 \lor \neg x_3\\
K_{g_3}(x) &= x_1 \land x_2
\end{aligned}
$$

- Gene expression has two (or more) levels: **0** (Inactive) and **1** (Active)
- K_{g_i} is a combination of basic Boolean operators: $\neg$, $\land$ and $\lor$

Logical (Boolean) models

Boolean model (regulatory graph + logical rules) $\mathcal{M} = (G, K)$ (Thomas and d'Ari, *Biological Feedback* 1989)

- $G = \{g_i\}_{i=1,\ldots,n}$ is a set of n regulatory components
- $\prod_{g_i \in G}\{0,1\}$ defines the state space $\mathcal{S}$
- K is the set of regulatory functions $K_{g_i} \in K : \mathcal{S} \rightarrow \{0,1\}$



$$K_{g_1}(x) = x_1 \vee (\neg x_1 \wedge \neg x_2) \vee (\neg x_1 \wedge x_3)$$
$$K_{g_2}(x) = \neg x_1 \vee \neg x_3$$
$$K_{g_3}(x) = x_1 \wedge x_2$$

- Gene expression has two (or more) levels: **0** (Inactive) and **1** (Active)
- K_{g_i} is a combination of basic Boolean operators: $\neg$, $\wedge$ and $\vee$

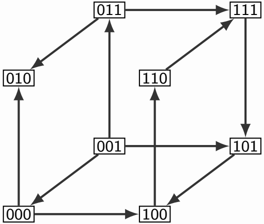
No quantitative parameters but requires a precise description of competing effects
(rates, updating scheme, etc.)

Gene regulatory networks dynamics

State transition graph (STG)

- Represents the dynamics of the Boolean model
- Nodes are states in $\mathcal{S}$
- Arcs $(i, j) \in \mathcal{S}^2$ denote transitions between states

Successor state: $\forall_{g_i \in G}, g_i^{t+1} = K_{g_i}(g_1^t, \ldots, g_n^t)$

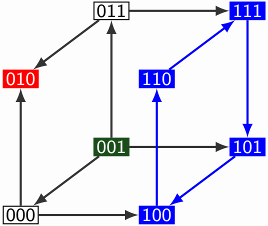


Gene regulatory networks dynamics

State transition graph (STG)

- Represents the dynamics of the Boolean model
- Nodes are states in $\mathcal{S}$
- Arcs $(i, j) \in \mathcal{S}^2$ denote transitions between states

Successor state: $\forall_{g_i \in G}, g_i^{t+1} = K_{g_i}(g_1^t, \ldots, g_n^t)$



Attractors

Correspond to (biologically relevant) asymptotic behaviours

- **Stable state**: all gene levels are maintained
  e.g., differentiated states, death, ...
- **Complex attractor**: long-lasting oscillatory behaviour
  e.g., cell cycle, ...

Gene regulatory network dynamics

Interesting properties

- What are the attractors of the system? (stable states, complex attractors)
  Is there multistability?
- Are these attractors reachable from initial conditions?
- Are these attractors maintained under input variations?
- What is the likelihood of reaching an attractor from a given portion of the state space?
- ...

Logical formalism - Challenge #1

Combinatorial explosion of number of states

# Components	# States	
	Boolean	**3-valued**
3	8	27
10	1 024	59 049
20	1 048 576	3 486 784 401
30	1 073 741 824	205 891 132 094 649
40	1 099 511 627 776	12 157 665 459 056 928 801

Logical formalism - Challenge #1

Combinatorial explosion of number of states

# Components	**# States**	
	Boolean	**3-valued**
3	8	27
10	1 024	59 049
20	1 048 576	3 486 784 401
30	1 073 741 824	205 891 132 094 649
40	1 099 511 627 776	12 157 665 459 056 928 801

- As the model size increases ⇒ manually intractable

Logical formalism - Challenge #1

Combinatorial explosion of number of states

# Components	# States	
	Boolean	**3-valued**
3	8	27
10	1 024	59 049
20	1 048 576	3 486 784 401
30	1 073 741 824	205 891 132 094 649
40	1 099 511 627 776	12 157 665 459 056 928 801

- As the model size increases ⇒ manually intractable
- State transitions graphs are not practical for most current models

Logical formalism - Challenge #1

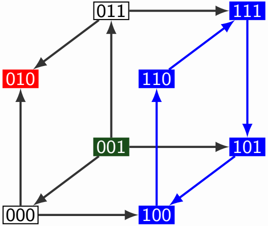
Combinatorial explosion of number of states

# Components	# States	
	Boolean	**3-valued**
3	8	27
10	1 024	59 049
20	1 048 576	3 486 784 401
30	1 073 741 824	205 891 132 094 649
40	1 099 511 627 776	12 157 665 459 056 928 801

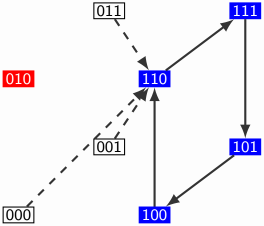
- As the model size increases ⇒ manually intractable
- State transitions graphs are not practical for most current models
- **Logical models enable efficient analytical methods**

Regulatory network dynamics

Asyncronous updating (non-deterministic)

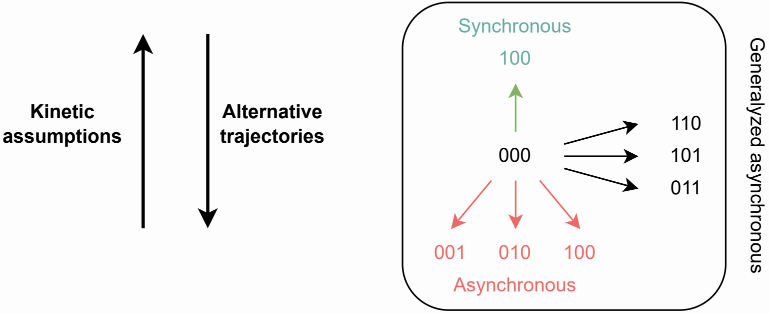


Syncronous updating (deterministic)



Logical formalism - Challenge #2

Explosion of number of transitions

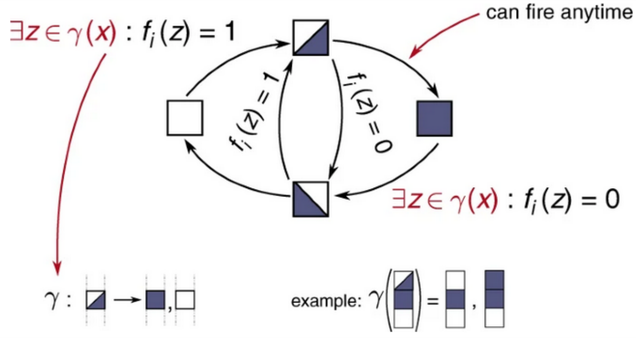


In absence of kinetic knowledge, many alternative trajectories are possible

Logical formalism - Challenge #2

Explosion of number of transitions

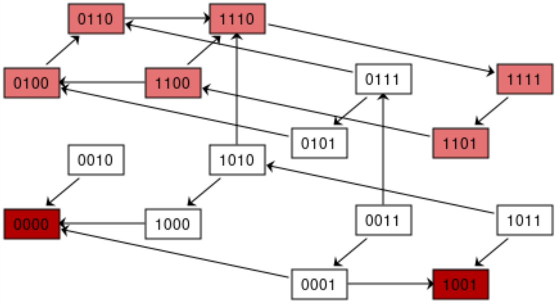
Most-permissive semantics


can fire anytime
$\exists z \in \gamma(x) : f_i(z) = 1$
$f_i(z) = 1$
$f_i(z) = 0$
$\exists z \in \gamma(x) : f_i(z) = 0$
γ :
example: γ

