Reengineering Biological Systems via Boolean Networks

IV105 Bionformatics Seminar

Faculty of Informatics

David Šafránek, 12.10.2022

sybila
systems biology laboratory

Motivation

- To reveal how phenotypes emerge from molecular interactions
- To cope with incomplete information on molecular interactions

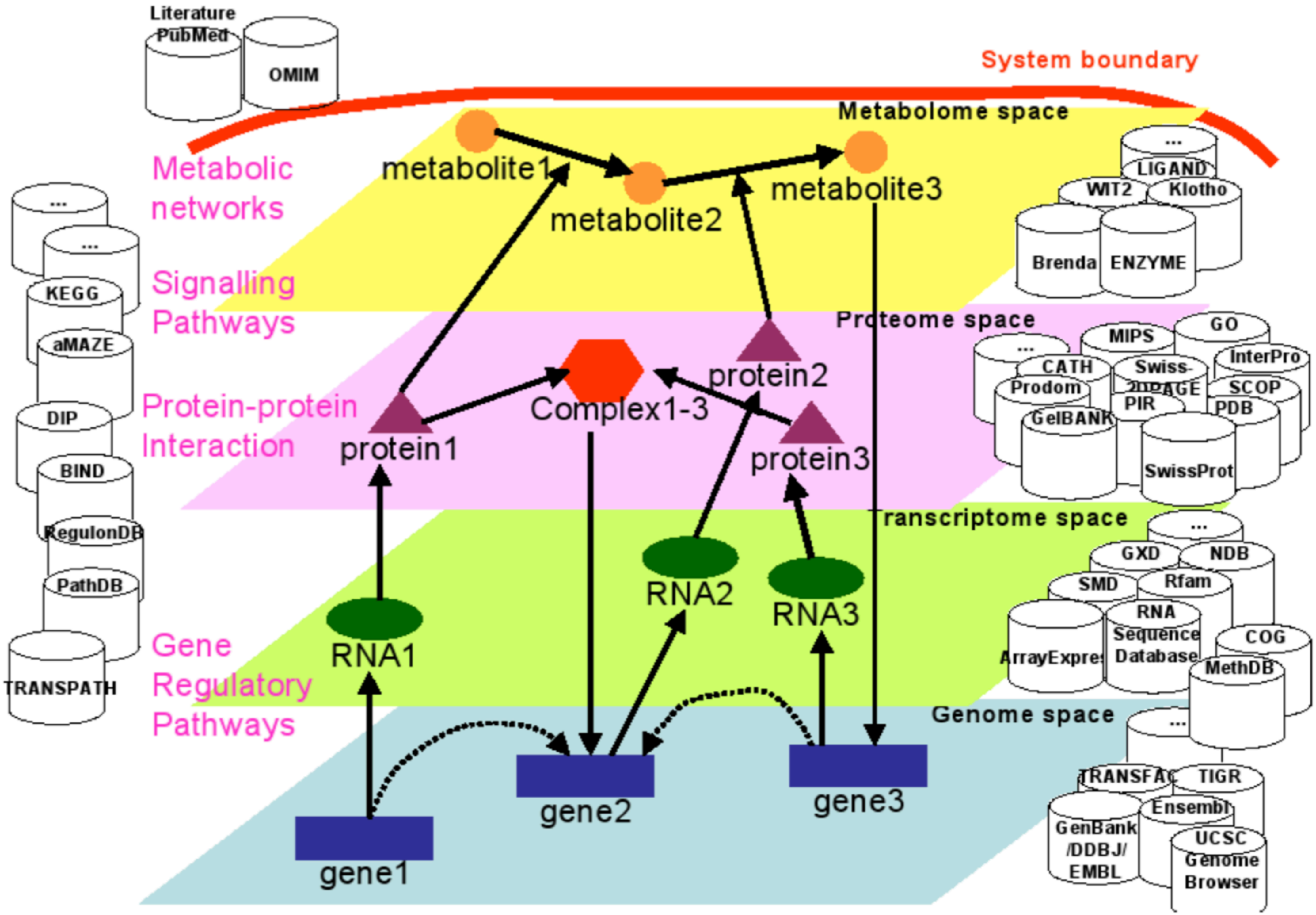
THEORETICAL

- To provide a robust modelling framework
    - Maximise information gain from models
    - Make models capable of guiding experimental design
    - Use models to design control/reprogramming strategies

TECHNICAL

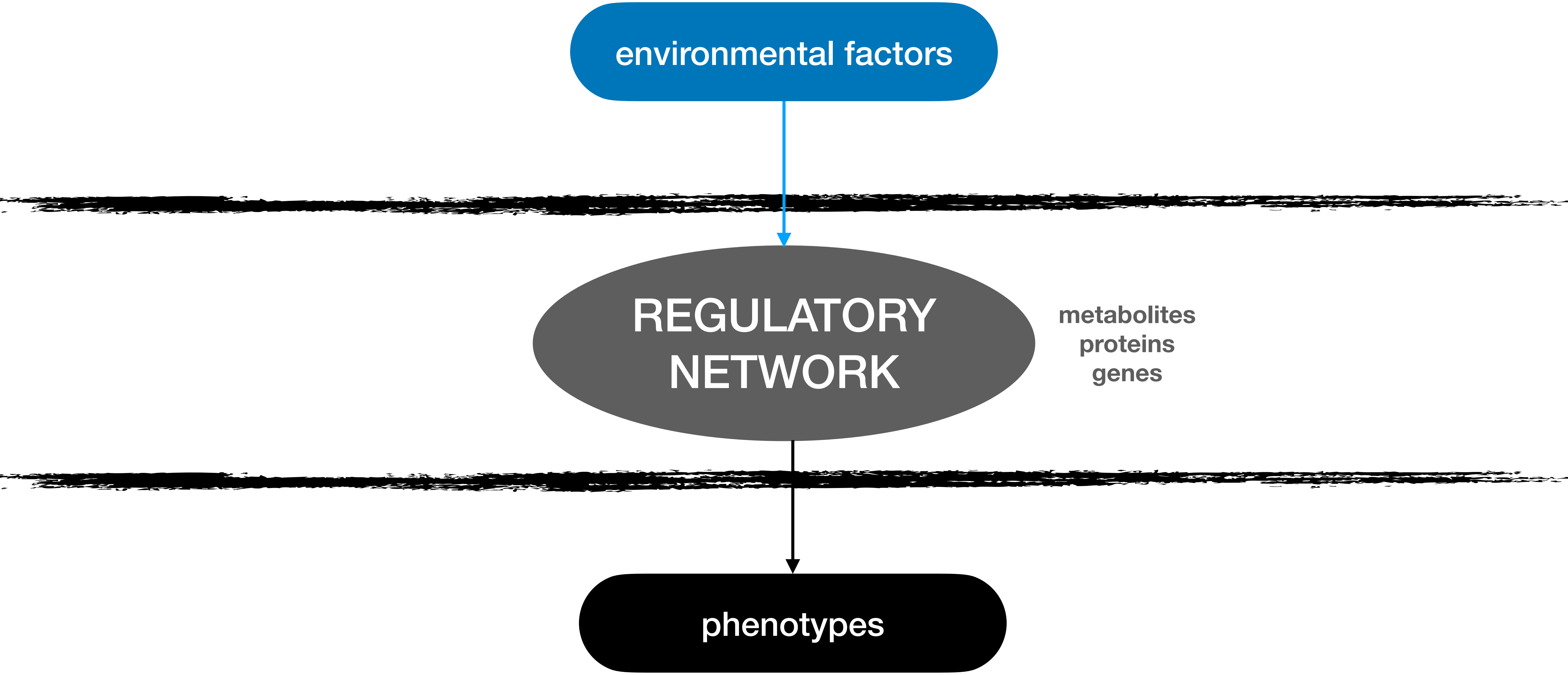
Systems View of a Living Cell

Chemical Interactions Regulate Information Flow



slide credits: David Gilbert (Brunel Univ.)

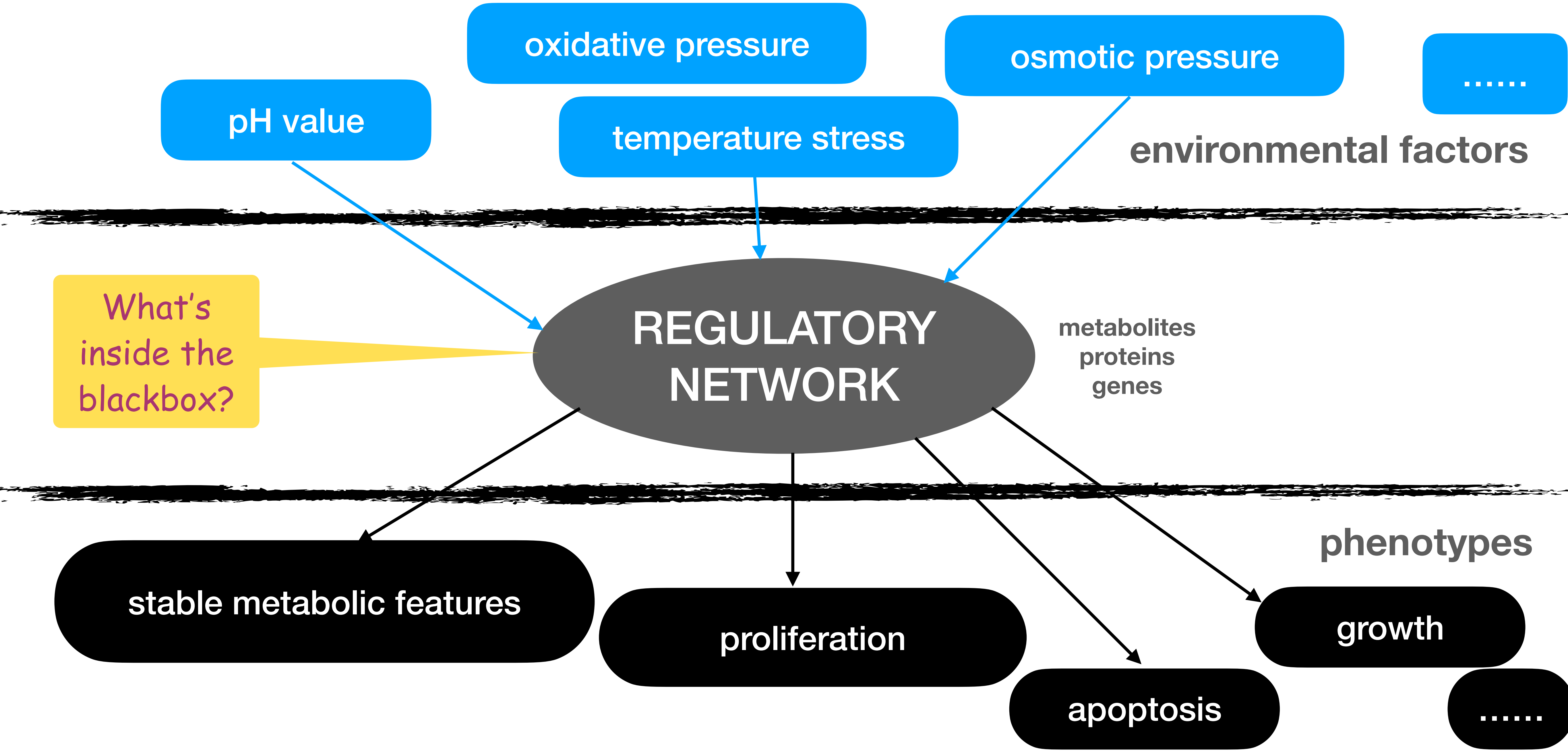
Regulatory Networks



Protein interactions and gene expression regulation compute the systems response…

