‘Apps’ Biologicas

Natalio Krasnogor
Newcastle University

Twitter: @NKrasnogor
http://homepages.cs.ncl.ac.uk/natalio.krasnogor/

I(C)$_2$S

Centre For Bacterial Cell Biology

Newcastle University

Centre for
Synthetic Biology
and the Bioeconomy

Contenido

- Introduccion a la BioComputacion

•’Apps’ biologicas: como programar organismos vivos (modelos, herramientas, ejemplos)

•Infraestructuras informaticas para el modelado 3D de colonias bacterianas

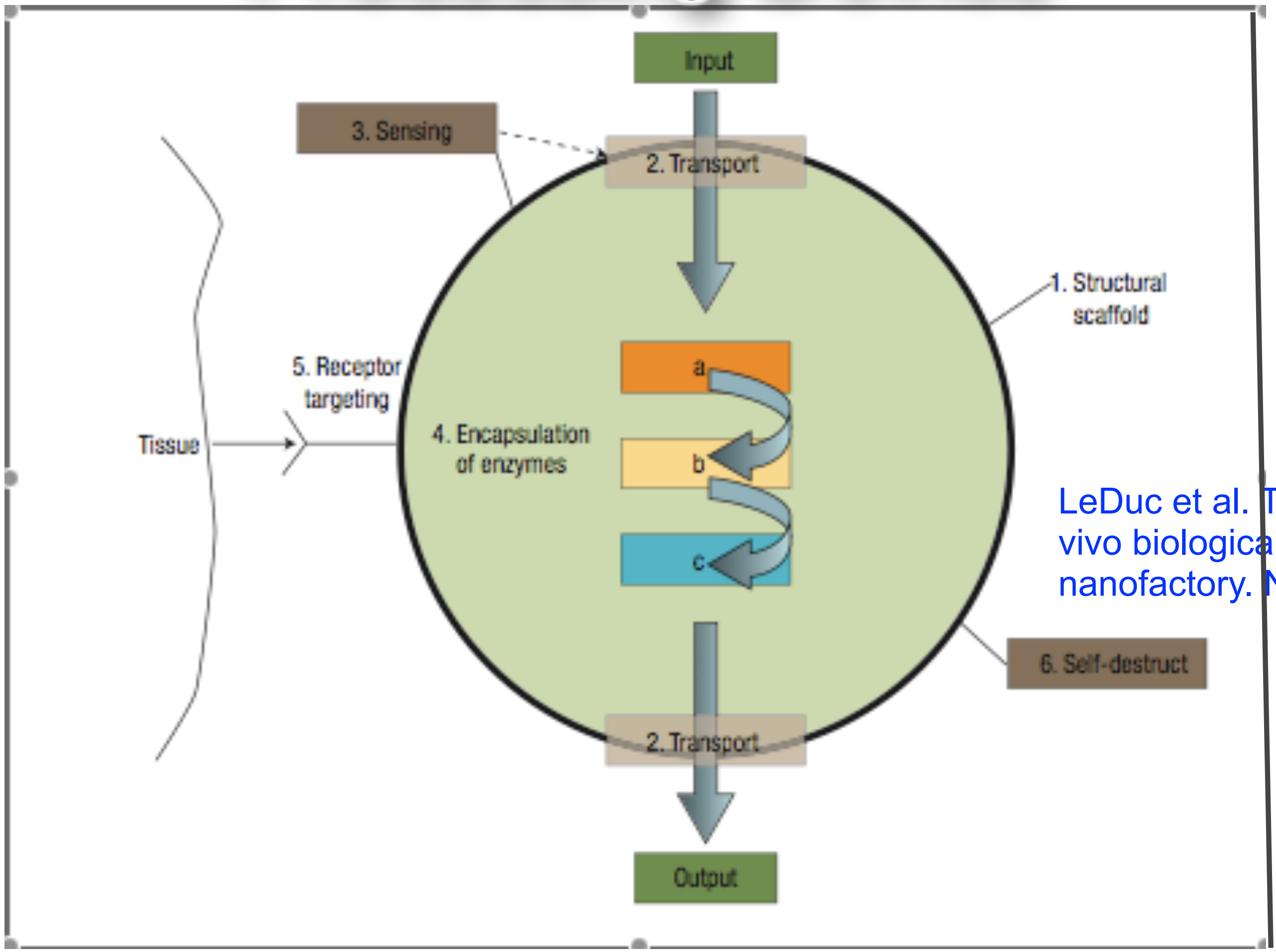
Outline

- The cell as a Computing Device for Biocomputing
- P Systems for Representing Cellular Computation
- Examples of Biocomputation
- Conclusions