(Pauleve et al., *Nat. Comm.* 2020)

Useful to compute over-approximations of reachability properties

Methods: identification of attractors



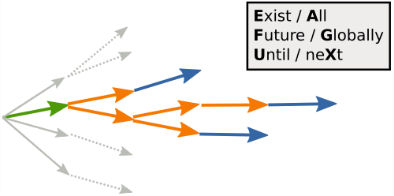
Methods

- Stable states: fixed points of the dynamics
  Constraint solving $\exists x : f(x) = x$ (Naldi et al., *CMSB* 2007)
- Complex attractors: cycles of the dynamics
  Symbolic exploration $\exists C : \forall x \in C : F(x) = C$ (Garg et al., *Bioinformatics* 2008)
- Trap spaces: stable patterns of the dynamics
  Constraint solving $\exists C : \forall x \in C : f(x) \in C$ (Klarner et al., *Nat Comput* 2015)

Methods: reachability analysis

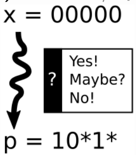
Identification of reachabilities (A → B)

- Explicitly generating the whole STG memory inefficient
- Symbolic model-checking (NuSMV, ...) (Monteiro et al., *WODES* 2014)



Exist / **A**ll
Future / **G**lobally
Until / ne**X**t

- Static analysis on prime-implicant graph (Pint) efficient/approximation (Pauleve et al., *Math Structures Comput Sci* 2012)



x = 00000

?
Yes!
Maybe?
No!

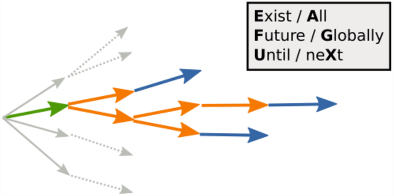
p = 10*1*

Methods: reachability analysis

Identification of reachabilities (A → B)

- Explicitly generating the whole STG memory inefficient
- Symbolic model-checking (NuSMV, ...)

(Monteiro et al., *WODES* 2014)

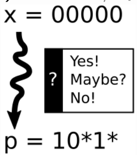


Exist / **A**ll
Future / **G**lobally
Until / ne**X**t

See Ben Hall's lecture!

- Static analysis on prime-implicant graph (Pint) efficient/approximation

(Pauleve et al., *Math Structures Comput Sci* 2012)



x = 00000

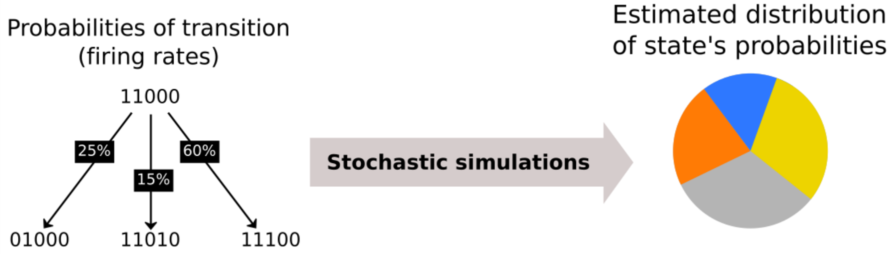
?

Yes!
Maybe?
No!

p = 10*1*

Methods: reachability analysis

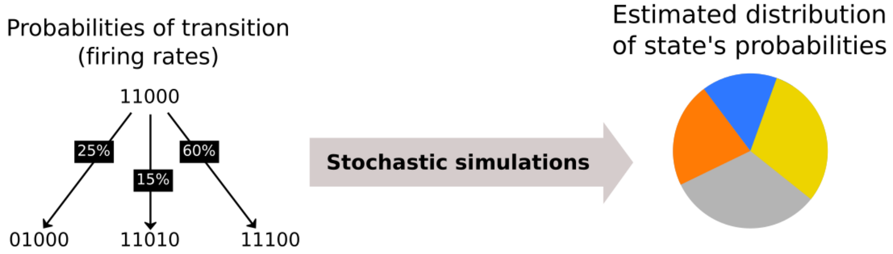
Quantification of reachability


Probabilities of transition (firing rates)
11000
25%
15%
60%
01000
11010
11100
Stochastic simulations
Estimated distribution of state's probabilities

- **Avatar** (Mendes *et al.*, *Frontiers in Physiology* 2018) @(bioLQM&GINsim)
- **MaBoSS** (Stoll *et al.*, *BMC Syst Biol* 2012)

Methods: reachability analysis

Quantification of reachability



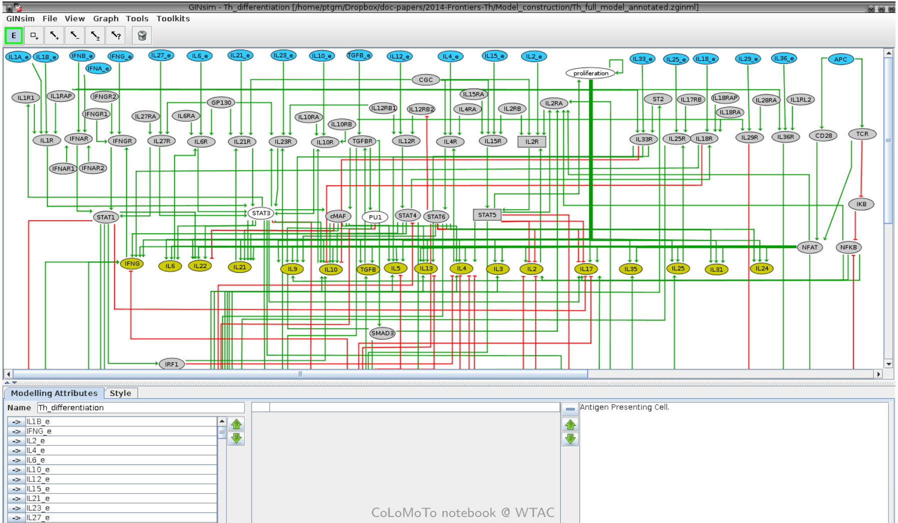
Probabilities of transition (firing rates)
11000
25%
15%
60%
01000
11010
11100
Stochastic simulations
Estimated distribution of state's probabilities

- **Avatar** (Mendes *et al.*, *Frontiers in Physiology* 2018) @(bioLQM&GINsim)
- **MaBoSS** (Stoll *et al.*, *BMC Syst Biol* 2012)

See Vincent Noel's lecture!