Modelling Regulatory Networks

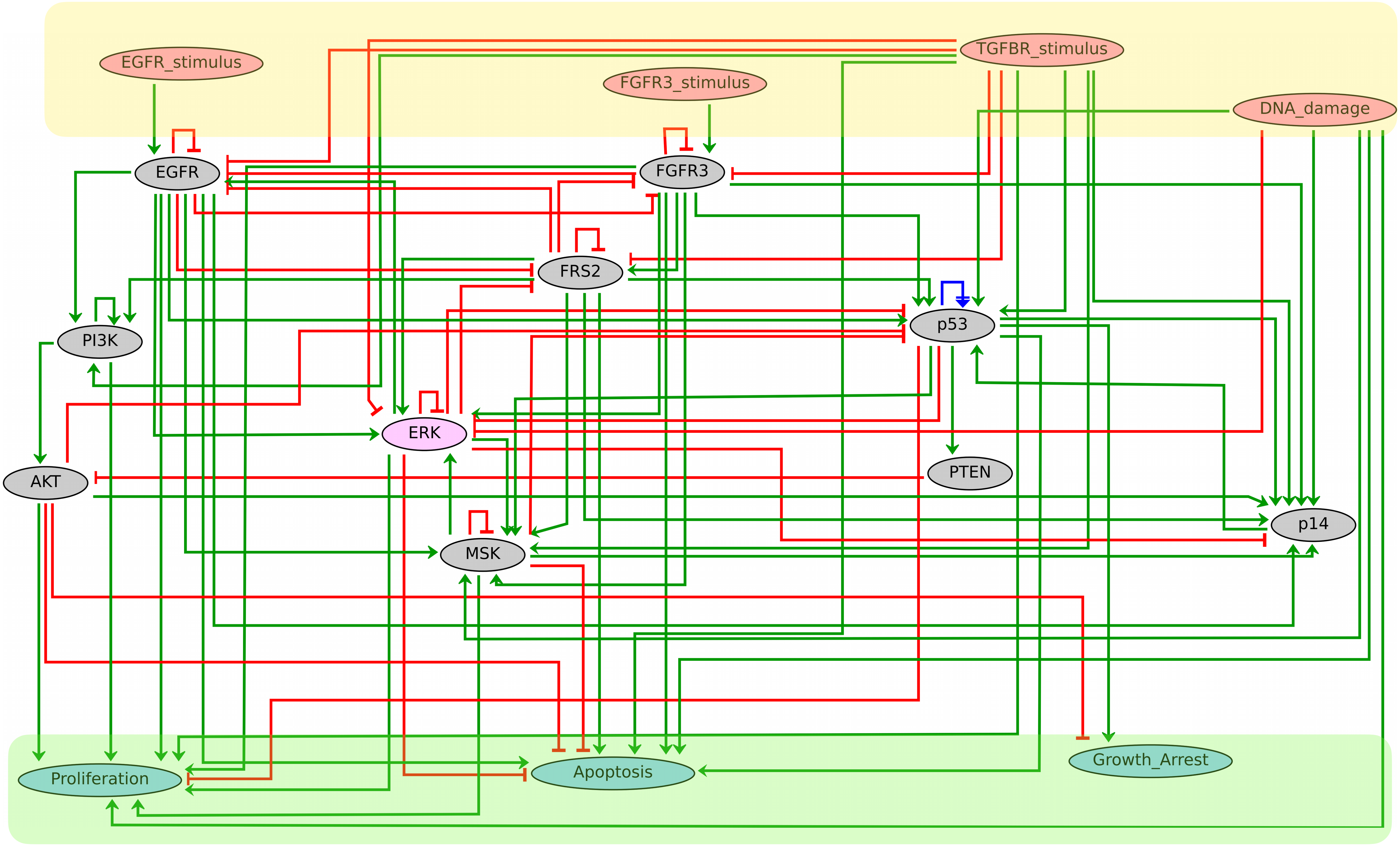
Reengineer Mechanisms Controlling Systems Response



Identify key input and regulatory nodes affecting particular phenotypes.

Example: MAPK signalling

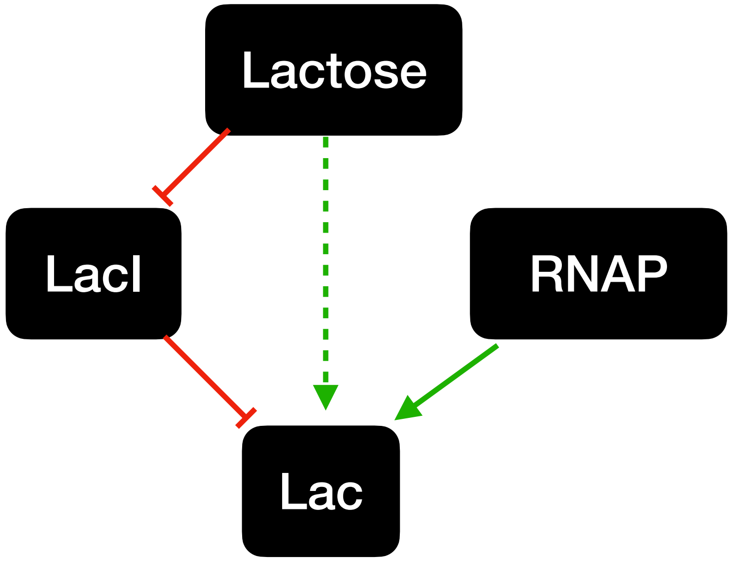
Phenotype Regulation in Cancer Cells [Grieco et al., Plos Comp. Bio. 2013]



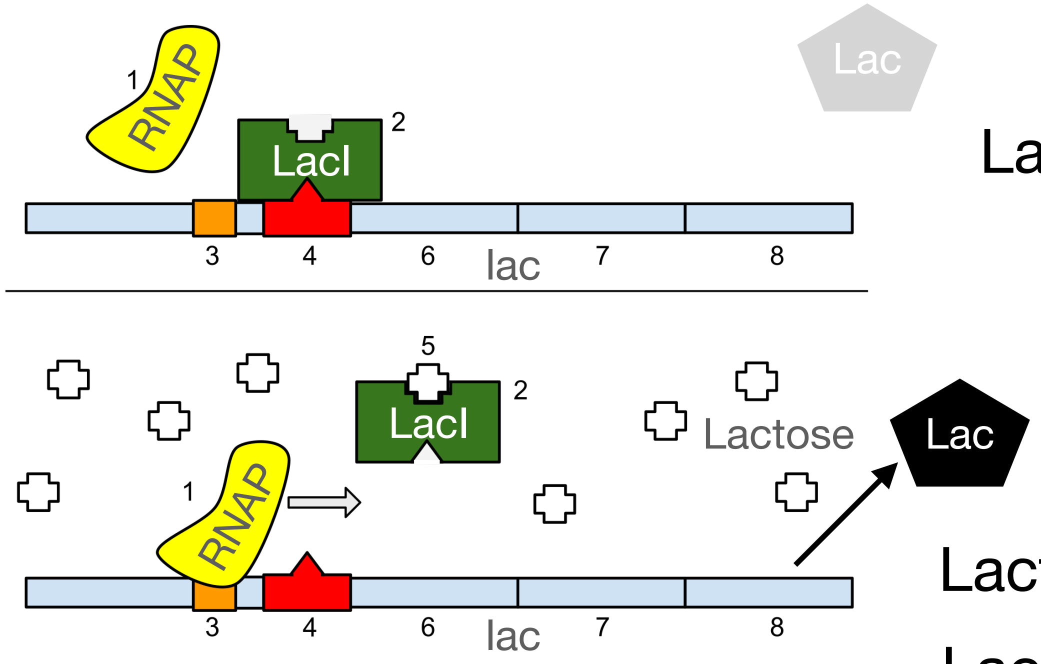
Modelled Mechanisms

Some Basic Examples of Regulatory Interactions

- Gene expression
- Protein activation/deactivation
- Catalysis



Regulatory Network



LacI ⊣ Lac

Lactose ⇢ Lac

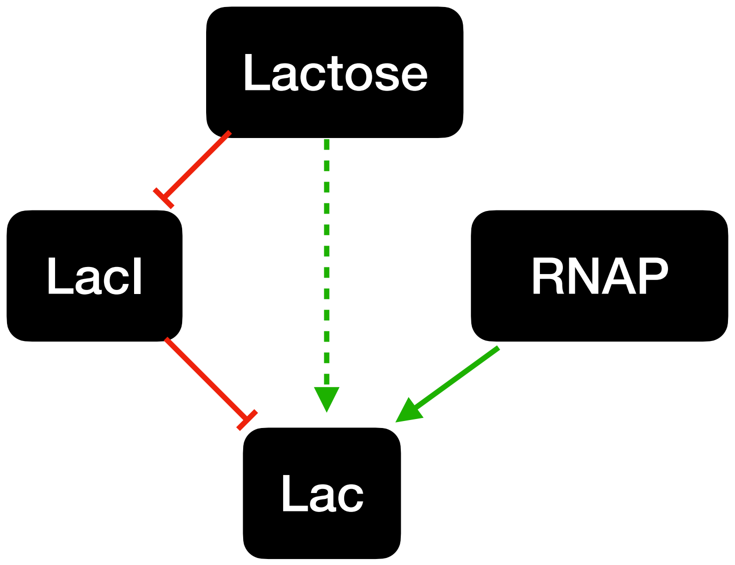
Lactose ⊣ LacI

RNAP → Lac

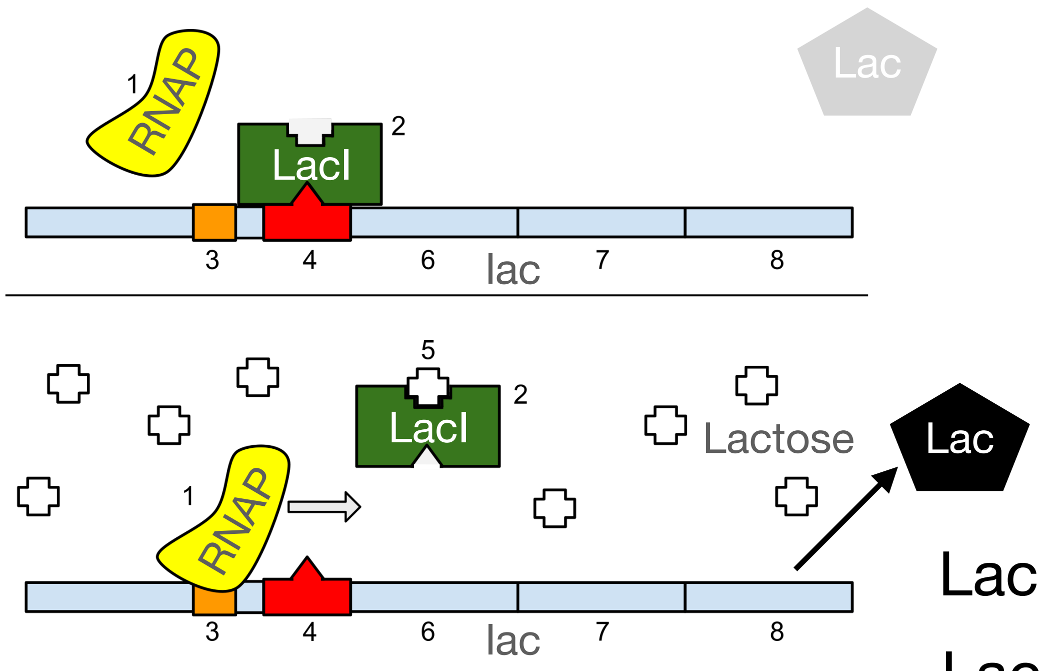
Modelled Mechanisms

Dynamics Driven by Regulatory Interactions

- Gene expression
- Protein activation/deactivation
- Catalysis



Regulatory Network



Rules driving the systems dynamics in time

Lac(t+1) = !LacI(t) AND RNAP(t)