Outline

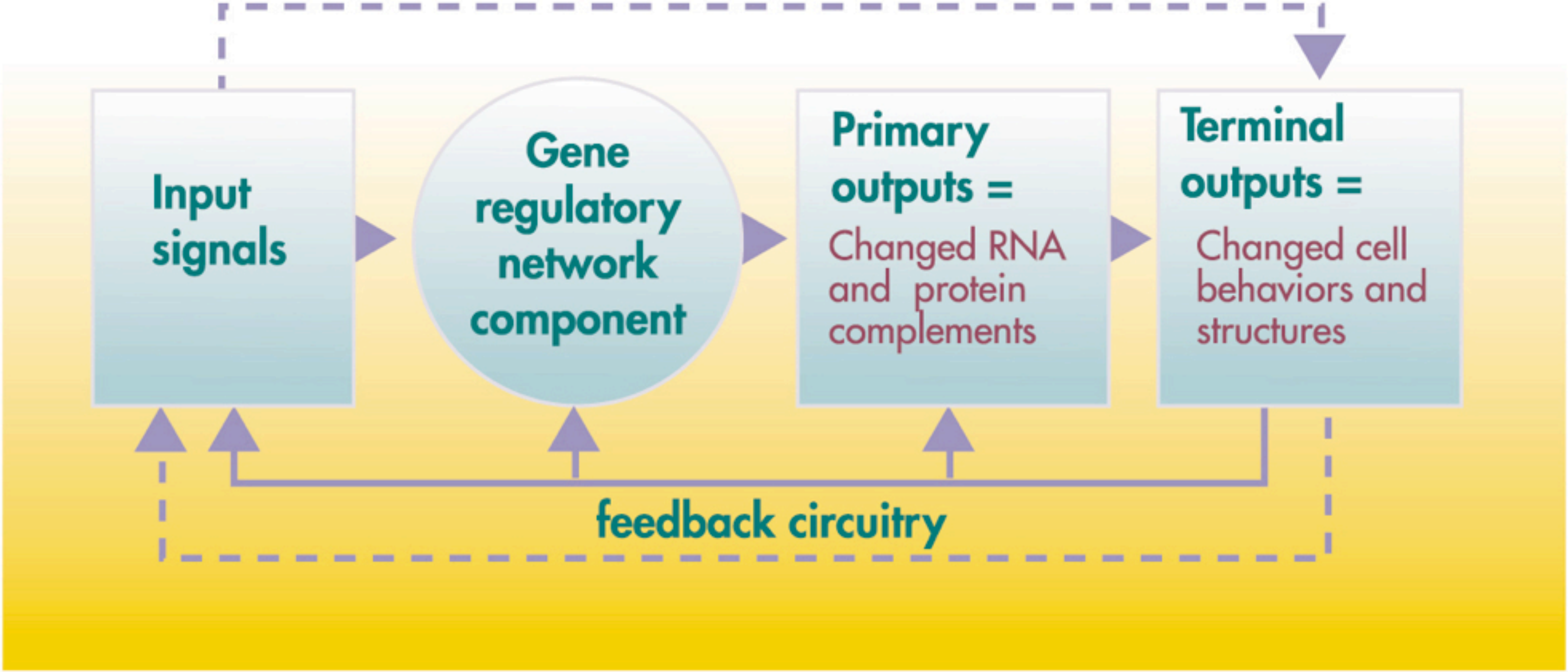
- The cell as a Computing Device for Biocomputing
- P Systems for Representing Cellular Computation
- Examples of Biocomputation
- Conclusions

The Cell as an Information Processing Device



LeDuc et al. Towards an in vivo biologically inspired nanofactory. Nature (2007)

The Cell as an Intelligent (Evolved) Machine



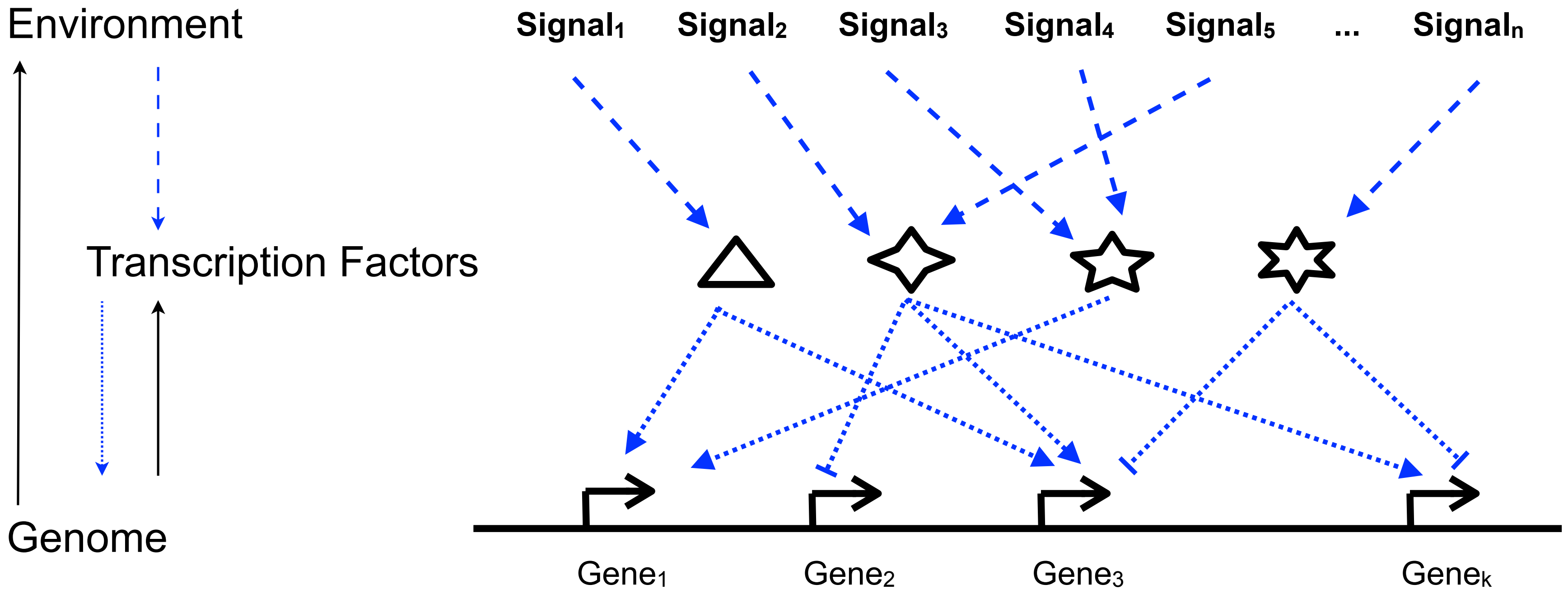
YGG 01-0086 Wikimedia Commons

Amir Mitchell, et al., Adaptive prediction of environmental changes by microorganisms. Nature June 2009.