Gene Interaction Network simulation - GINsim

Gene Interaction Network simulation is a computer tool for the modeling and simulation of genetic regulatory networks implementing the **logical formalism** http://ginsim.org


GINsim File View Graph Tools Toolkits
E
IL1A_e IL1B_e IFNB_e IFNA_e IFNG_e IL27_e IL6_e IL21_e IL23_e IL10_e TGFB_e IL12_e IL4_e IL15_e IL2_e IL33_e IL25_e IL18_e IL29_e IL36_e APC
proliferation CGC
IL1R1 IL1RAP IFNGR2 IFNGR1 GP130 IL27RA IL6RA IL10RA IL10RB IL12RB1 IL12RB2 IL15RA IL4RA IL2RB IL2RA ST2 IL17RB IL18RAP IL18RA IL28RA IL1RL2
IL1R IFNAR IFNGR IL27R IL6R IL21R IL23R IL10R TGFBR IL12R IL4R IL15R IL2R IL33R IL25R IL18R IL29R IL36R CD28 TCR
IFNAR1 IFNAR2
STAT1 STAT3 cMAF PU1 STAT4 STAT6 STAT5 IKB NFAT NFKB
IFNG IL6 IL22 IL21 IL9 IL10 TGFB IL5 IL13 IL4 IL3 IL2 IL17 IL35 IL25 IL31 IL24
SMAD3 IRF1
Modelling Attributes Style

GINsim - Th_differentiation [/home/ptgm/Dropbox/doc-papers/2014-Frontiers-Th/Model_construction/Th_full_model_annotated.zginml]

Name Th_differentiation

-> IL1B_e
-> IFNG_e
-> IL2_e
-> IL4_e
-> IL6_e
-> IL10_e
-> IL12_e
-> IL15_e
-> IL21_e
-> IL23_e
-> IL27_e

Antigen Presenting Cell.

Gene Interaction Network simulation - GINsim

Gene **I**nteraction **N**etwork **sim**ulation is a computer tool for the modeling and simulation of genetic regulatory networks implementing the **logical formalism** http://ginsim.org

Analysis methods

Static Analysis:

- Stable state search
  (Naldi *et al.*, *CMSB* 2007)
- Circuit analysis
  (Thieffry, *Brief. Bioinform.* 2007)
- Model reduction
  (Naldi *et al.*, *Theor. Comput. Sci.* 2011)

Dynamics:

- STG construction
- HTG construction
  (Bérenguier *et al.*, *Chaos* 2013)
- Priority classes
  (Fauré *et al.*, *Bioinform.* 2006)
- Perturbations
- Reachability (explicit)

Gene Interaction Network simulation - GINsim

Gene **I**nteraction **N**etwork **sim**ulation is a computer tool for the modeling and simulation of genetic regulatory networks implementing the **logical formalism** http://ginsim.org

Analysis methods

Static Analysis:

- Stable state search

  (Naldi *et al.*, *CMSB* 2007)
- Circuit analysis

  (Thieffry, *Brief. Bioinform.* 2007)
- Model reduction

  (Naldi *et al.*, *Theor. Comput. Sci.* 2011)

Dynamics:

- STG construction
- HTG construction

  (Bérenguier *et al.*, *Chaos* 2013)
- Priority classes

  (Fauré *et al.*, *Bioinform.* 2006)
- Perturbations
- Reachability (explicit)

Many other tools exist, though!

Software tools

Trajectory analysis
NuSMV

Network editor
Cell Designer

Model building
CaSQ

Knowledge base
Simulations
The Cell Collective

Attractors
Reachability
Pint

Modification
Visualization
Attractors
bioLQM
GINsim

Exchange format
SBML **qual**

BoolNet
Simulation
Generation
Delays

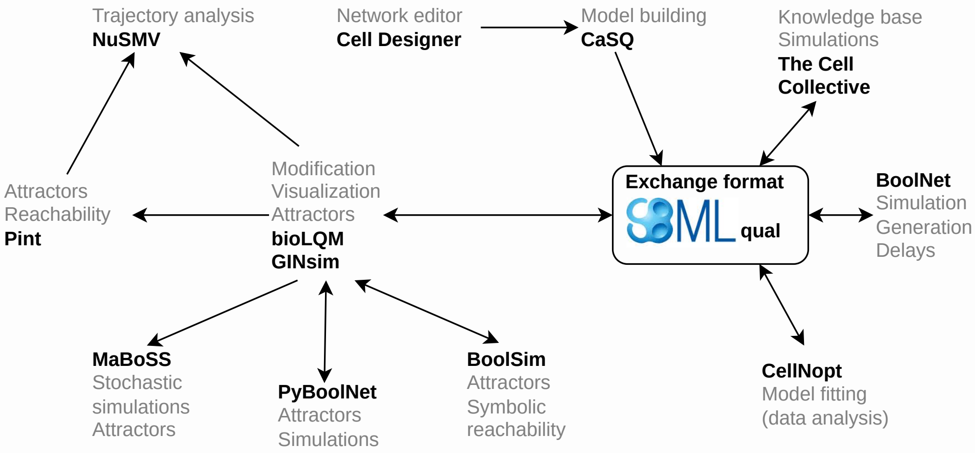
MaBoSS
Stochastic simulations
Attractors

PyBoolNet
Attractors
Simulations

BoolSim
Attractors
Symbolic reachability

CellNopt
Model fitting (data analysis)

Software tools



Trajectory analysis
NuSMV

Network editor
Cell Designer

Model building
CaSQ

Knowledge base
Simulations
The Cell Collective

Attractors
Reachability
Pint

Modification
Visualization
Attractors
bioLQM
GINsim

Exchange format
SBML **qual**

BoolNet
Simulation
Generation
Delays

MaBoSS
Stochastic simulations
Attractors

PyBoolNet
Attractors
Simulations

BoolSim
Attractors
Symbolic reachability

CellNopt
Model fitting
(data analysis)

CoLoMoTo - http://colomoto.org

Consortium for **Lo**gical **Mo**dels and **To**ols

Install

Conda packages, Docker image

Perform analysis

Python API for all tools

Jupyter: semi-interactive interface

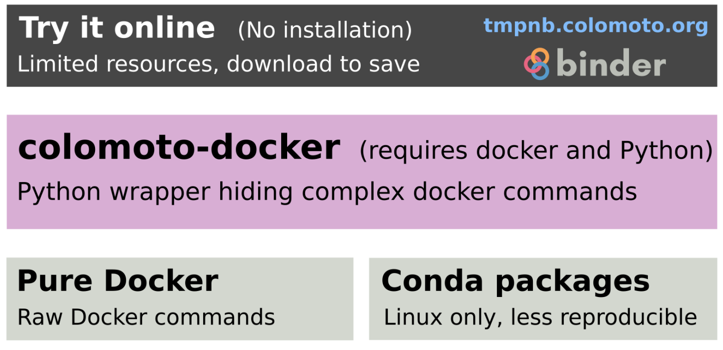
Share

Analysis: Jupyter notebooks

Models: SBML qual exchange format

(Naldi *et al.*, *Bioinformatics* 2015)