Lactose ⇢ Lac

Lactose ⊣ LacI

RNAP → Lac

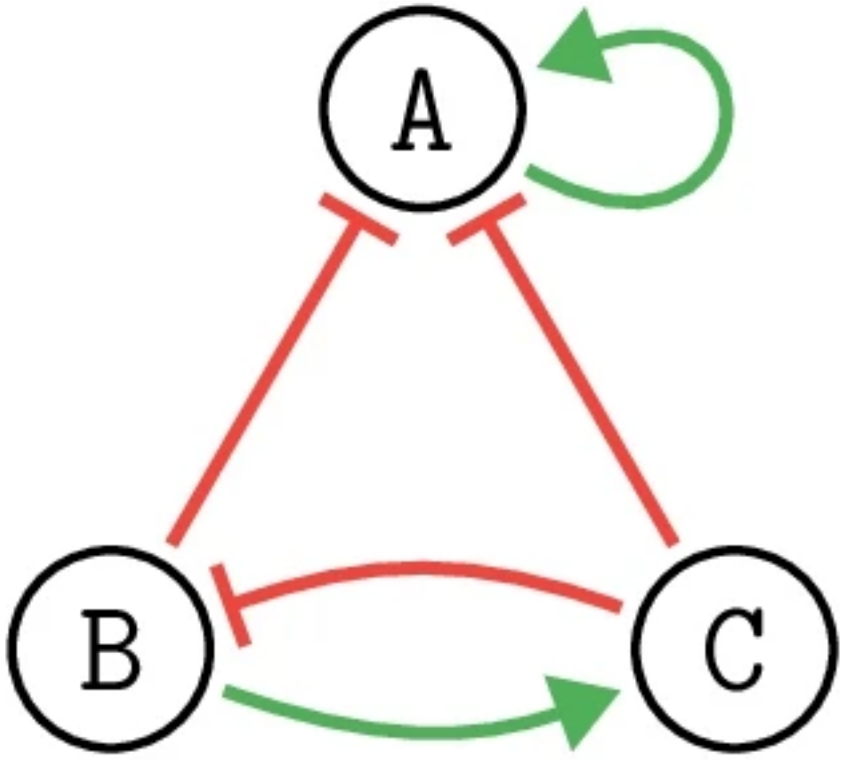
LacI ⊣ Lac

Dynamics of Regulatory Networks

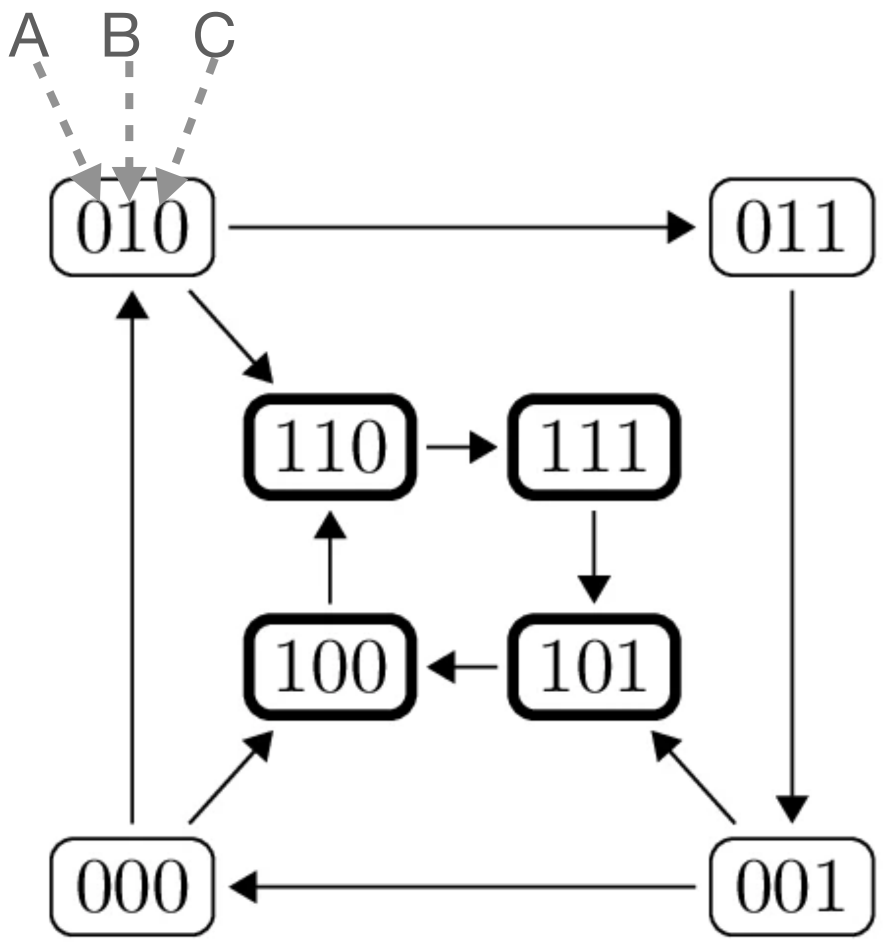
Boolean Networks

[Thomas et al., Bull. of. Math. Biol. (1995)]

$F_{\mathtt{A}} = \mathtt{A} \vee \neg\mathtt{B} \vee \neg\mathtt{C}$



$F_{\mathtt{B}} = \neg\mathtt{C}$ $F_{\mathtt{C}} = \mathtt{B}$



regulatory network + update logics => system dynamics

Boolean Models of RNs

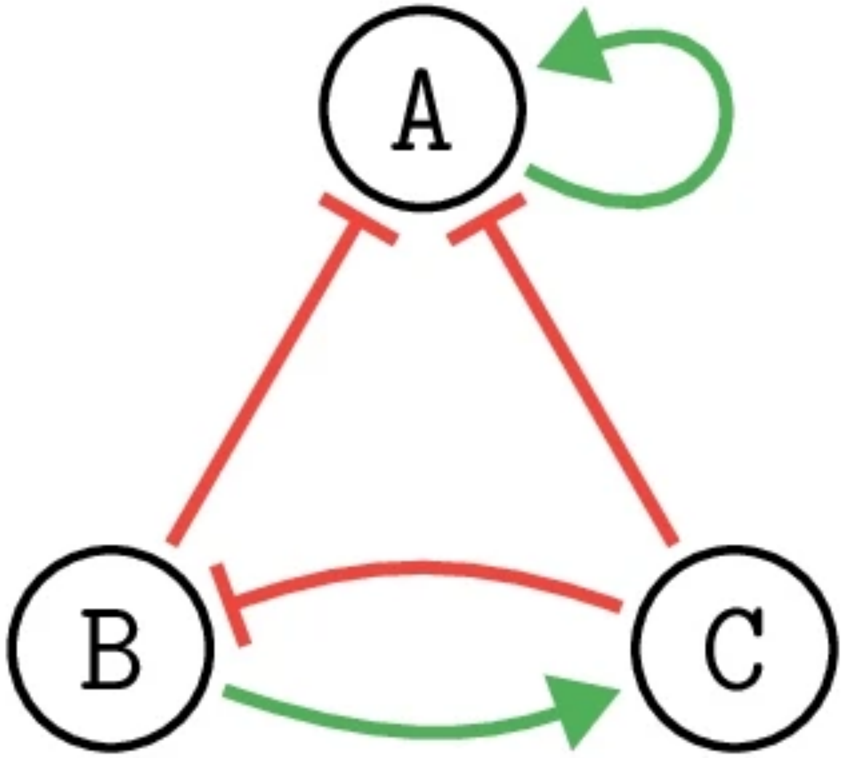
Boolean Networks

- representing regulatory dynamics abstractly:
  - qualitative states: 0/1 -- gene OFF/ON
    (not expressed/expressed)
  - **Boolean semantics** of system variables (Boolean logics)
  - discrete (Boolean) dynamics in discrete time-steps (instead of real time we assume time-steps of unspecified duration)
  - parallel update of variables (expression of individual genes occurs simultaneously in time => various **update schemes**)

Boolean Models of RNs

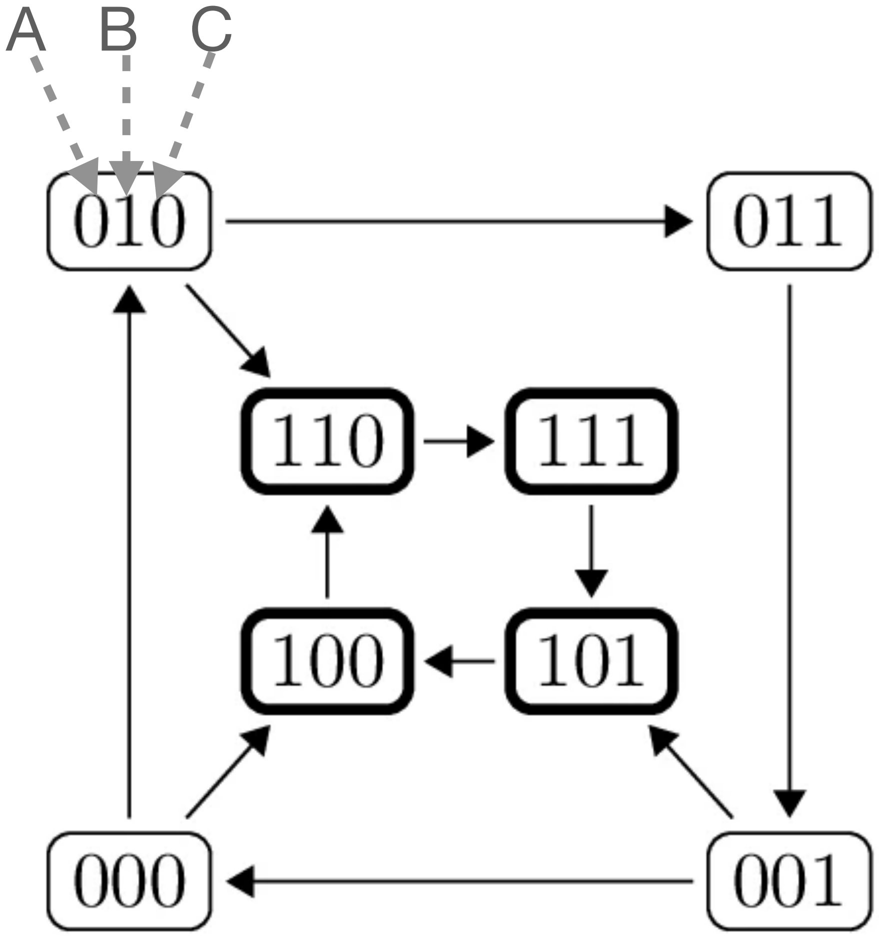
Boolean Networks

$F_{\mathtt{A}} = \mathtt{A} \vee \neg\mathtt{B} \vee \neg\mathtt{C}$



$F_{\mathtt{B}} = \neg\mathtt{C}$ $F_{\mathtt{C}} = \mathtt{B}$

regulatory network + update logics



asynchronous update

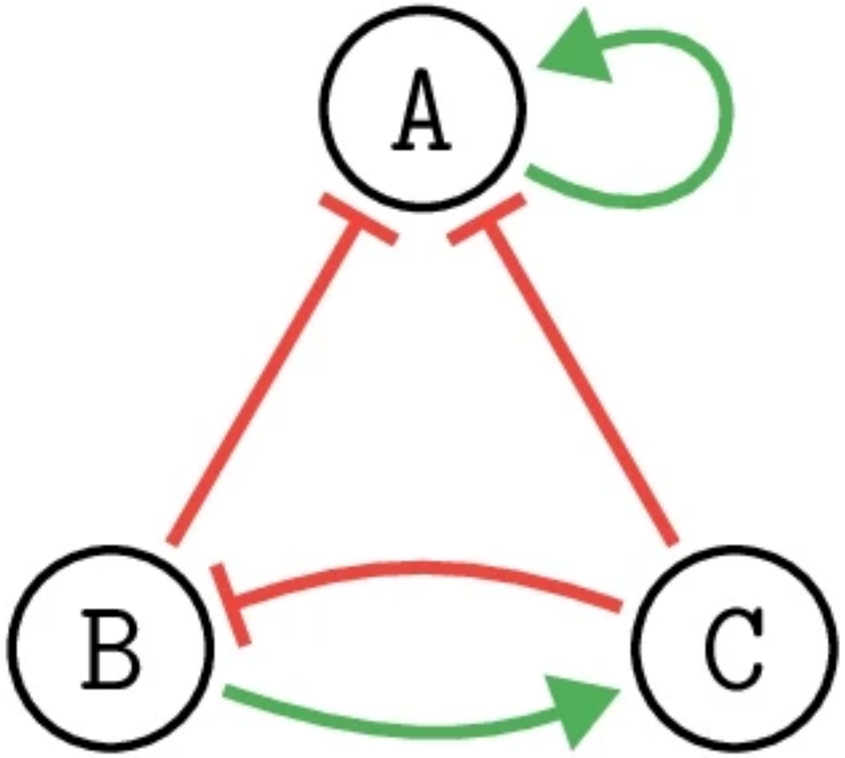
every event represents an instantaneous change of a single variable
parallelism modelled via non-determinism

update function: $F = \langle F_{\mathtt{A}}, F_{\mathtt{B}}, F_{\mathtt{C}} \rangle$

Boolean Models of RNs

Boolean Networks

$F_{\mathtt{A}} = \mathtt{A} \vee \neg\mathtt{B} \vee \neg\mathtt{C}$



$F_{\mathtt{B}} = \neg\mathtt{C}$

$F_{\mathtt{C}} = \mathtt{B}$

regulatory network + update logics

other update schemes:

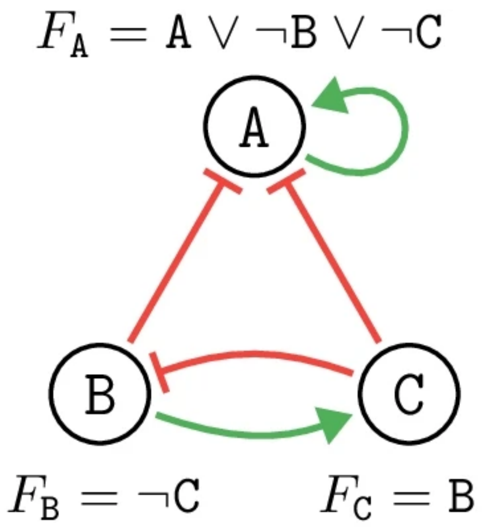
- synchronous semantics
  all vars updated simultaneously
  [Kaufman S., Nature (1969)]