Centre For Bacterial Cell Biology

University
Centre for Synthetic Biology and the Bioeconomy

Transcription Networks



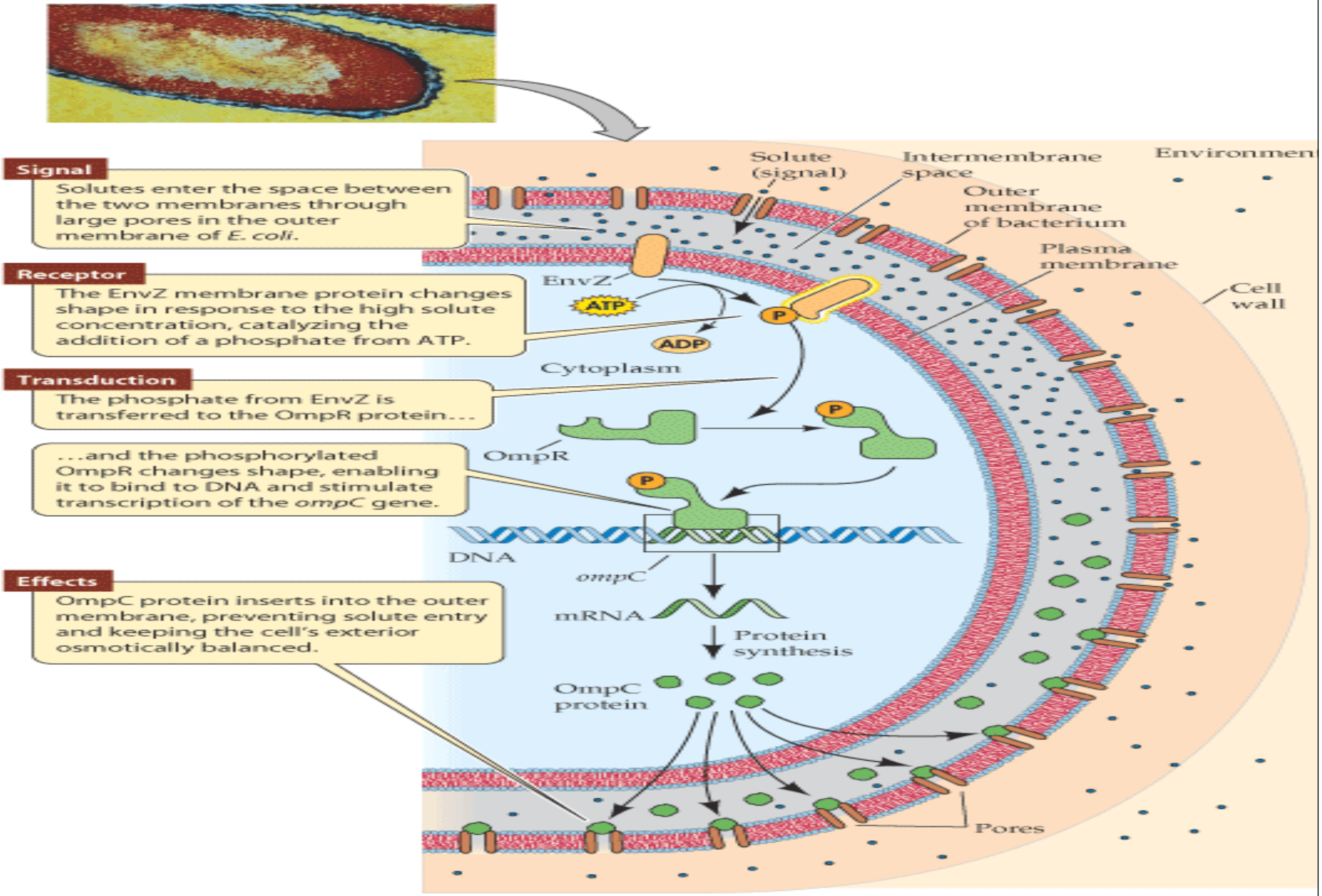
Environment

Transcription Factors

Genome

Signal$_1$ Signal$_2$ Signal$_3$ Signal$_4$ Signal$_5$ ... Signal$_n$

Gene$_1$ Gene$_2$ Gene$_3$ Gene$_k$



Signal

Solutes enter the space between the two membranes through large pores in the outer membrane of *E. coli*.

Receptor

The EnvZ membrane protein changes shape in response to the high solute concentration, catalyzing the addition of a phosphate from ATP.