CoLoMoTo notebook - http://colomoto.org/notebook



Try it online (No installation) **tmpnb.colomoto.org**
Limited resources, download to save **binder**

colomoto-docker (requires docker and Python)
Python wrapper hiding complex docker commands

Pure Docker
Raw Docker commands

Conda packages
Linux only, less reproducible

(Naldi *et al.*, *Frontiers in Physiology* 2018)

Maintained by:

Initially

Aurélien Naldi

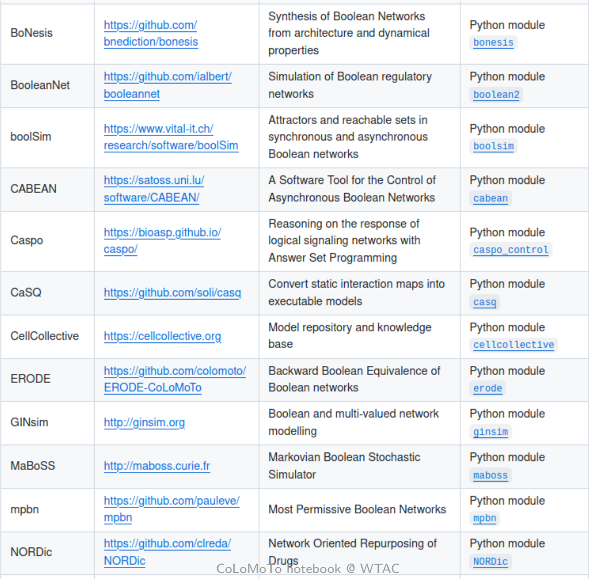
Sporadically

Pedro Monteiro

Mainly

Loic Paulevé

CoLoMoTo software tools - https://colomoto.github.io/colomoto-docker/



BoNesis	https://github.com/bnediction/bonesis	Synthesis of Boolean Networks from architecture and dynamical properties	Python module `bonesis`
BooleanNet	https://github.com/ialbert/booleannet	Simulation of Boolean regulatory networks	Python module `boolean2`
boolSim	https://www.vital-it.ch/research/software/boolSim	Attractors and reachable sets in synchronous and asynchronous Boolean networks	Python module `boolsim`
CABEAN	https://satoss.uni.lu/software/CABEAN/	A Software Tool for the Control of Asynchronous Boolean Networks	Python module `cabean`
Caspo	https://bioasp.github.io/caspo/	Reasoning on the response of logical signaling networks with Answer Set Programming	Python module `caspo_control`
CaSQ	https://github.com/soli/casq	Convert static interaction maps into executable models	Python module `casq`
CellCollective	https://cellcollective.org	Model repository and knowledge base	Python module `cellcollective`
ERODE	https://github.com/colomoto/ERODE-CoLoMoTo	Backward Boolean Equivalence of Boolean networks	Python module `erode`
GINsim	http://ginsim.org	Boolean and multi-valued network modelling	Python module `ginsim`
MaBoSS	http://maboss.curie.fr	Markovian Boolean Stochastic Simulator	Python module `maboss`
mpbn	https://github.com/pauleve/mpbn	Most Permissive Boolean Networks	Python module `mpbn`
NORDic	https://github.com/clreda/NORDic	Network Oriented Repurposing of Drugs	Python module `NORDic`

Software tools

Problems

- Installation problems (OS, dependencies, ...)
- Interaction between tools (formats, versions, ...)
- Correct order of steps/commands
- Which parameters each tool
- . . .

Software tools

Problems

- Installation problems (OS, dependencies, ...)
- Interaction between tools (formats, versions, ...)
- Correct order of steps/commands
- Which parameters each tool
- . . .

CoLoMoTo docker/notebook

- Integrated access to (pre-installed) tools
- Interoperability between tools
- Reproducibility of analyses

CoLoMoTo docker

https://github.com/colomoto/colomoto-docker

Support: Linux, MacOS, Windows

Requirements: Docker, Python

Installation (or update)

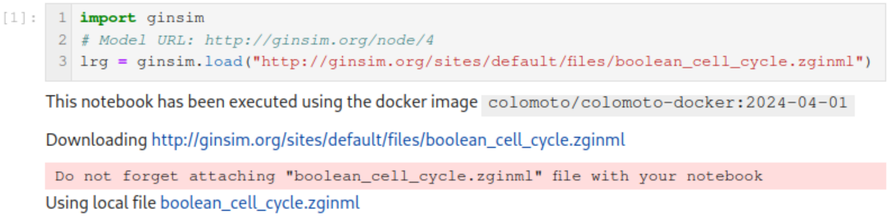
```
pip install -U colomoto-docker
```

Usage (start the notebook)

```
colomoto-docker                    uses the most recently fetched image
colomoto-docker -V latest               fetches the latest published image
colomoto-docker -V 2018-05-29                     fetches a specific image
```

CoLoMoTo notebook - http://colomoto.org/notebook

Load model



[1]:

```
import ginsim
# Model URL: http://ginsim.org/node/4
lrg = ginsim.load("http://ginsim.org/sites/default/files/boolean_cell_cycle.zginml")
```

This notebook has been executed using the docker image `colomoto/colomoto-docker:2024-04-01`

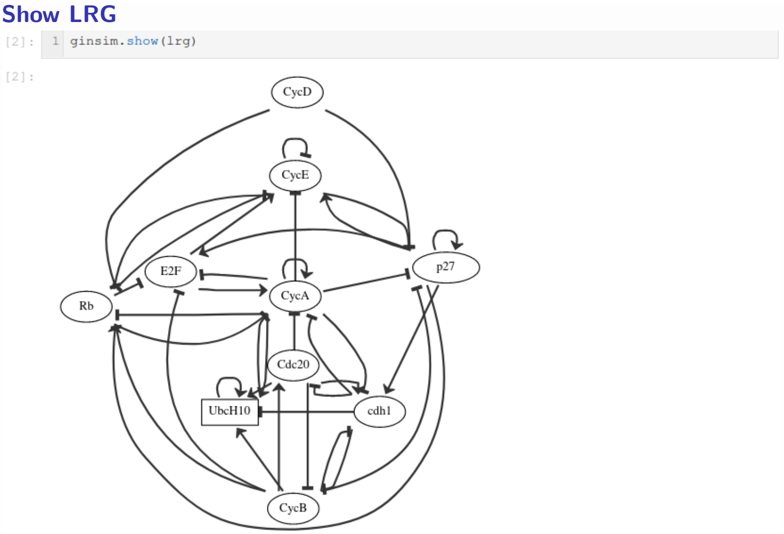
Downloading http://ginsim.org/sites/default/files/boolean_cell_cycle.zginml

```
Do not forget attaching "boolean_cell_cycle.zginml" file with your notebook
```

Using local file boolean_cell_cycle.zginml

(model is saved in a local directory)

```
colomoto-docker --bind .                the notebook uses the current directory
```