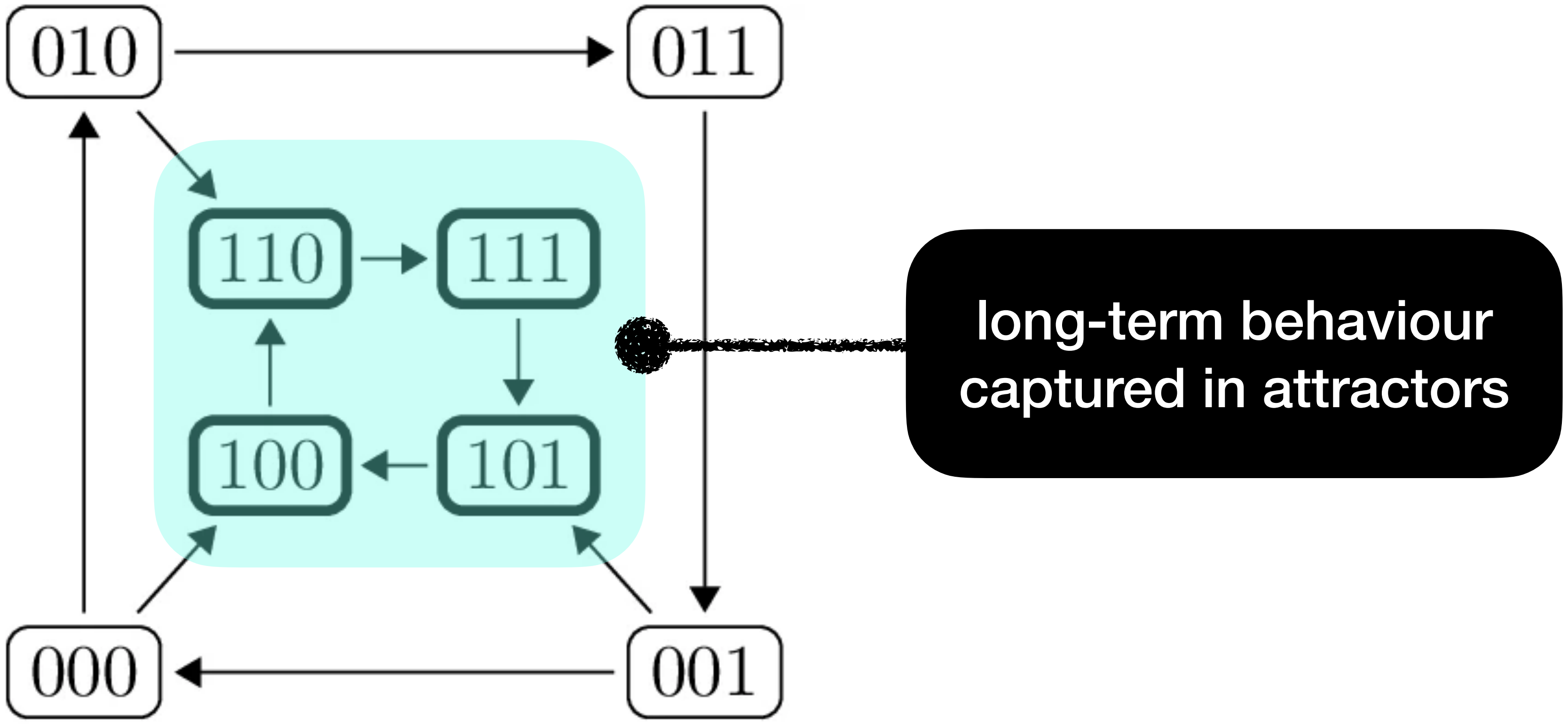
- general asynchronous semantics
  synchronous + asynchronous
  [Aracena et al., Biosystems (2009)]

- most-permissive semantics
  update is not assumed to be
  an instantaneous event in time
  [Paulevé et al., bioRxiv 2020]

Boolean Models of RNs

Attractors





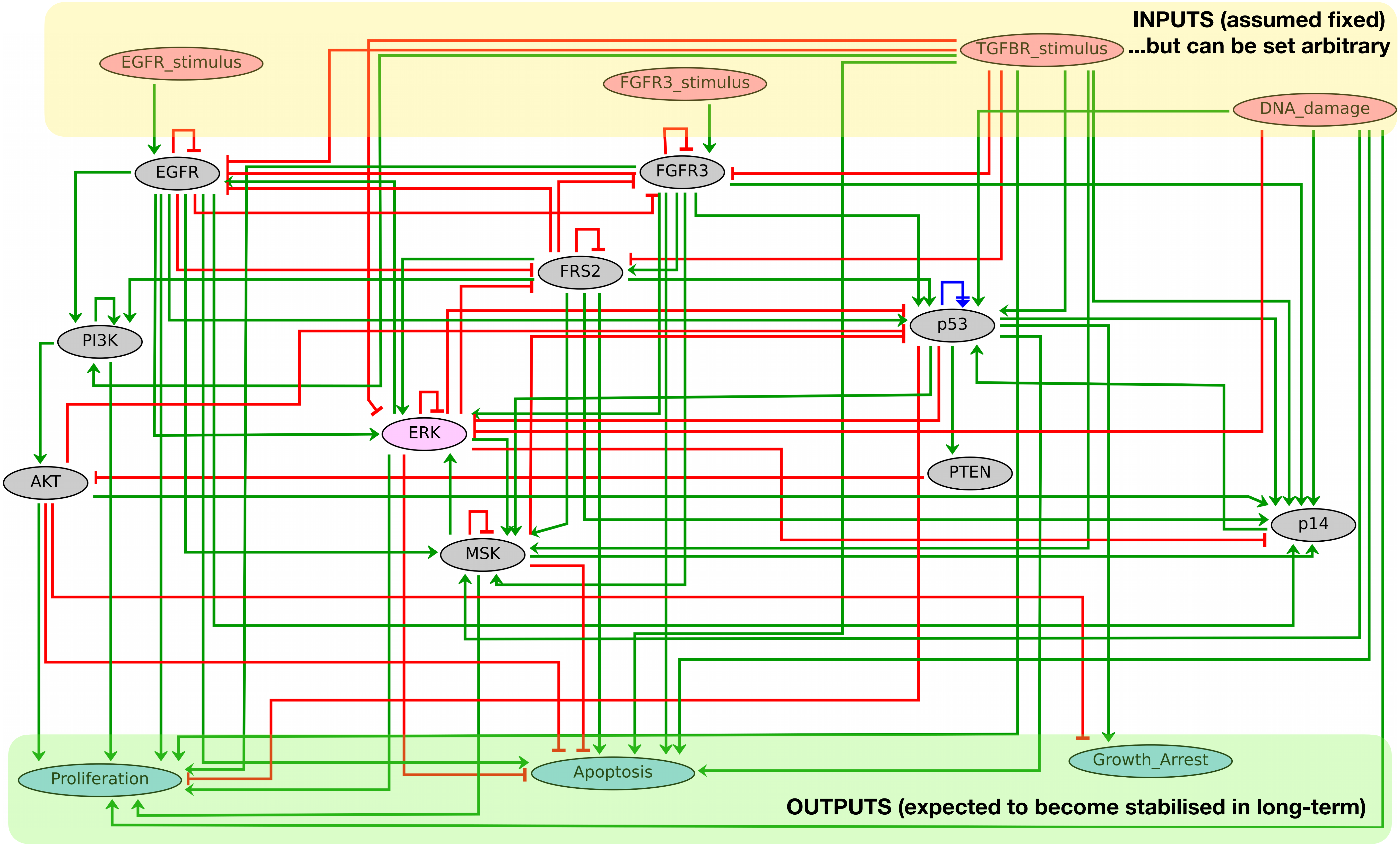
long-term behaviour captured in attractors

terminal SCCs in systems dynamics: ***steady states, oscillations, disordered behaviour***

multiple attractors can coexist (and can be alternatively reachable -- decision points)

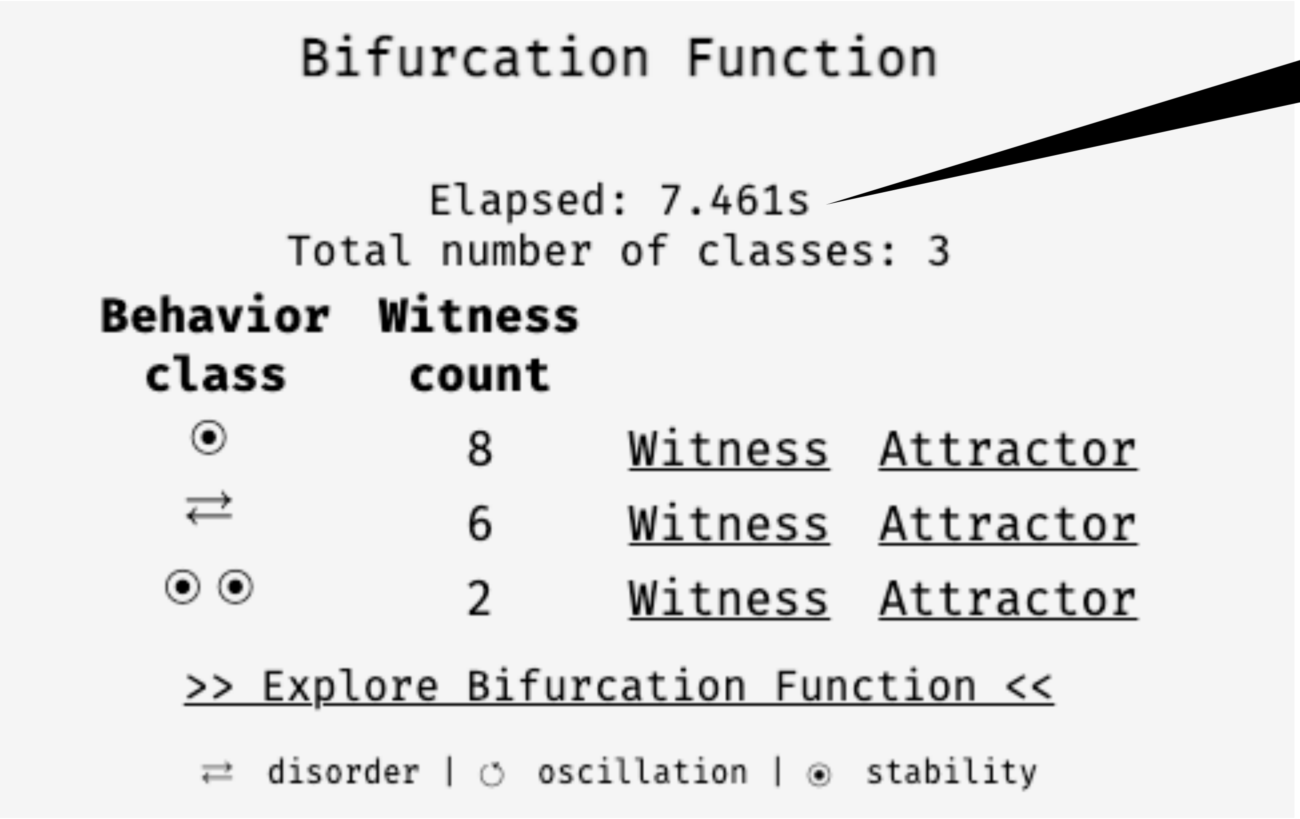
Example: MAPK signalling

Simulation in Cell Collective [Helikar et al., BMC Sys. Bio. 2012]



Example: MAPK signalling

Attractor Analysis in AEON [Beneš et al., BMC Bioinformatics (2022)]



Bifurcation Function

Elapsed: 7.461s
Total number of classes: 3

Behavior class	Witness count		
⦿	8	Witness	Attractor
⇄	6	Witness	Attractor
⦿ ⦿	2	Witness	Attractor

>> Explore Bifurcation Function <<

⇄ disorder | ⟳ oscillation | ⦿ stability

thanks to fully symbolic algorithms

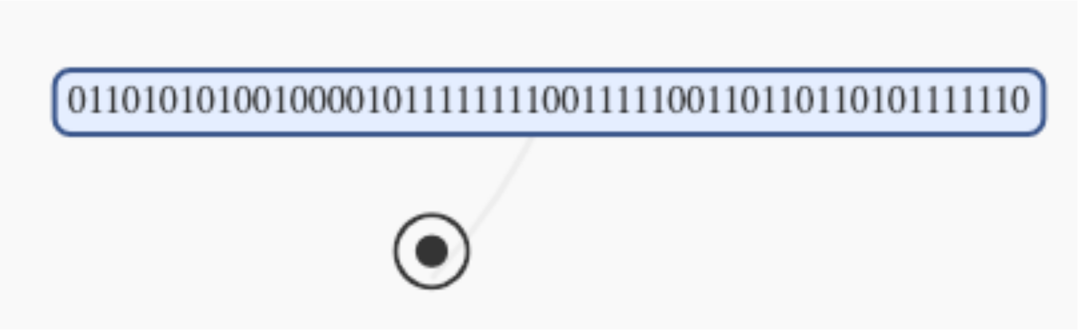
attractors (their number and shape) can change with different settings of input conditions
in the MAPK model: 4 inputs => 2^4 different situations

https://biodivine.fi.muni.cz/aeon/

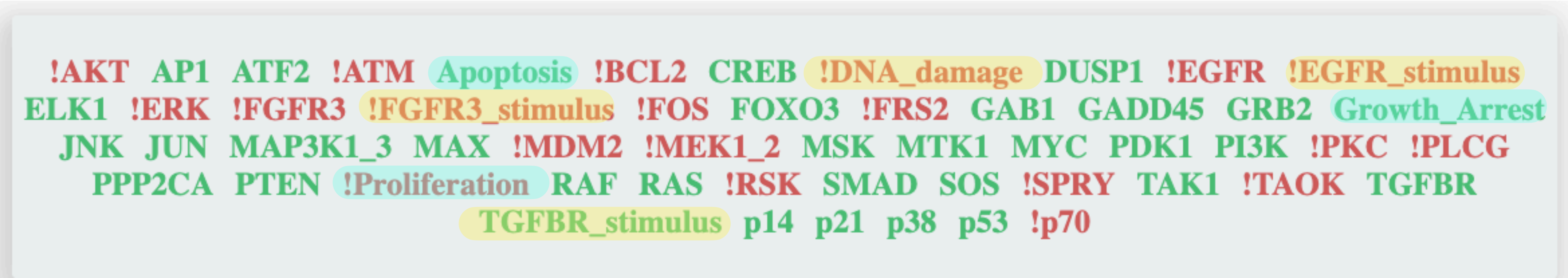
Example: MAPK signalling

Attractor Analysis in AEON [Beneš et al., BMC Bioinformatics (2022)]

an example of a single-attractor situation obtained from the model by AEON:



01101010100100001011111111001111100110110110101111110

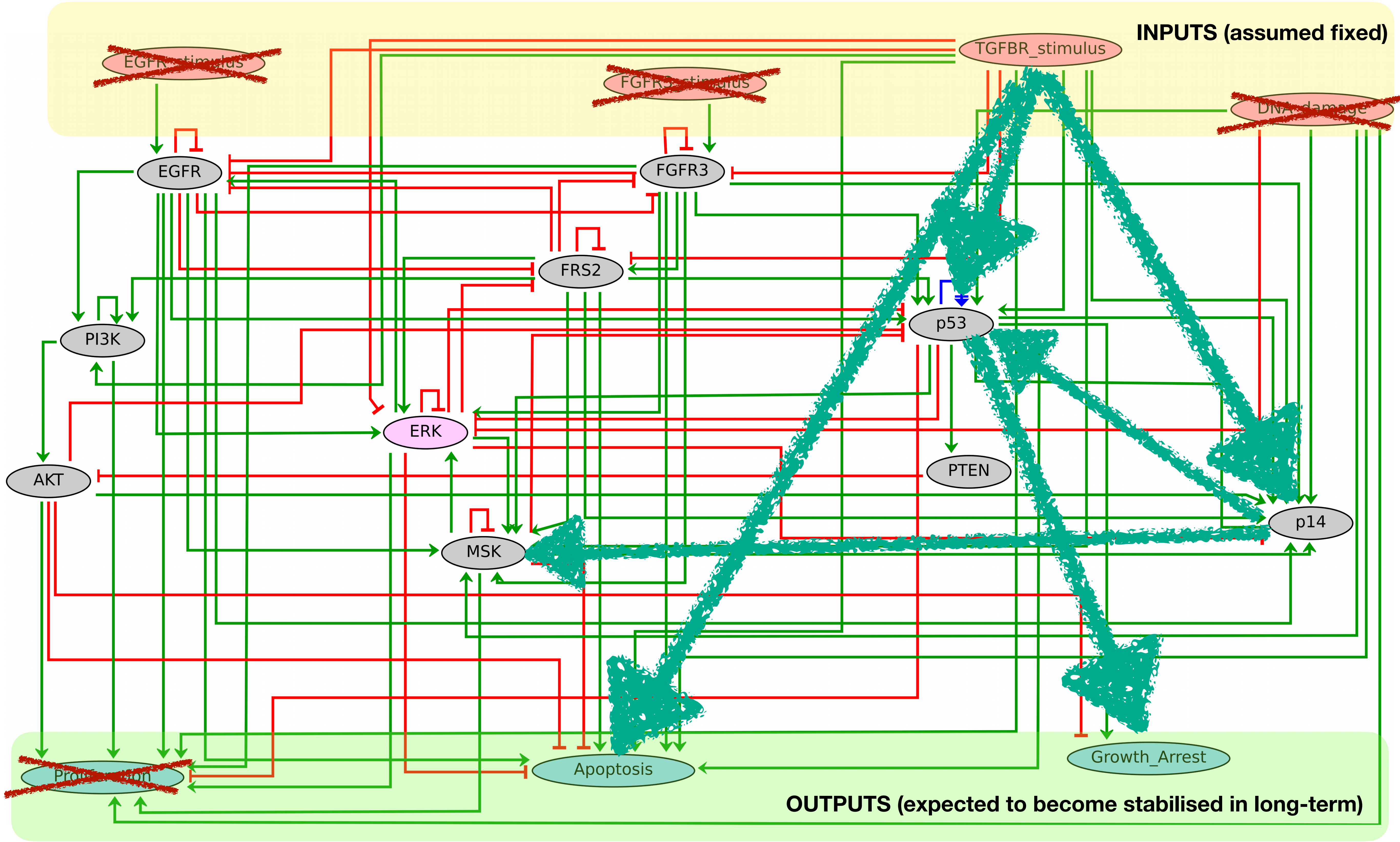


!AKT AP1 ATF2 !ATM Apoptosis !BCL2 CREB !DNA_damage DUSP1 !EGFR !EGFR_stimulus
ELK1 !ERK !FGFR3 !FGFR3_stimulus !FOS FOXO3 !FRS2 GAB1 GADD45 GRB2 Growth_Arrest
JNK JUN MAP3K1_3 MAX !MDM2 !MEK1_2 MSK MTK1 MYC PDK1 PI3K !PKC !PLCG
PPP2CA PTEN !Proliferation RAF RAS !RSK SMAD SOS !SPRY TAK1 !TAOK TGFBR
TGFBR_stimulus p14 p21 p38 p53 !p70

in case of no permanent DNA damage, no EGF/EGF stimuli, the cell decides for apoptosis
TGFBR stimulus appears to be the cause in this long-term scenario

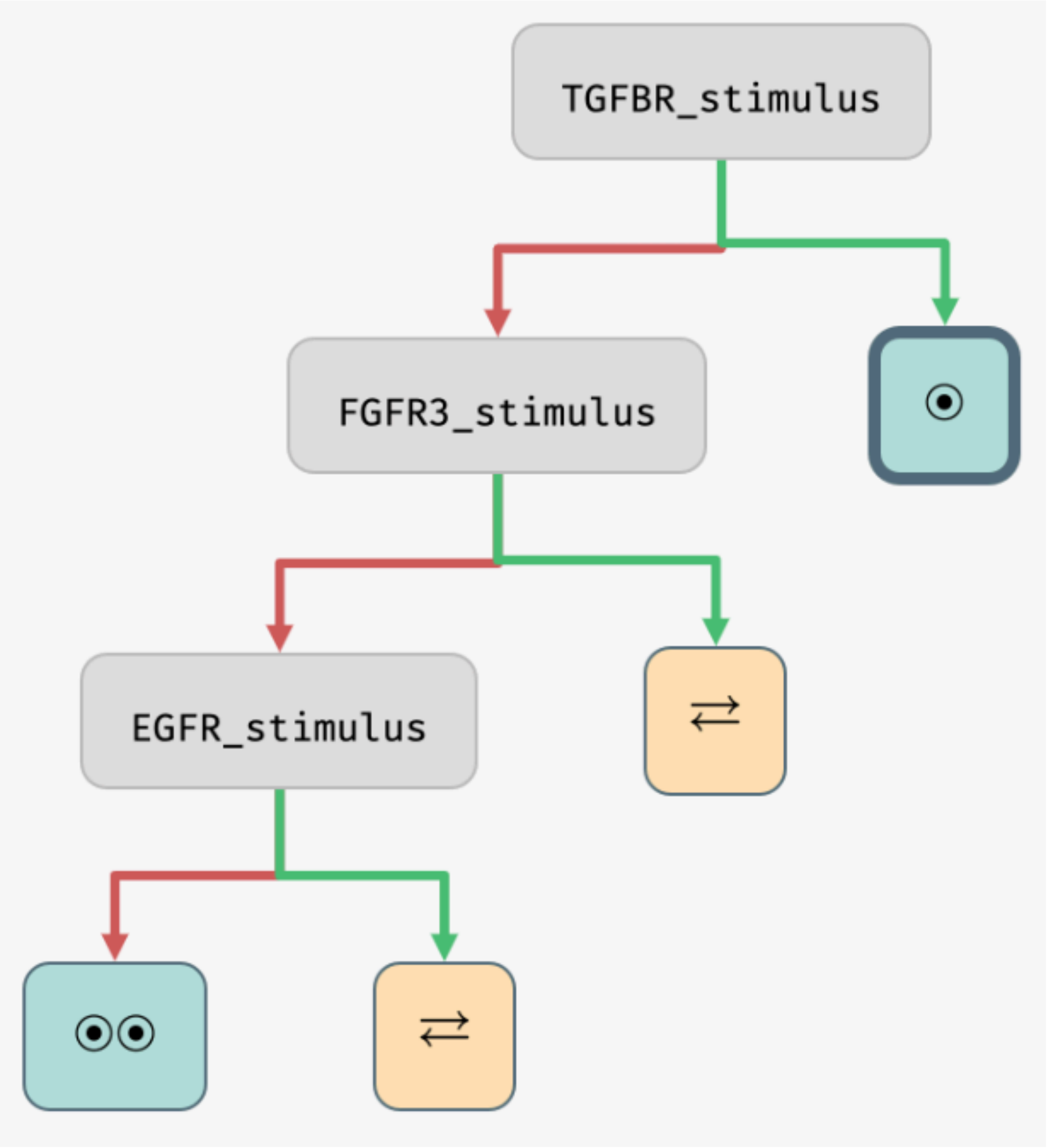
Example: MAPK signalling

Information processing in the attractor



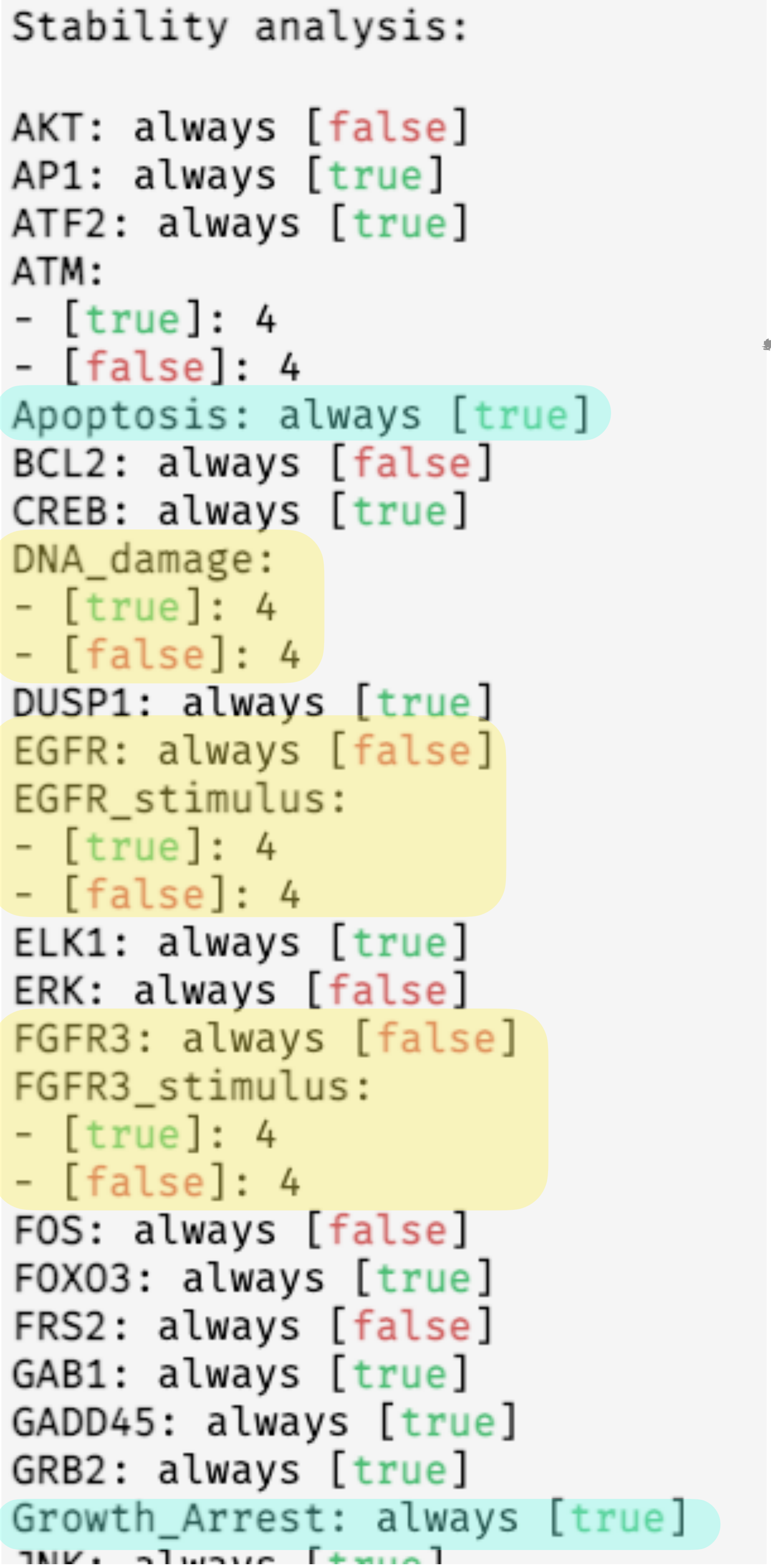
Example: MAPK signalling

Using ML to reveal how inputs affect attractors (in general)