Transduction

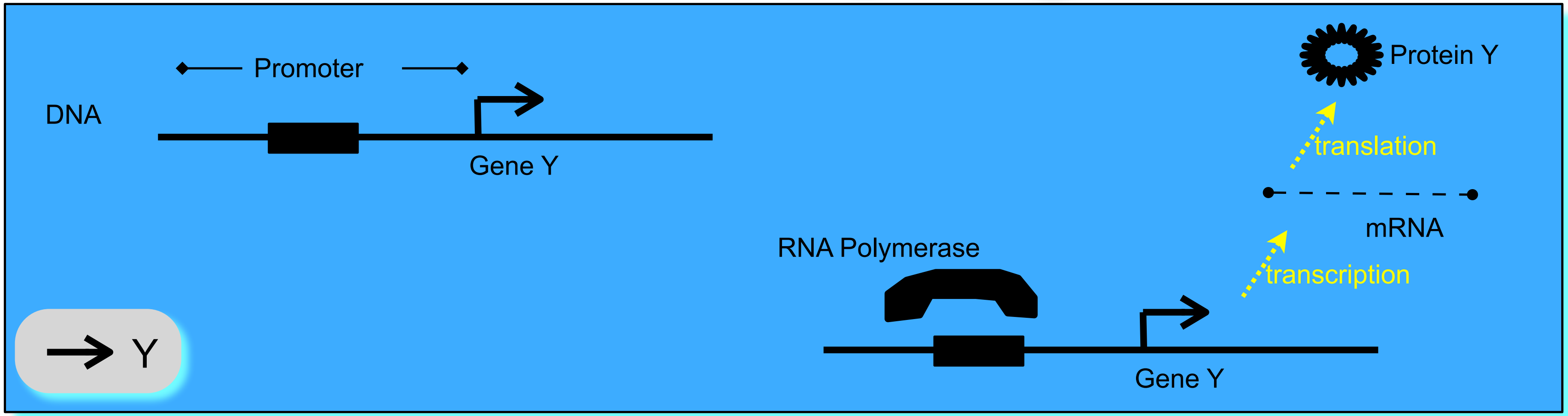
The phosphate from EnvZ is transferred to the OmpR protein…

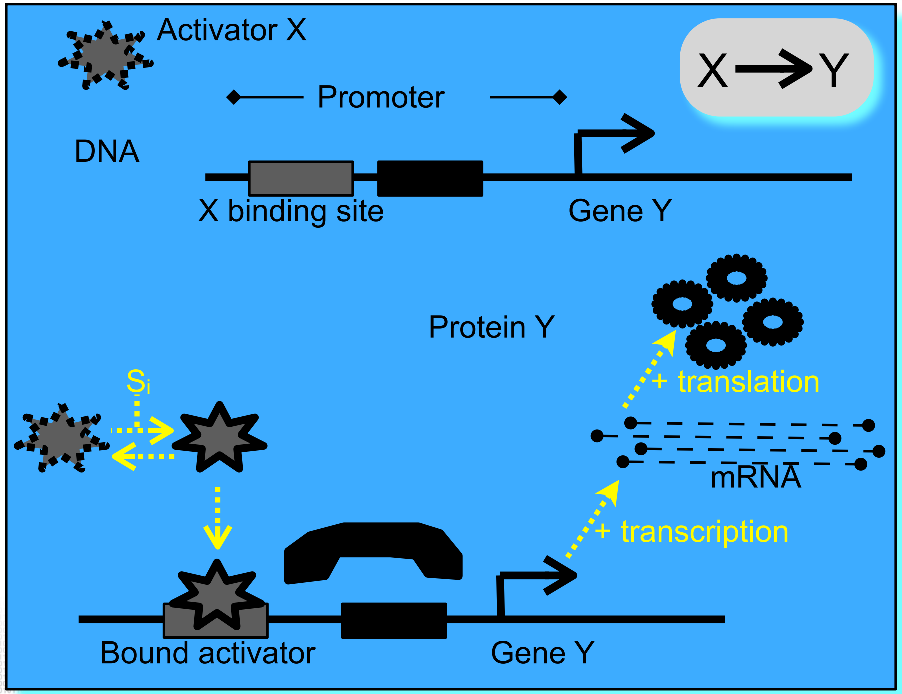
…and the phosphorylated OmpR changes shape, enabling it to bind to DNA and stimulate transcription of the *ompC* gene.

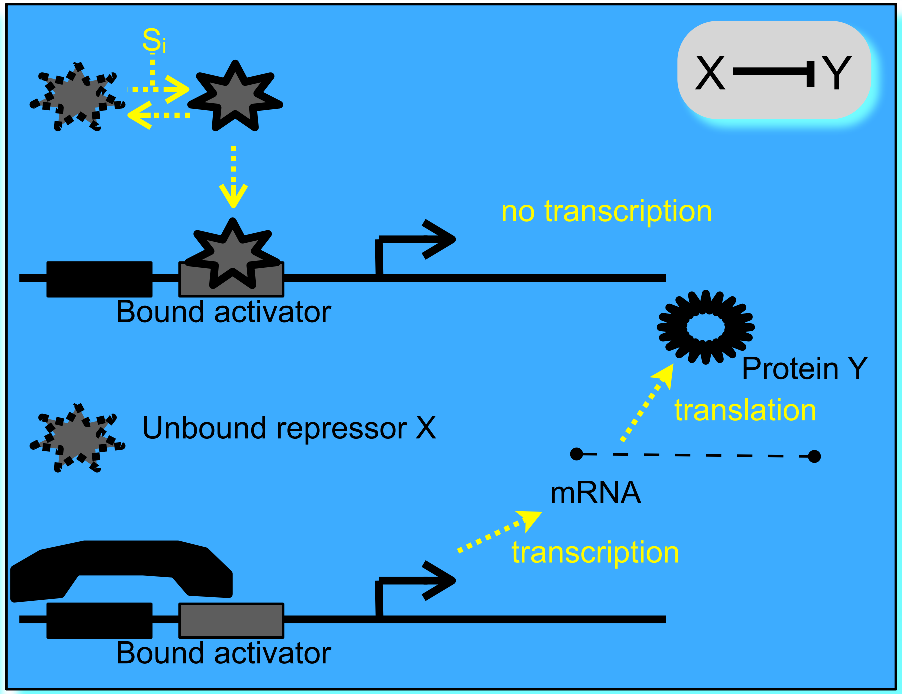
Effects

OmpC protein inserts into the outer membrane, preventing solute entry and keeping the cell's exterior osmotically balanced.