Show LRG

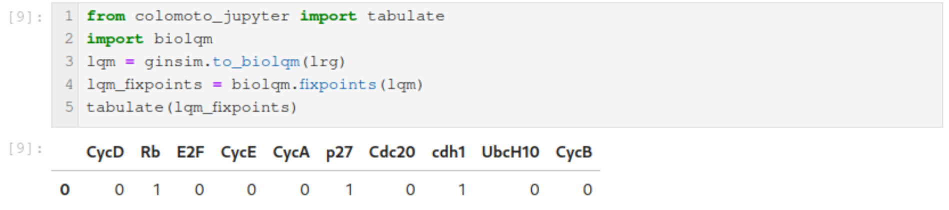
```
[2]: ginsim.show(lrg)
```

[2]:

CoLoMoTo notebook - http://colomoto.org/notebook

Attractor identification

bioLQM



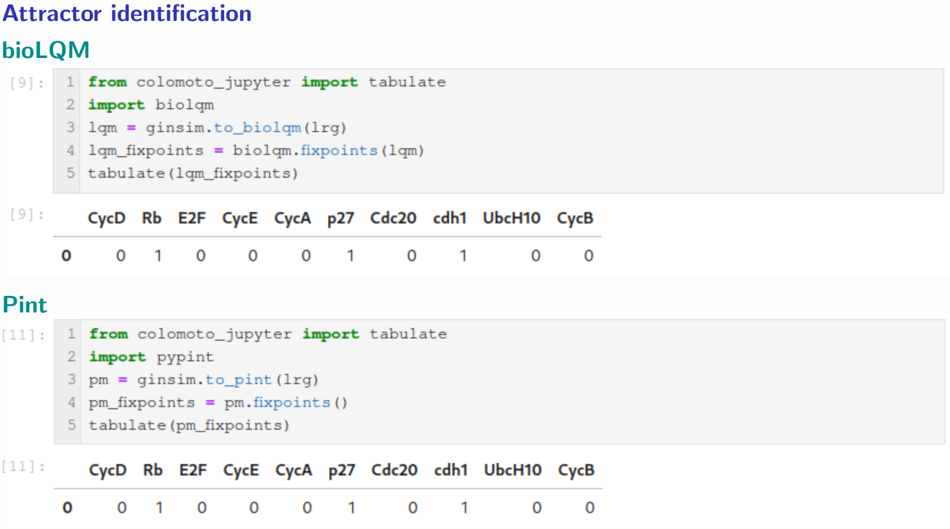
[9]:

```python
from colomoto_jupyter import tabulate
import biolqm
lqm = ginsim.to_biolqm(lrg)
lqm_fixpoints = biolqm.fixpoints(lqm)
tabulate(lqm_fixpoints)
```

[9]:

	CycD	Rb	E2F	CycE	CycA	p27	Cdc20	cdh1	UbcH10	CycB
0	0	1	0	0	0	1	0	1	0	0

CoLoMoTo notebook - http://colomoto.org/notebook



Attractor identification

bioLQM

[9]:

```python
from colomoto_jupyter import tabulate
import biolqm
lqm = ginsim.to_biolqm(lrg)
lqm_fixpoints = biolqm.fixpoints(lqm)
tabulate(lqm_fixpoints)
```

[9]:

	CycD	Rb	E2F	CycE	CycA	p27	Cdc20	cdh1	UbcH10	CycB
0	0	1	0	0	0	1	0	1	0	0

Pint

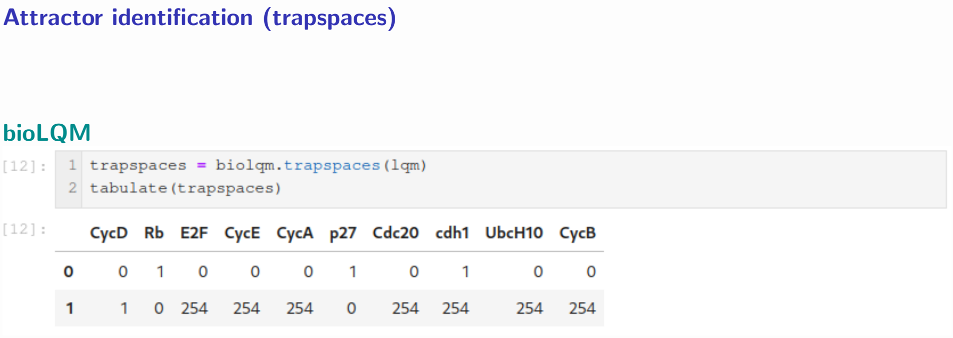
[11]:

```python
from colomoto_jupyter import tabulate
import pypint
pm = ginsim.to_pint(lrg)
pm_fixpoints = pm.fixpoints()
tabulate(pm_fixpoints)
```

[11]:

	CycD	Rb	E2F	CycE	CycA	p27	Cdc20	cdh1	UbcH10	CycB
0	0	1	0	0	0	1	0	1	0	0

CoLoMoTo notebook - http://colomoto.org/notebook



Attractor identification (trapspaces)

bioLQM

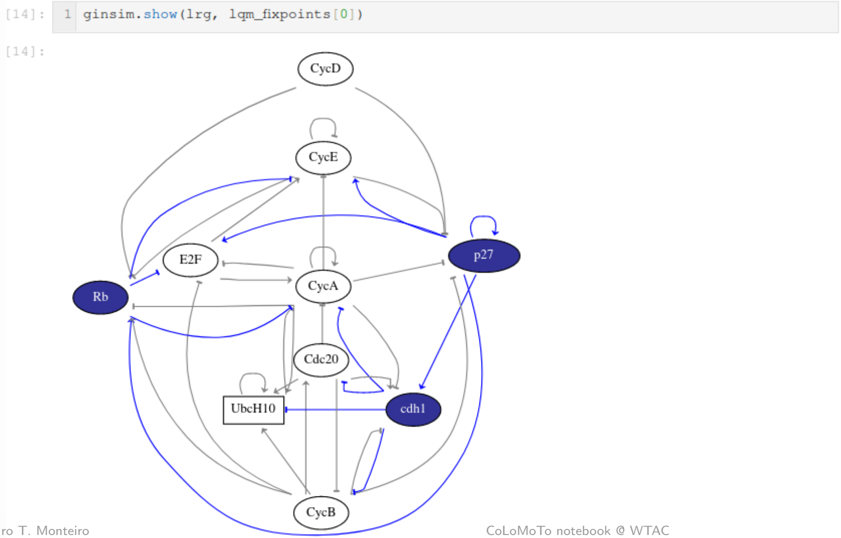
[12]:

```
trapspaces = biolqm.trapspaces(lqm)
tabulate(trapspaces)
```

[12]:

	CycD	Rb	E2F	CycE	CycA	p27	Cdc20	cdh1	UbcH10	CycB
0	0	1	0	0	0	1	0	1	0	0
1	1	0	254	254	254	0	254	254	254	254

Map attractors on model LRG (stable states)