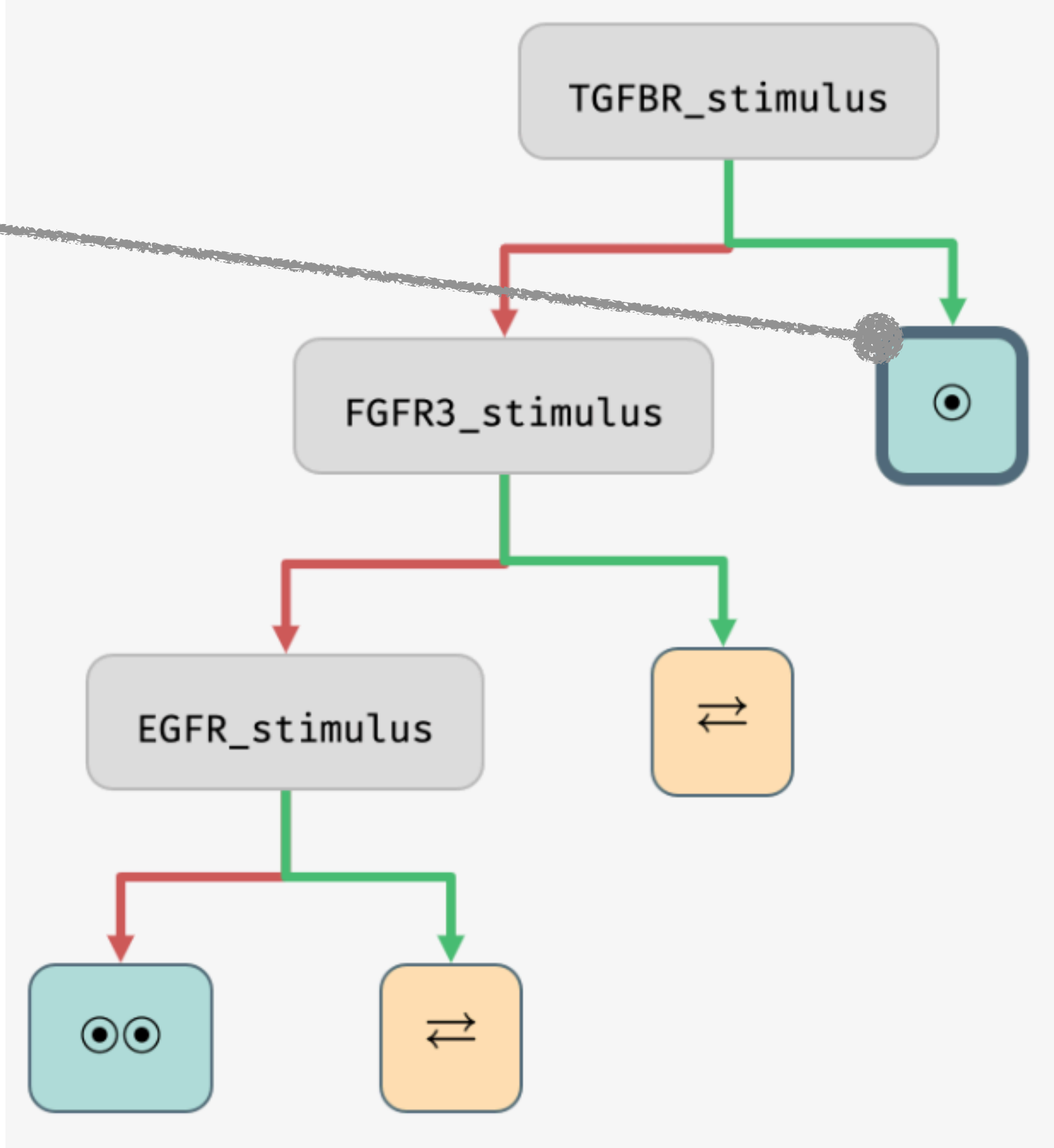


TGFBR stimulus has a direct impact to stabilise system in a single attractor inferred automatically using Decision Trees

Example: MAPK signalling



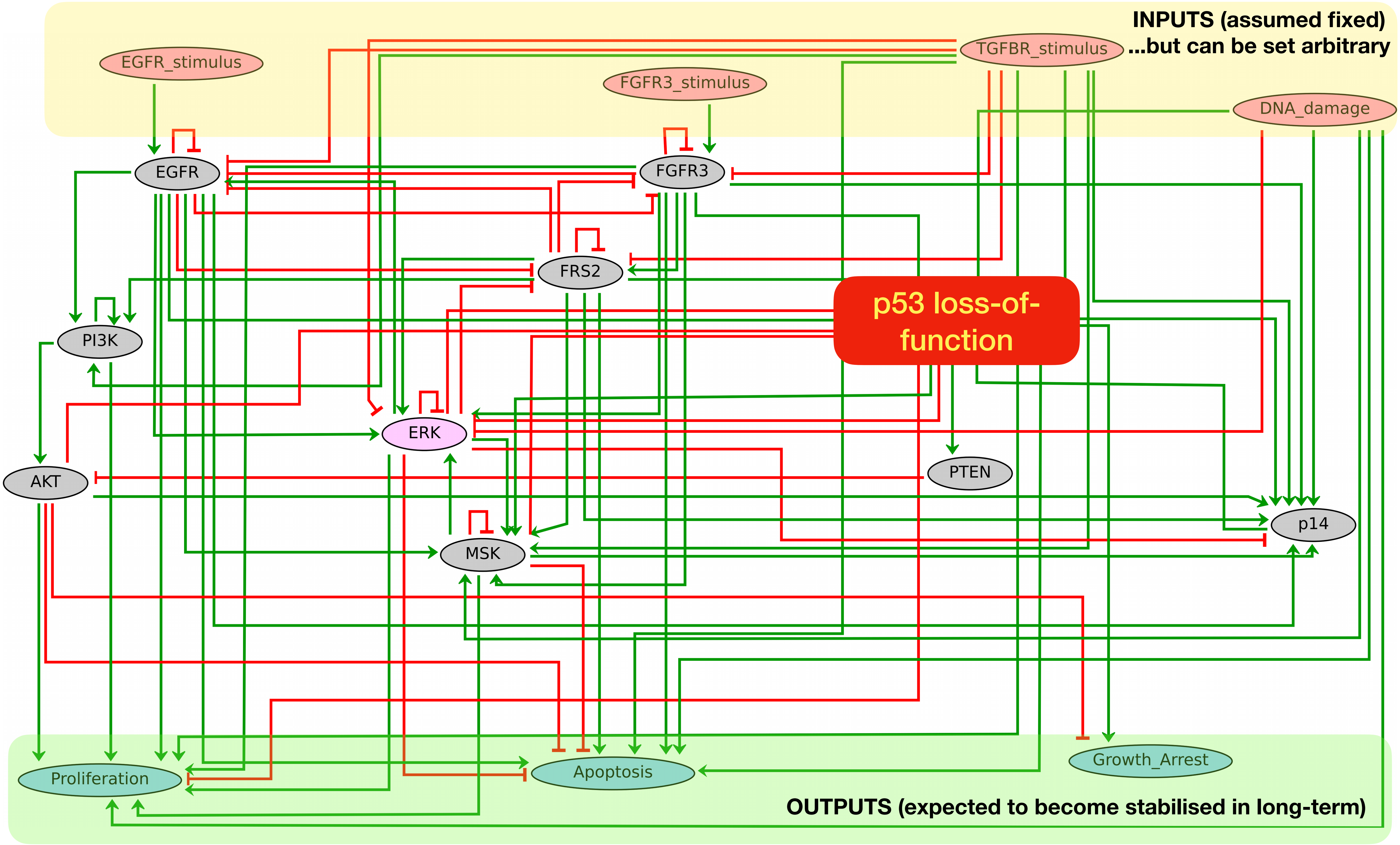
```
Stability analysis:

AKT: always [false]
AP1: always [true]
ATF2: always [true]
ATM:
- [true]: 4
- [false]: 4
Apoptosis: always [true]
BCL2: always [false]
CREB: always [true]
DNA_damage:
- [true]: 4
- [false]: 4
DUSP1: always [true]
EGFR: always [false]
EGFR_stimulus:
- [true]: 4
- [false]: 4
ELK1: always [true]
ERK: always [false]
FGFR3: always [false]
FGFR3_stimulus:
- [true]: 4
- [false]: 4
FOS: always [false]
FOXO3: always [true]
FRS2: always [false]
GAB1: always [true]
GADD45: always [true]
GRB2: always [true]
Growth_Arrest: always [true]
JNK: always [true]
```



stability analysis reveals details of the particular long-term behaviour

Example: MAPK signalling

How to incorporate perturbations (e.g., cancer deregulations)?



INPUTS (assumed fixed)
...but can be set arbitrary

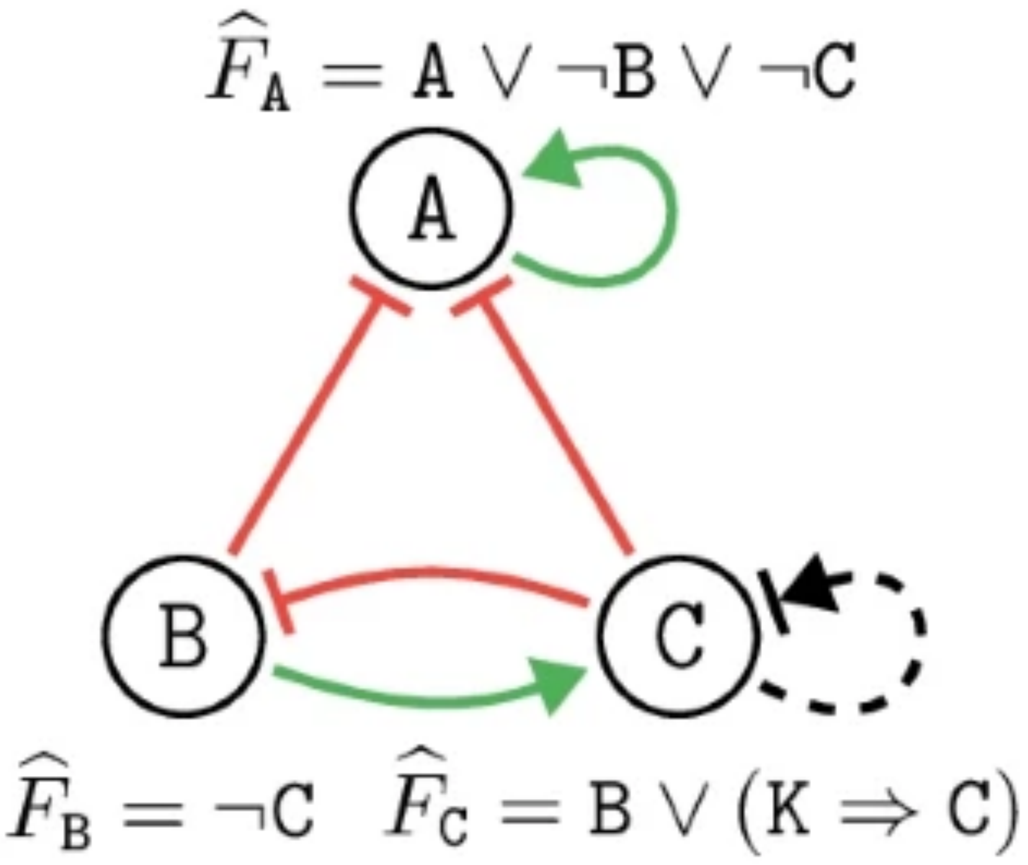
p53 loss-of-function

OUTPUTS (expected to become stabilised in long-term)