The Basic Unit: A Gene’s Transcription Regulation Mechanics







Network Motifs: Evolution’s Preferred Circuits

- Biological networks are complex and vast
- To understand their functionality in a *scalable* way one must choose the correct abstraction

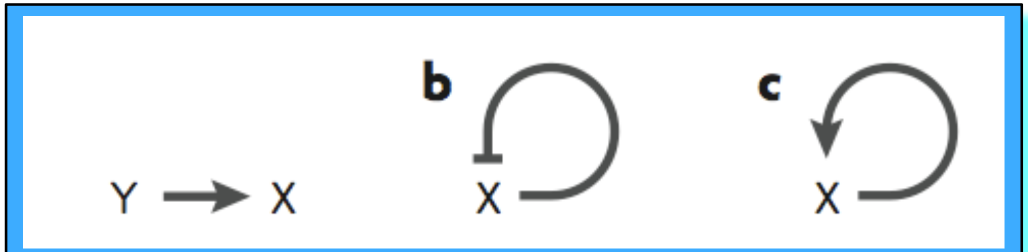
> ***“Patterns that occur in the real network significantly more often than in randomized networks are called network motifs”***

Shai S. Shen-Orr et al., Network motifs in the transcriptional regulation network of Escherichia coli. Nature Genetics 31, 64 - 68 (2002)

- Moreover, these patterns are organised in non-trivial/non-random hierarchies

Radu Dobrin et al., Aggregation of topological motifs in the Escherichia coli transcriptional regulatory network. BMC Bioinformatics. 2004; 5: 10.

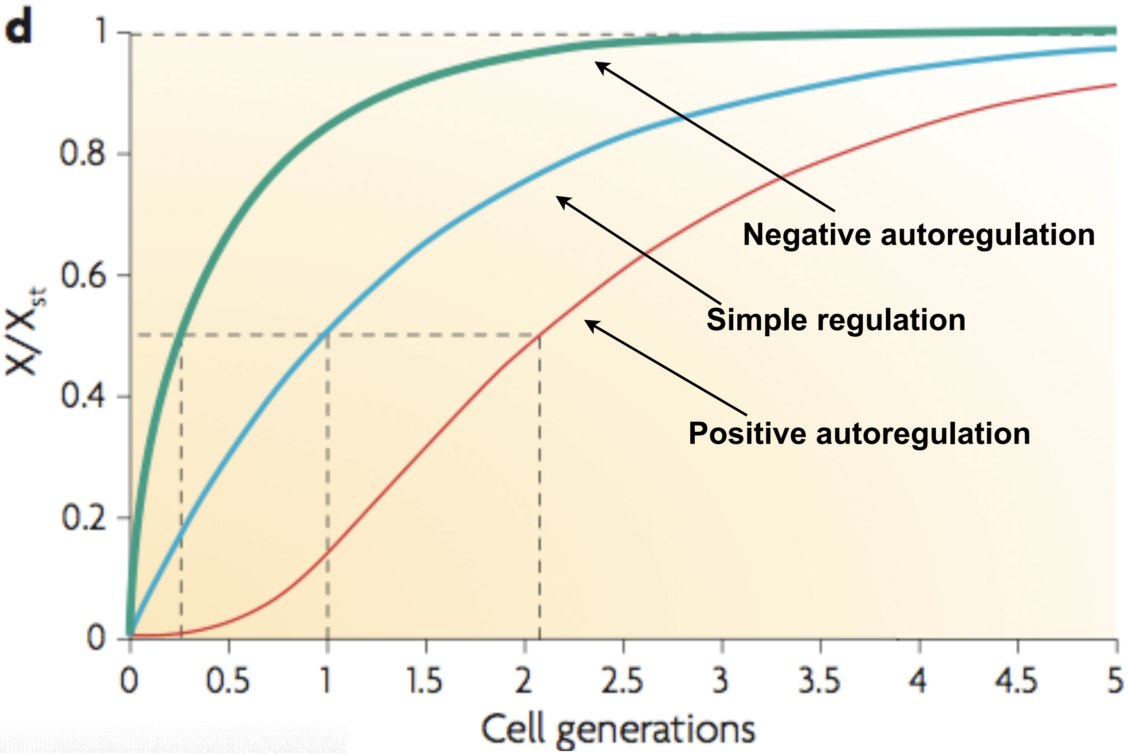
- ***Each network motif carries out a specific information-processing function***