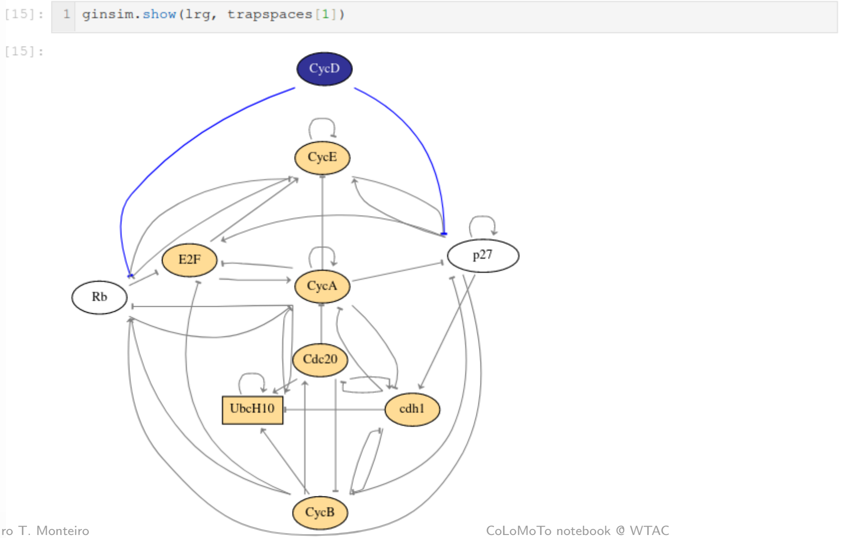


```
[14]: ginsim.show(lrg, lqm_fixpoints[0])
```

```
[14]:
```

CoLoMoTo notebook - http://colomoto.org/notebook

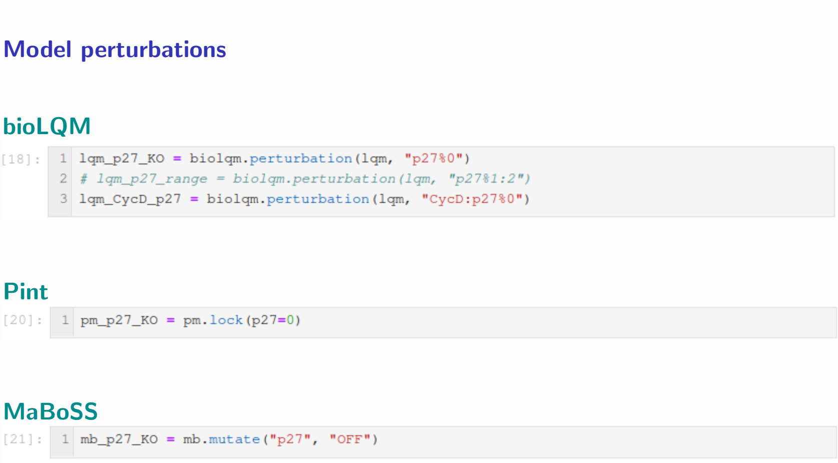
Map attractors on model LRG (trap spaces)



```
[15]: ginsim.show(lrg, trapspaces[1])
```

[15]:

CoLoMoTo notebook - http://colomoto.org/notebook



Model perturbations

bioLQM

```
[18]: lqm_p27_KO = biolqm.perturbation(lqm, "p27%0")
      # lqm_p27_range = biolqm.perturbation(lqm, "p27%1:2")
      lqm_CycD_p27 = biolqm.perturbation(lqm, "CycD:p27%0")
```

Pint

```
[20]: pm_p27_KO = pm.lock(p27=0)
```

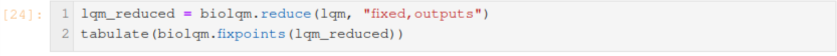
MaBoSS

```
[21]: mb_p27_KO = mb.mutate("p27", "OFF")
```

CoLoMoTo notebook - http://colomoto.org/notebook

Model reduction

bioLQM



[24]:

```
lqm_reduced = biolqm.reduce(lqm, "fixed,outputs")
tabulate(biolqm.fixpoints(lqm_reduced))
```

(Naldi et al., *TCS* 2011)

(Naldi et al., *CMSB* 2012)

CoLoMoTo notebook - http://colomoto.org/notebook

Formal reachability analysis

Pint

```
an = biolqm.to_pint(lqm)
target_state = {"Apoptosis":1}
an.reachability(goal=target_state)
```

```
True
```

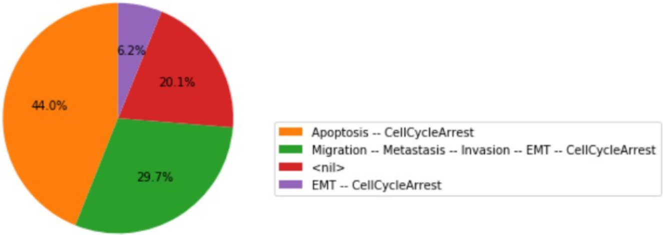
NuSMV model checker

```
smv = an.to_nusmv()
ctl_specs = {
    "stable-apoptosis": EF(AG(S(Apoptosis=1)))
}
smv.add_ctls(ctl_specs)
smv.verify()
```

```
{'stable-apoptosis': False}
```

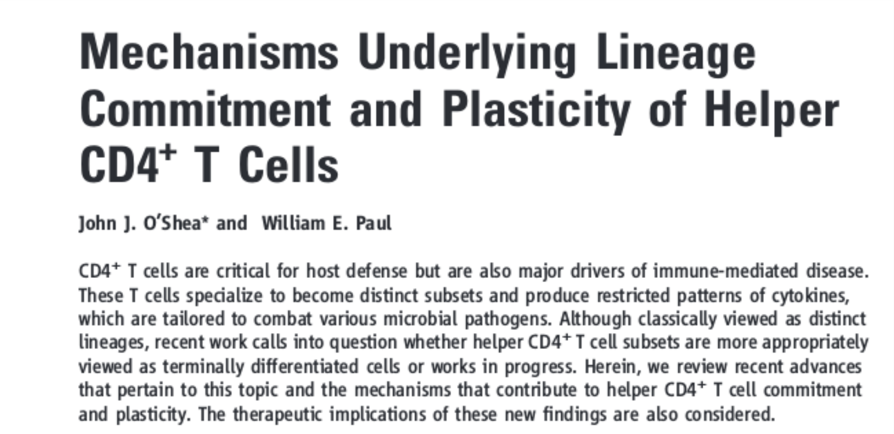
Estimation of reachability probabilities

```
masim = biolqm.to_maboss(lqm)
mares = masim.run()
mares.plot_piechart()
```



Break....

Case study - T-helper cell differentiation



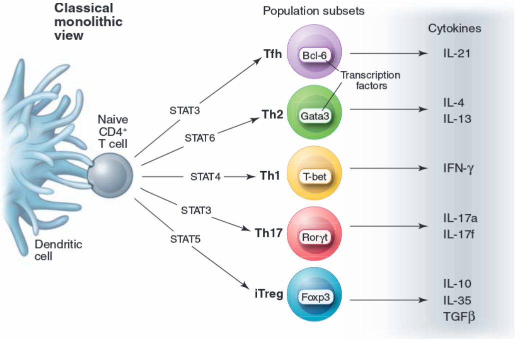
Mechanisms Underlying Lineage Commitment and Plasticity of Helper CD4$^+$ T Cells

John J. O'Shea* and William E. Paul

CD4$^+$ T cells are critical for host defense but are also major drivers of immune-mediated disease. These T cells specialize to become distinct subsets and produce restricted patterns of cytokines, which are tailored to combat various microbial pathogens. Although classically viewed as distinct lineages, recent work calls into question whether helper CD4$^+$ T cell subsets are more appropriately viewed as terminally differentiated cells or works in progress. Herein, we review recent advances that pertain to this topic and the mechanisms that contribute to helper CD4$^+$ T cell commitment and plasticity. The therapeutic implications of these new findings are also considered.