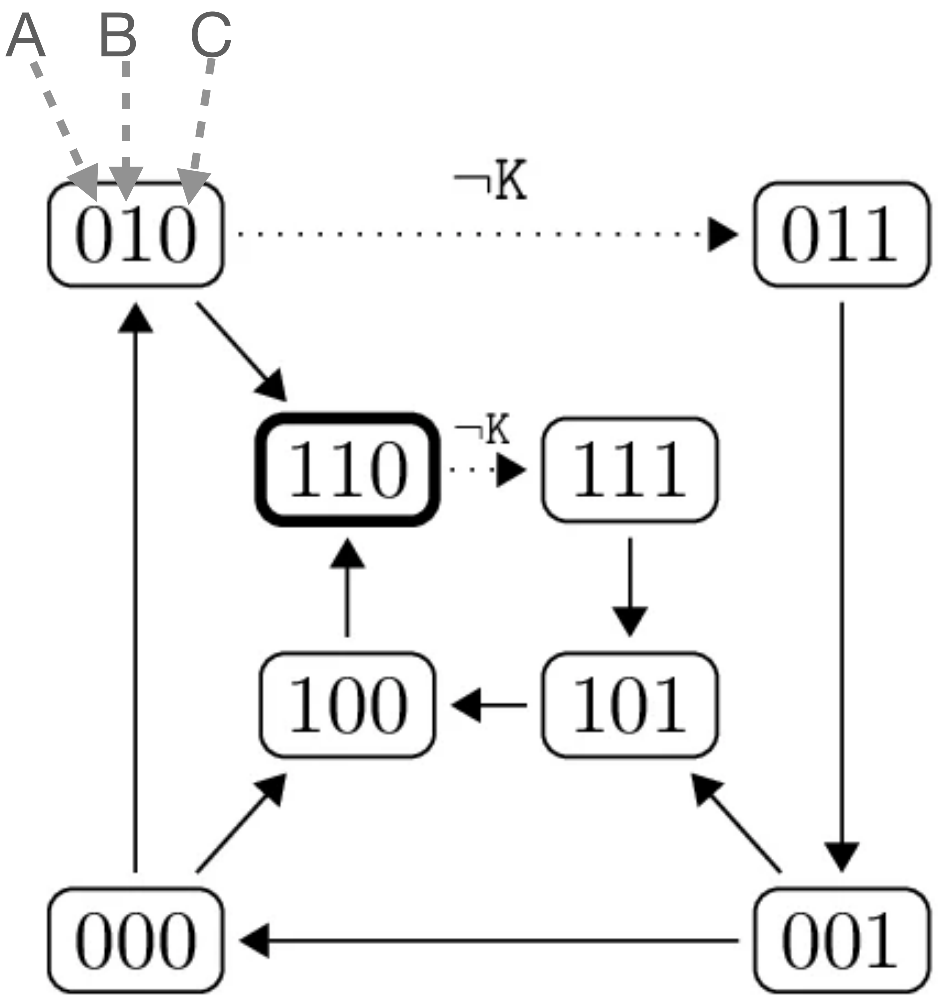
Boolean Models of RNs

Partially-Specified Boolean Networks

[Beneš et al., ICFEM 2019]



$\widehat{F}_A = A \lor \neg B \lor \neg C$
A
B
C
$\widehat{F}_B = \neg C$
$\widehat{F}_C = B \lor (K \Rightarrow C)$

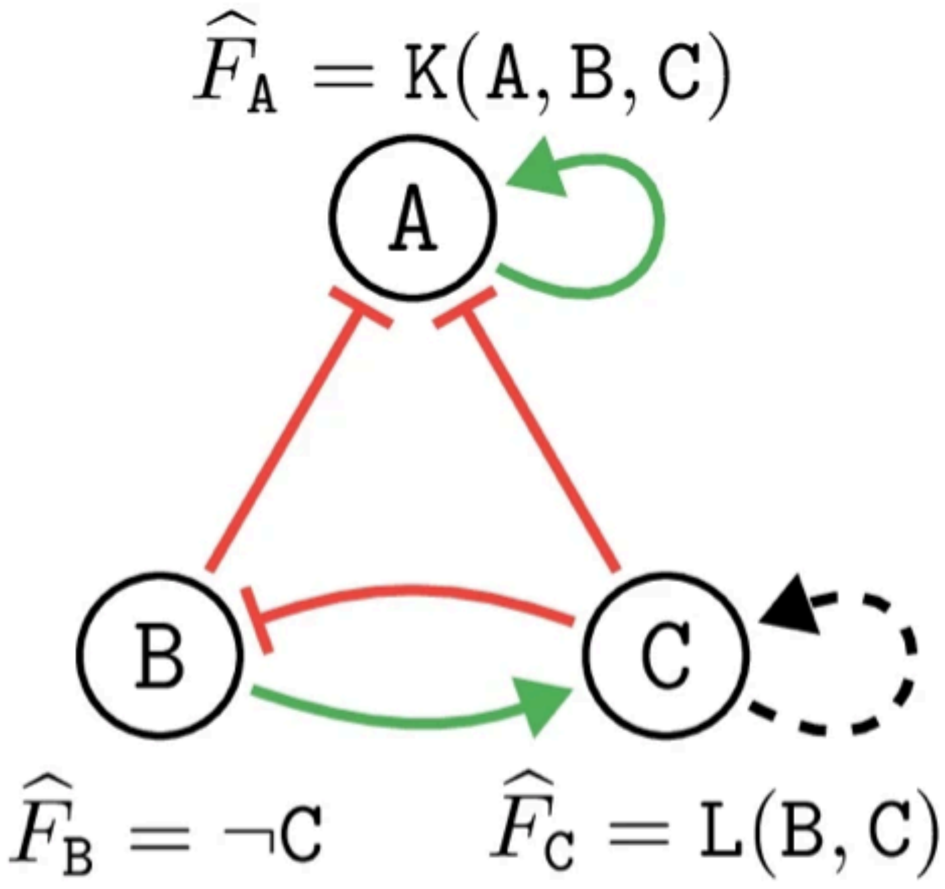


update logics with unknown information K => system dynamics with "coloured" events

Boolean Models of RNs

Partially-Specified Boolean Networks (psBNs)

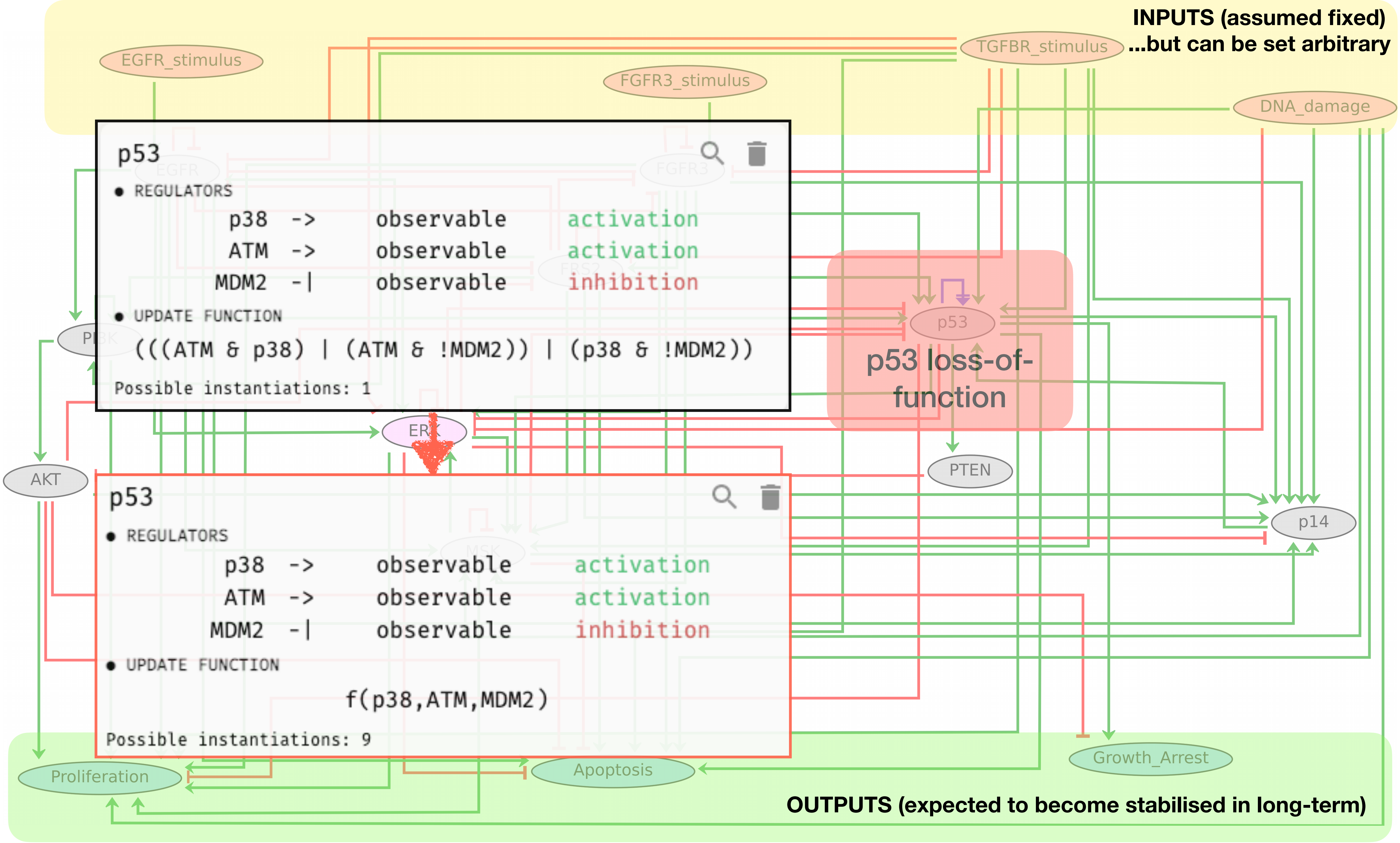
[Beneš et al., CAV 2020]


$\widehat{F}_{\mathtt{A}} = \mathtt{K}(\mathtt{A},\mathtt{B},\mathtt{C})$
A
B
C
$\widehat{F}_{\mathtt{B}} = \neg \mathtt{C}$
$\widehat{F}_{\mathtt{C}} = \mathtt{L}(\mathtt{B},\mathtt{C})$

- implemented in AEON
  uninterpreted functions
- fully symbolic psBN representation
  utilising Binary Decision Diagrams
- partially-specified information:
  fixed inputs
  logic operators in update functions
  incl. arity (essentiality) and regulation types

Example: MAPK signalling

How to incorporate perturbations (e.g., cancer deregulations)?



INPUTS (assumed fixed)
...but can be set arbitrary

```
p53
● REGULATORS
        p38  ->    observable    activation
        ATM  ->    observable    activation
       MDM2  -|    observable    inhibition
● UPDATE FUNCTION
 (((ATM & p38) | (ATM & !MDM2)) | (p38 & !MDM2))
Possible instantiations: 1
```

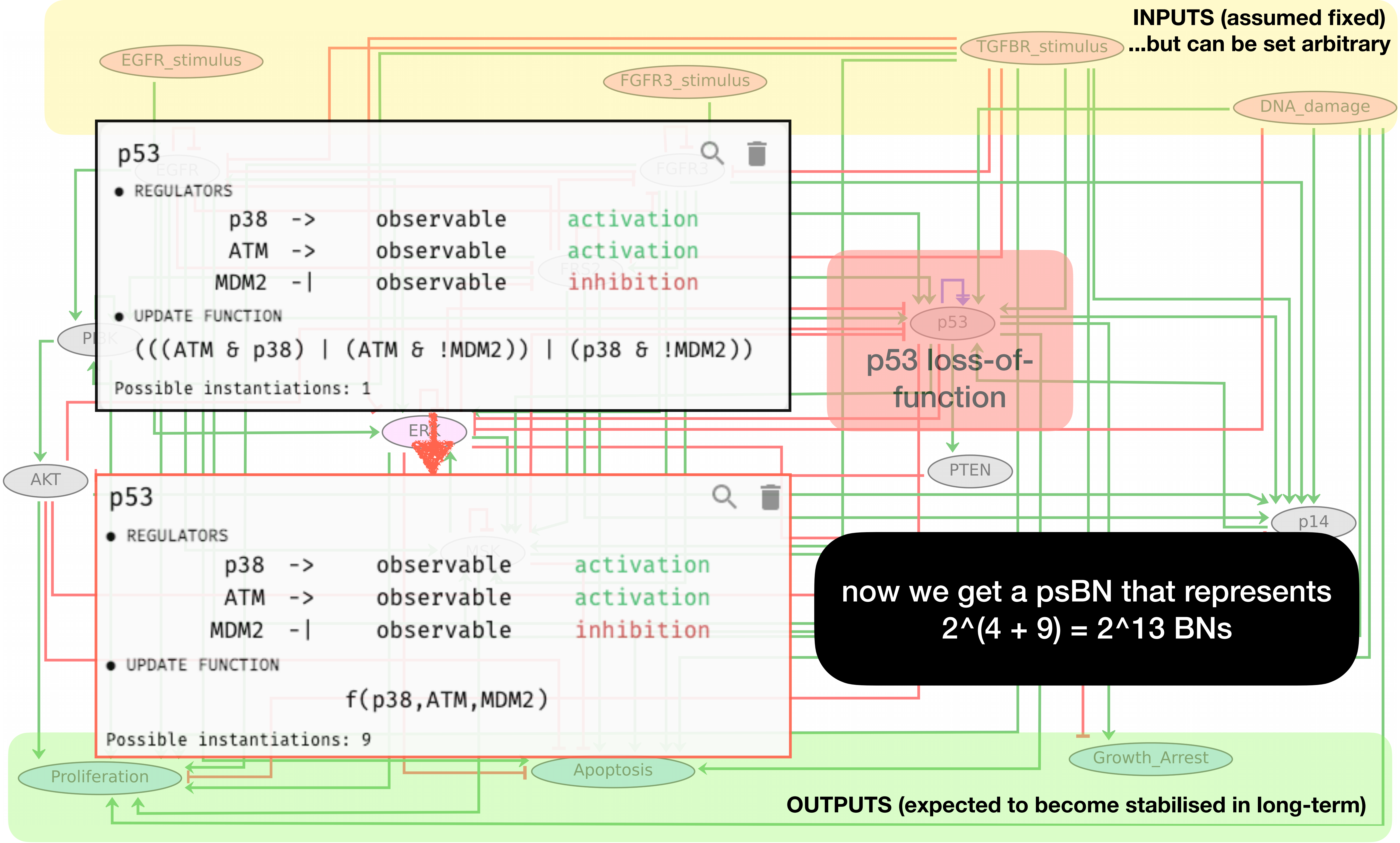
p53 loss-of-function

```
p53
● REGULATORS
        p38  ->    observable    activation
        ATM  ->    observable    activation
       MDM2  -|    observable    inhibition
● UPDATE FUNCTION
                 f(p38,ATM,MDM2)
Possible instantiations: 9
```

OUTPUTS (expected to become stabilised in long-term)

Example: MAPK signalling

How to incorporate perturbations (e.g., cancer deregulations)?