Y positively regulates X

Negative autoregulation

Positive autoregulation



Negative autoregulation

Simple regulation

Positive autoregulation

$\rightarrow$ X

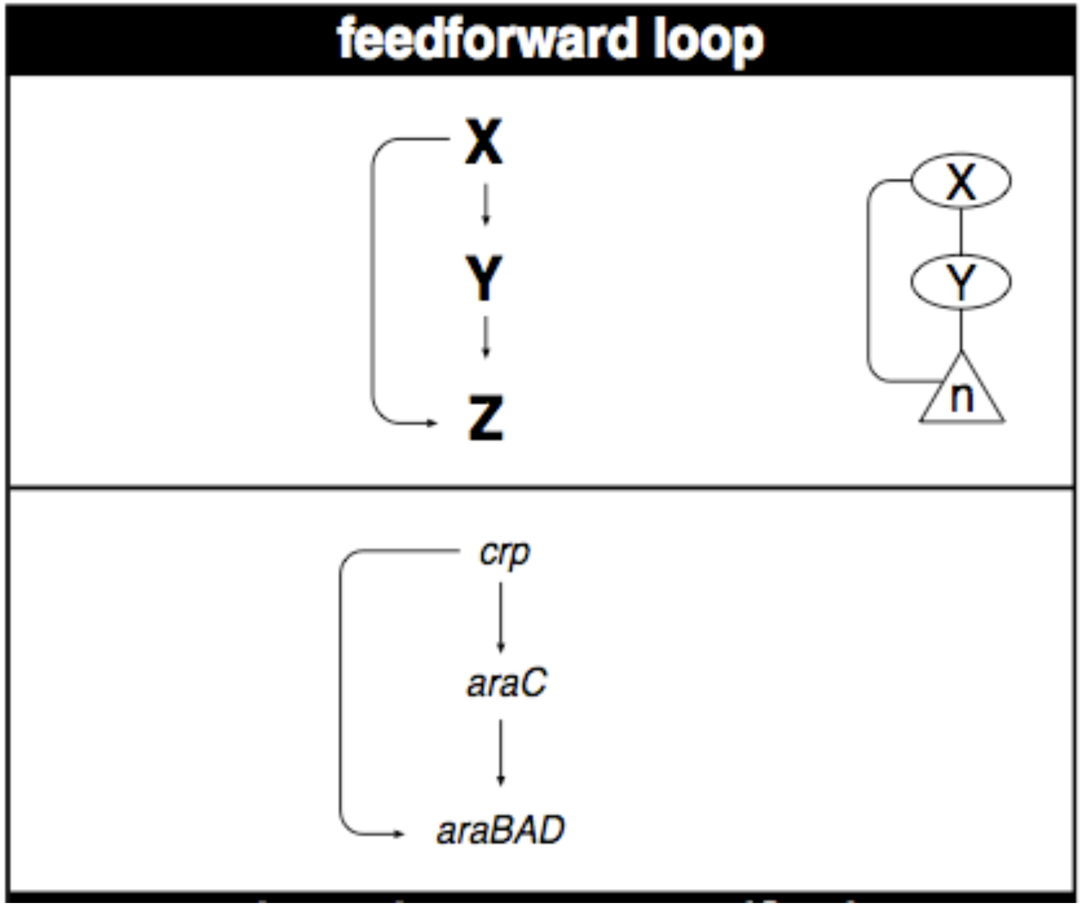
$$\frac{dx}{dt} = \beta - \alpha * x$$

$$x_{st} = \frac{\beta}{\alpha}$$

$$x(t) = x_{st} e^{-\alpha * t}$$

$$t_{\frac{1}{2}} = \frac{\log 2}{\alpha}$$

U. Alon. Network motifs: theory and experimental approaches. Nature Reviews Genetics (2007) vol. 8 (6) pp. 450-461

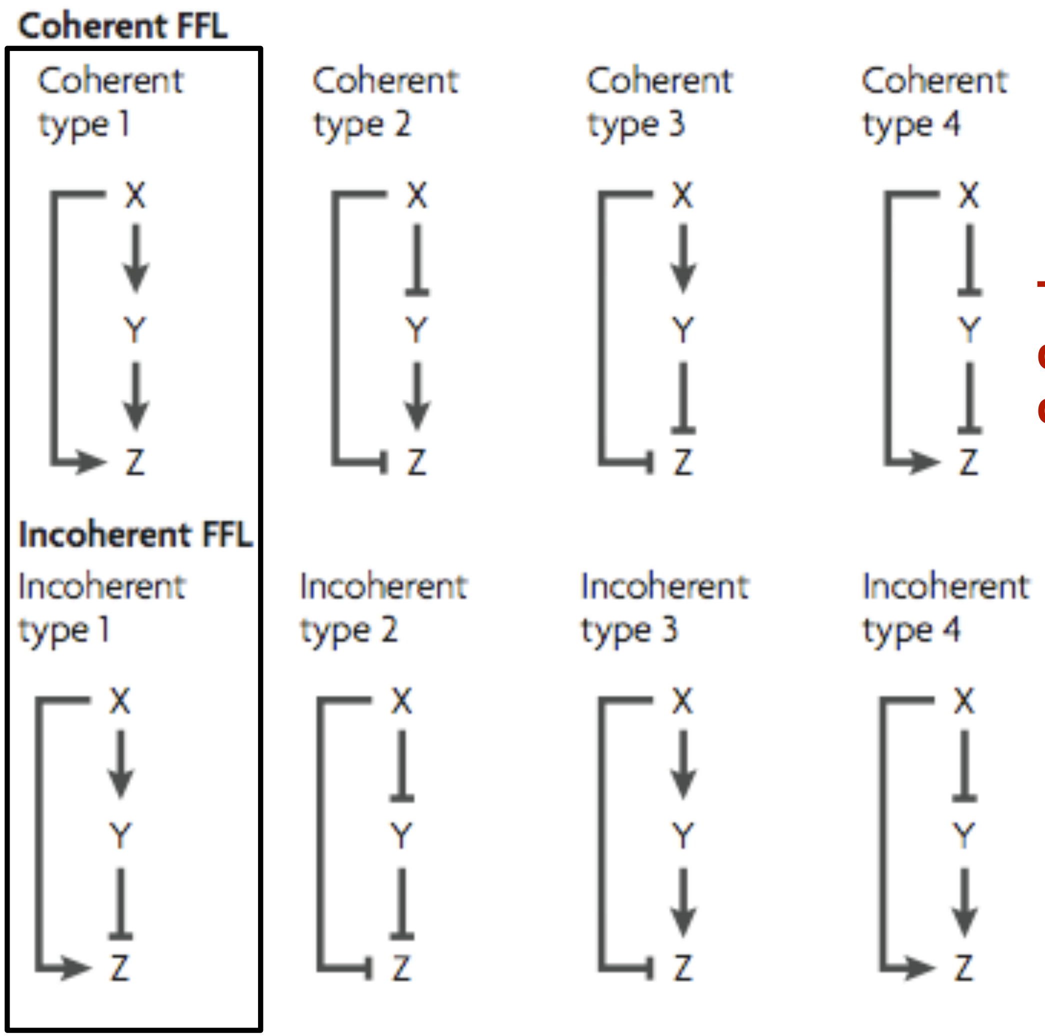


A general transcription factor regulating a second TF, called specific TF, such that both regulate effector operon Z.