(O'Shea and Paul, *Science* 2010)

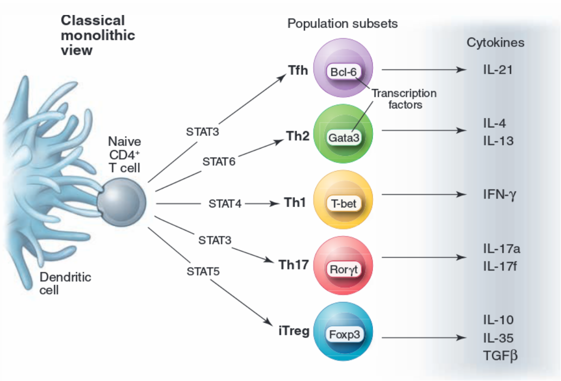
Case study - T-helper cell differentiation

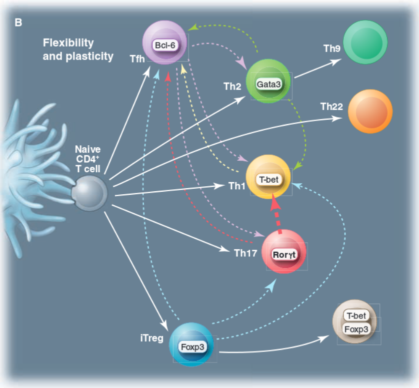


- T-helper (CD4+) lymphocytes play a role in the regulation of immune responses
- Faced with pathogens, naive CD4+ T cells differentiate into distinct Th cell populations expressing distinct cytokine profiles
- Polarization into cell types is governed by transcription factors

(O'Shea and Paul, *Science* 2010)

Case study - T-helper cell differentiation





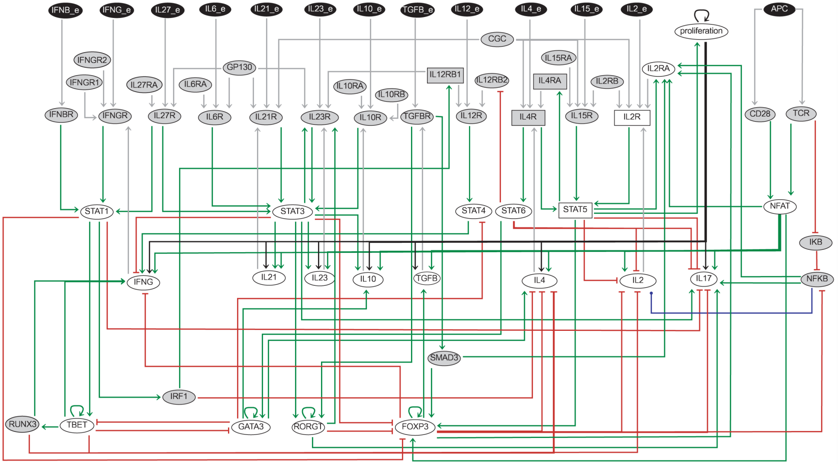
Objective

- Model capable of reproducing Th cell differentiation

  (Naldi *et al.*, *PLoS Comput. Biol.* 2010)

- Capable of exhibiting plasticity behaviours

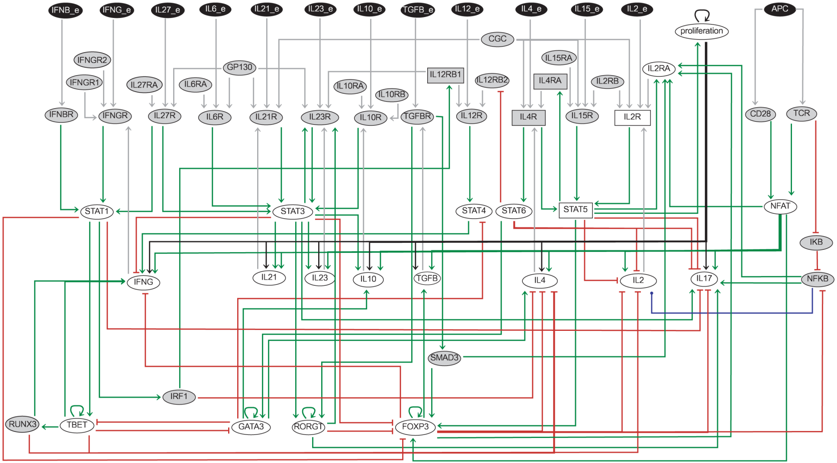
Logical modelling of the Th network



(Naldi et al., *PLoS Comp Biol* 2010)

Model size: 64 components (13 inputs + 51 internal) and 138 interactions

Logical modelling of the Th network



(Naldi et al., *PLoS Comp Biol* 2010)

Model size: 64 components (13 inputs + 51 internal) and 138 interactions

STG size: 2^{64} states!!

⇒ **Too large to perform meaningful simulations!**

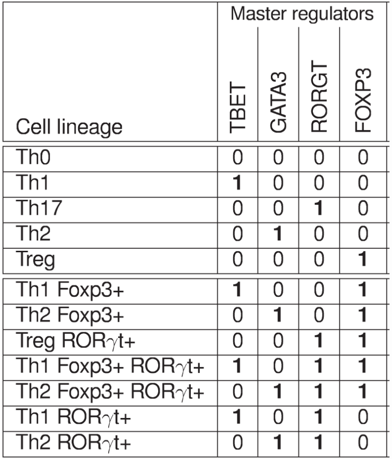
Logical modelling of the Th network

Approach to analyse the T-helper cell differentiation model:

- Identify the biologically relevant **states** corresponding to **Th cell subtypes**
- Compute the **reprogramming and plasticity** between **Th cell subtypes**