INPUTS (assumed fixed)
...but can be set arbitrary

```
p53
• REGULATORS
        p38   ->     observable     activation
        ATM   ->     observable     activation
       MDM2   -|     observable     inhibition
• UPDATE FUNCTION
 (((ATM & p38) | (ATM & !MDM2)) | (p38 & !MDM2))
Possible instantiations: 1
```

p53 loss-of-function

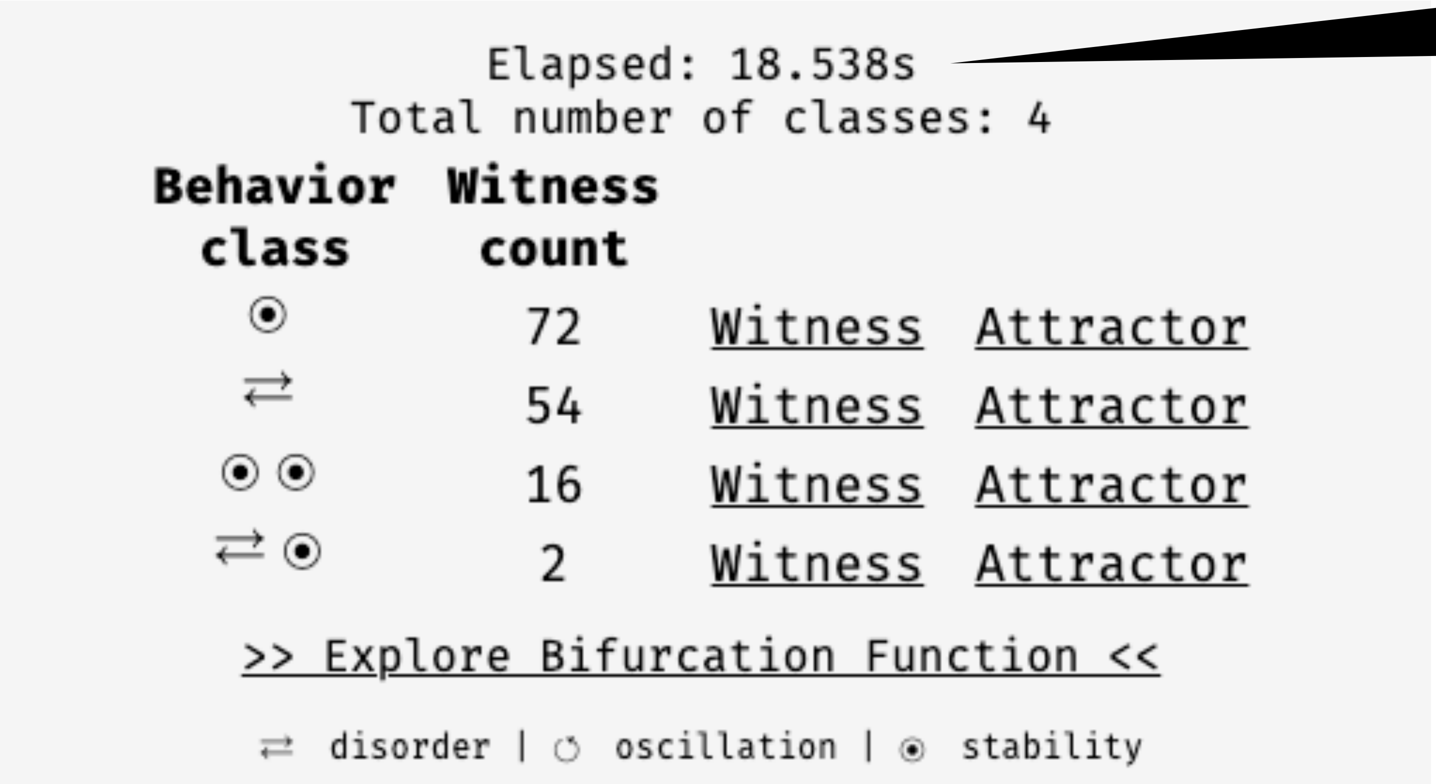
```
p53
• REGULATORS
        p38   ->     observable     activation
        ATM   ->     observable     activation
       MDM2   -|     observable     inhibition
• UPDATE FUNCTION
                 f(p38,ATM,MDM2)
Possible instantiations: 9
```

now we get a psBN that represents 2^(4 + 9) = 2^13 BNs

OUTPUTS (expected to become stabilised in long-term)

Example: MAPK signalling

Attractor Analysis of psBNs in AEON



Elapsed: 18.538s
Total number of classes: 4

Behavior class	Witness count		
⦿	72	Witness	Attractor
⇄	54	Witness	Attractor
⦿ ⦿	16	Witness	Attractor
⇄ ⦿	2	Witness	Attractor

>> Explore Bifurcation Function <<

⇄ disorder | ↻ oscillation | ⦿ stability

still fine due to symbolic algorithms

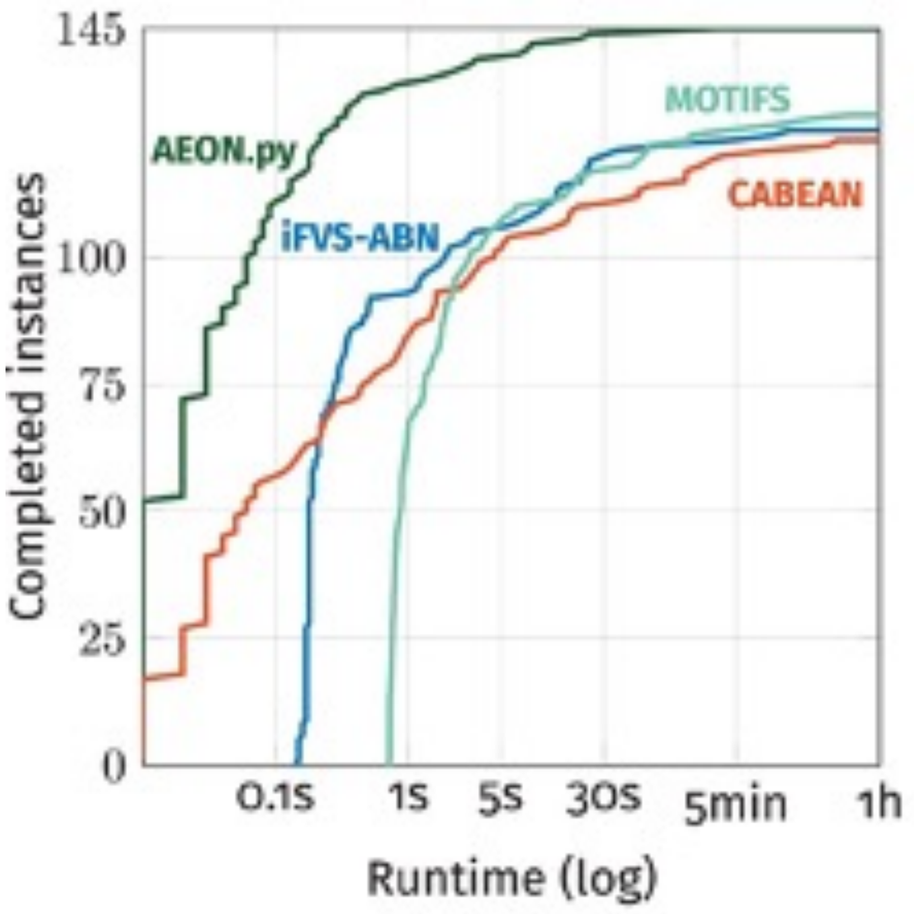
by assuming the quite general cause of p53 malfunction we have obtained a new class of attractors

we can even enhance the perturbations by affecting the update functions of vars regulated by p53

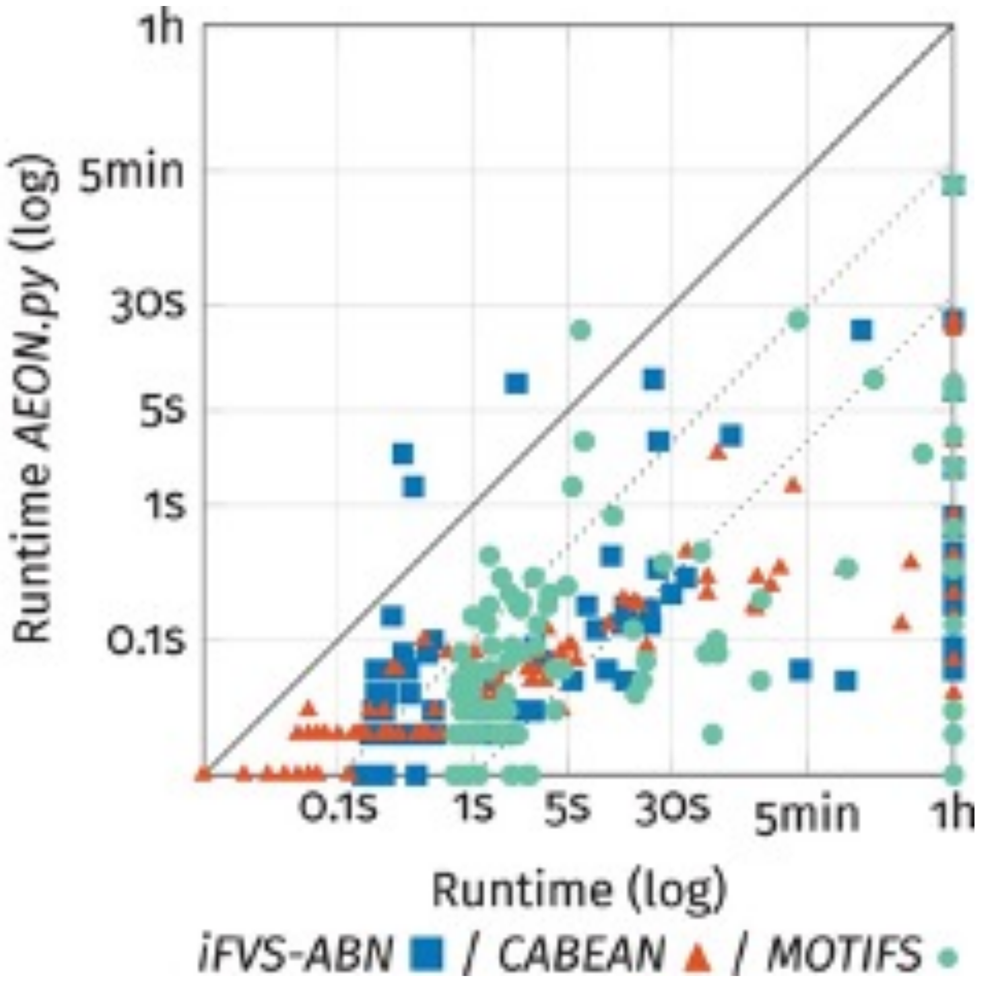
https://biodivine.fi.muni.cz/aeon/

Performance

Some advances on attractor analysis in AEON.py



evaluated on 145 real-word BNs



we have developed interleaved transition guided reduction (ITGR) [Beneš et al. CAV 2021]

this is based on pruning the non-attractor states during computation of attractors

check our most recent paper in Bioinformatics

Some Links

For those interested

- First algorithm for attractors in psBNs [ICFEM 2019]
- AEON first release [CAV 2020]
- AEON 2021 (with decision trees) [CMSB 2021]
- Symbolic SCC decomposition of coloured graphs [TACAS 2021]
- Transition guided reduction [CAV 2021]
- Control (reprogramming) of psBNs [Mathematics 2021]
- AEON in examples [BMC Bioinformatics 2022]
- AEON.py (API, optimisation, control) [Bioinformatics 2022]

Work in Progress

How to obtain the right BN model?

- Transform reaction network to BN (the case of MAPK example)
- Inference methods from (steady-state) expression data
  - Optimisation via ML-based methods (genetic programming)
  - Works with synchronous update scheme (simulation)
  - Tuned for synthetic data (DREAM)
- Reality: lack of data, data are noisy, there is some prior knowledge in literature, databases, …
- *Our approach:* compute all attractor-matching candidates with AEON and employ model checking to further prune w.r.t. prior knowledge

Thank you for your attention!