In a coherent FFL, the direct effect of the general transcription factor (X) has the same sign (+/-) than the indirect net effect through Y in the effector operon.

If the arrow from X to Z has different sign than the internal ones then the loop is an incoherent FFL

Shai S. Shen-Orr et al., Network motifs in the transcriptional regulation network of Escherichia coli. Nature Genetics 31, 64 - 68 (2002)

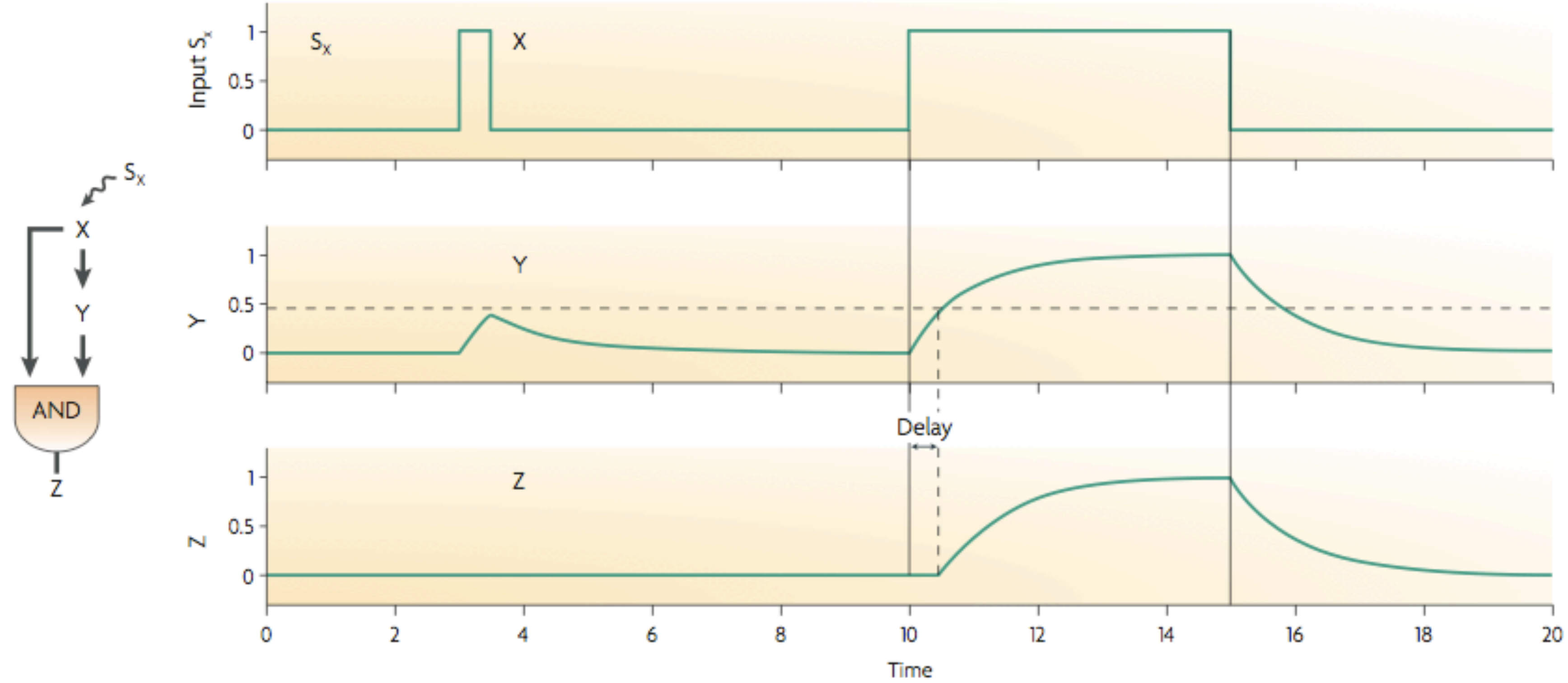


The C1-FFL is a 'sign-sensitive delay' element and a persistence detector.