Case study - T-helper cell differentiation

Start by computing the set of biologically relevant **Th cell subtypes**

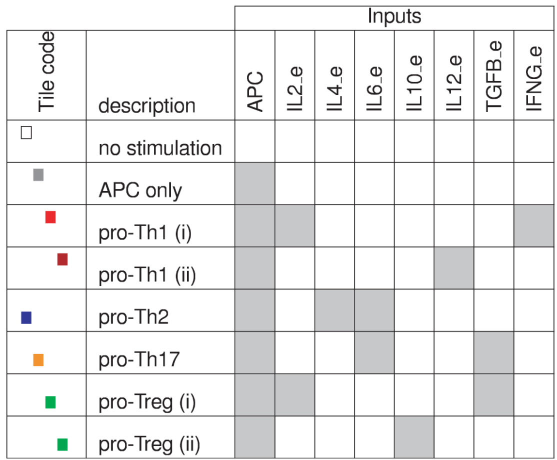


	Master regulators			
Cell lineage	TBET	GATA3	RORGT	FOXP3
Th0	0	0	0	0
Th1	**1**	0	0	0
Th17	0	0	**1**	0
Th2	0	**1**	0	0
Treg	0	0	0	**1**
Th1 Foxp3+	**1**	0	0	**1**
Th2 Foxp3+	0	**1**	0	**1**
Treg RORγt+	0	0	**1**	**1**
Th1 Foxp3+ RORγt+	**1**	0	**1**	**1**
Th2 Foxp3+ RORγt+	0	**1**	**1**	**1**
Th1 RORγt+	**1**	0	**1**	0
Th2 RORγt+	0	**1**	**1**	0

(Naldi et al., *PLoS Comp Biol* 2010)

Case study - T-helper cell differentiation

Then characterize the set of prototypic **environmental conditions**

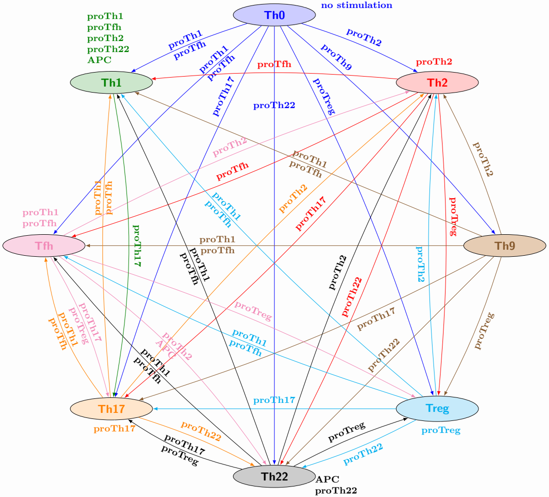


		Inputs							
Tile code	description	APC	IL2_e	IL4_e	IL6_e	IL10_e	IL12_e	TGFB_e	IFNG_e
□	no stimulation								
■	APC only	■							
🟥	pro-Th1 (i)	■	■						■
🟥	pro-Th1 (ii)	■					■		
🟦	pro-Th2	■		■	■				
🟧	pro-Th17	■			■			■	
🟩	pro-Treg (i)	■	■					■	
🟩	pro-Treg (ii)	■				■			

(Naldi et al., *PLoS Comp Biol* 2010)

Case study - Reachability analysis of relevant Th cell subtypes

Use **model checking** to systematically explore the reachability of all **Th cell subtypes** under all **environmental/input conditions**. Obtaining a “reprogramming map” between Th cell subtypes.

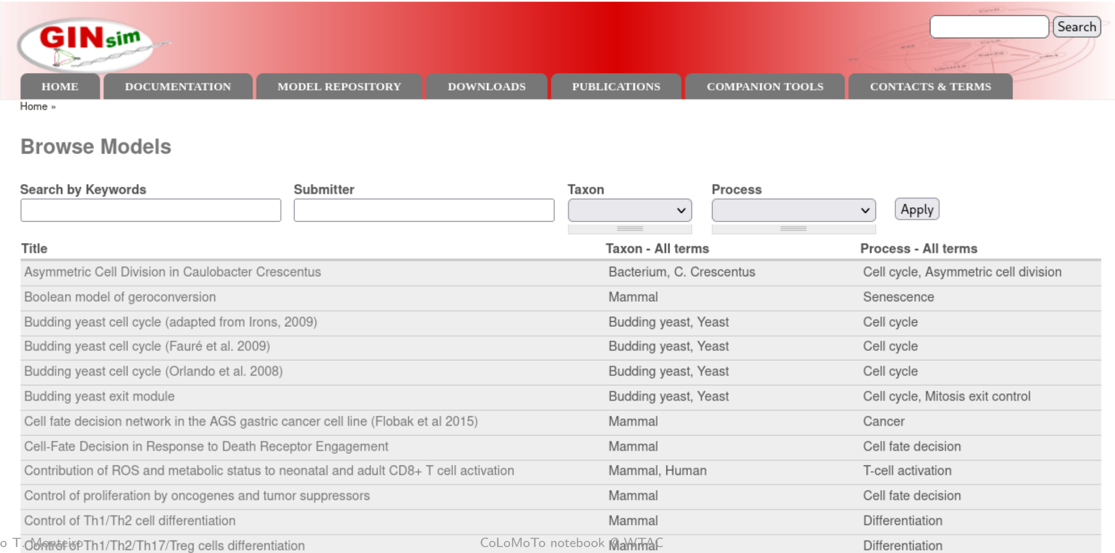


(Abou-Jaoude et al., *Frontiers Bioeng Biotech* 2015)

Gene Interaction Network simulation - GINsim

Model repository

http://ginsim.org/models_repository



Browse Models

Title	Taxon - All terms	Process - All terms
Asymmetric Cell Division in Caulobacter Crescentus	Bacterium, C. Crescentus	Cell cycle, Asymmetric cell division
Boolean model of geroconversion	Mammal	Senescence
Budding yeast cell cycle (adapted from Irons, 2009)	Budding yeast, Yeast	Cell cycle
Budding yeast cell cycle (Fauré et al. 2009)	Budding yeast, Yeast	Cell cycle
Budding yeast cell cycle (Orlando et al. 2008)	Budding yeast, Yeast	Cell cycle
Budding yeast exit module	Budding yeast, Yeast	Cell cycle, Mitosis exit control
Cell fate decision network in the AGS gastric cancer cell line (Flobak et al 2015)	Mammal	Cancer
Cell-Fate Decision in Response to Death Receptor Engagement	Mammal	Cell fate decision
Contribution of ROS and metabolic status to neonatal and adult CD8+ T cell activation	Mammal, Human	T-cell activation
Control of proliferation by oncogenes and tumor suppressors	Mammal	Cell fate decision
Control of Th1/Th2 cell differentiation	Mammal	Differentiation
Control of Th1/Th2/Th17/Treg cells differentiation	Mammal	Differentiation

Hands On

- Load the Th model from GINsim model repository
  `http://ginsim.org/sites/default/files/Frontiers-Th-Full-model-annotated.zginml`
  but for pratical reasons (time/memory) we'll use the older/simpler model
  `http://ginsim.org/sites/default/files/Th_17.zginml` (Mendoza 2006)
  `http://ginsim.org/sites/default/files/Th_differentiation_reduced_model.zginml` (Naldi 2010)

- Display the regulatory graph

- Compute all stable states using the `biolqm` (and then `Pint`) python modules

- Visualise attractor on LRG

- Perform reachability analysis using `boolsim`, `NuSMV`, `Pint`, ...

Acknowledgements

Curie, Paris

CoLoMoTo

INRIA, Paris

IBENS, Paris

I2M, Marseille

LaBRI, Bordeaux

IGC, Oeiras

Contacts

http://pedromonteiro.org

Pedro.Tiago.Monteiro@tecnico.ulisboa.pt