The I1-FFL is a pulse generator and response accelerator

most common
in E. Coli & S.
Cerevisiae

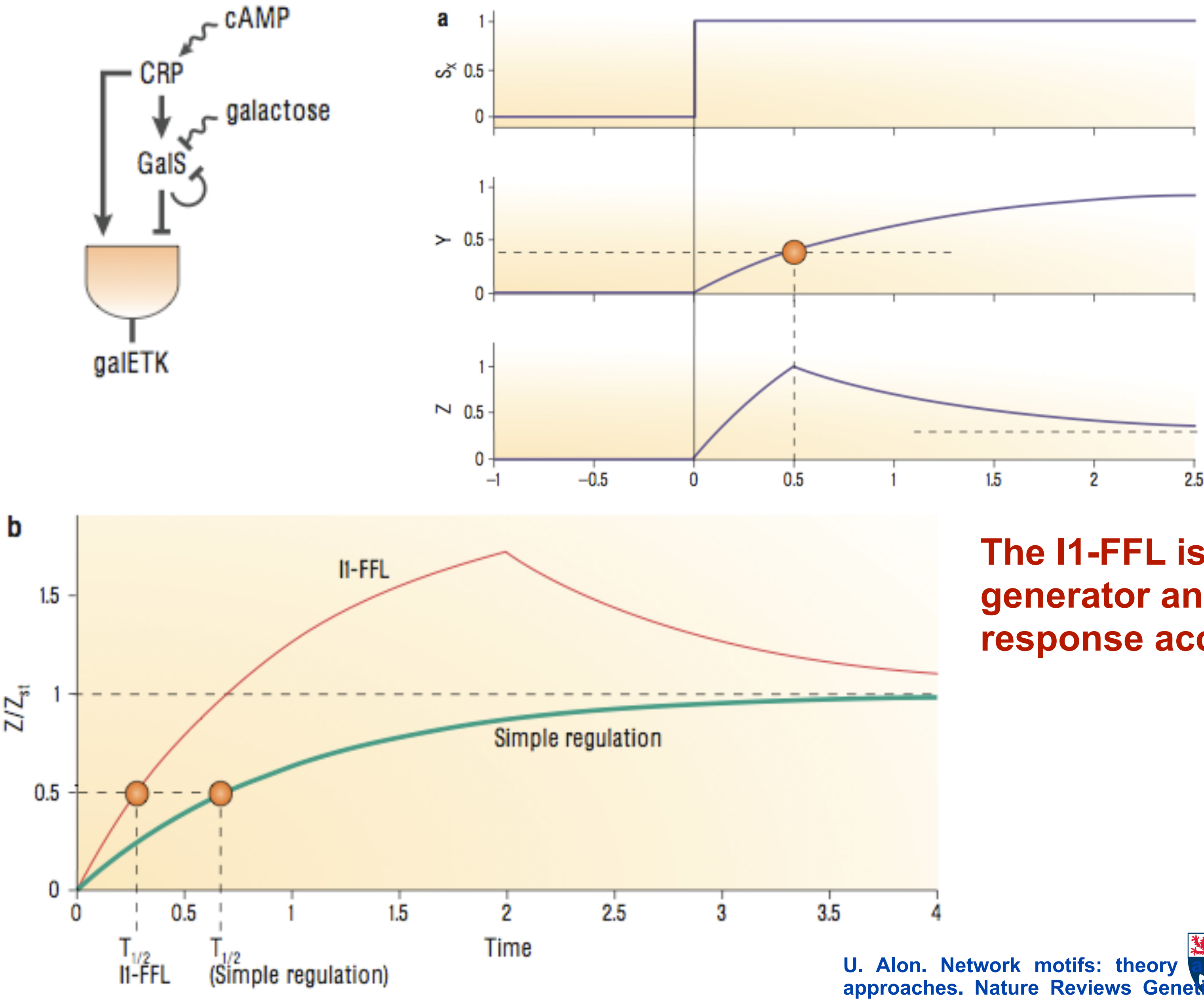
U. Alon. Network motifs: theory and experimental approaches. Nature Reviews Genetics (2007) vol. 8 (6) pp. 450-461



The C1-FFL is a ‘sign-sensitive delay’ element and a persistence detector.

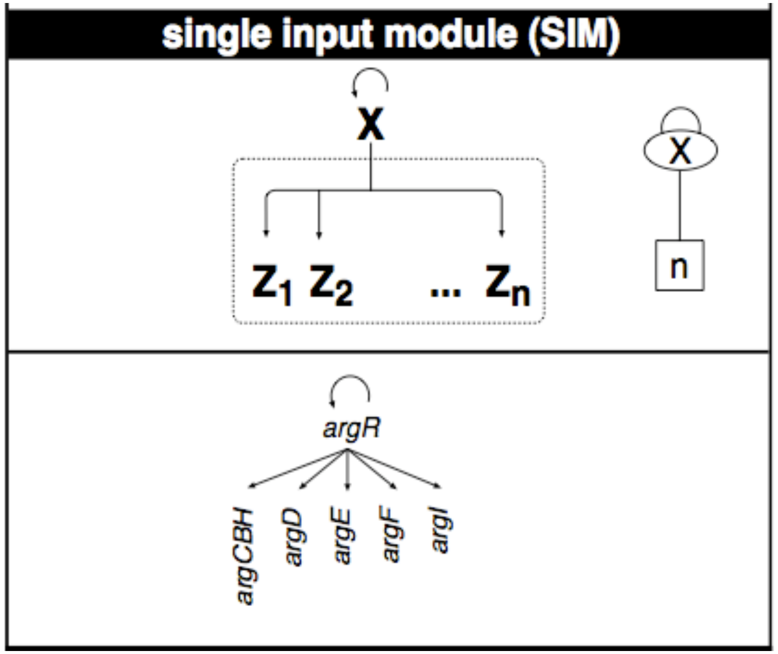
If the integration function is “OR” (rather than “AND”), C1-FFL has no delay after stimulation by Sx but, instead, manifests the delay when the stimulation stops.

U. Alon. Network motifs: theory and experimental approaches. Nature Reviews Genetics (2007) vol. 8 (6) pp. 450-461



The I1-FFL is a pulse generator and response accelerator

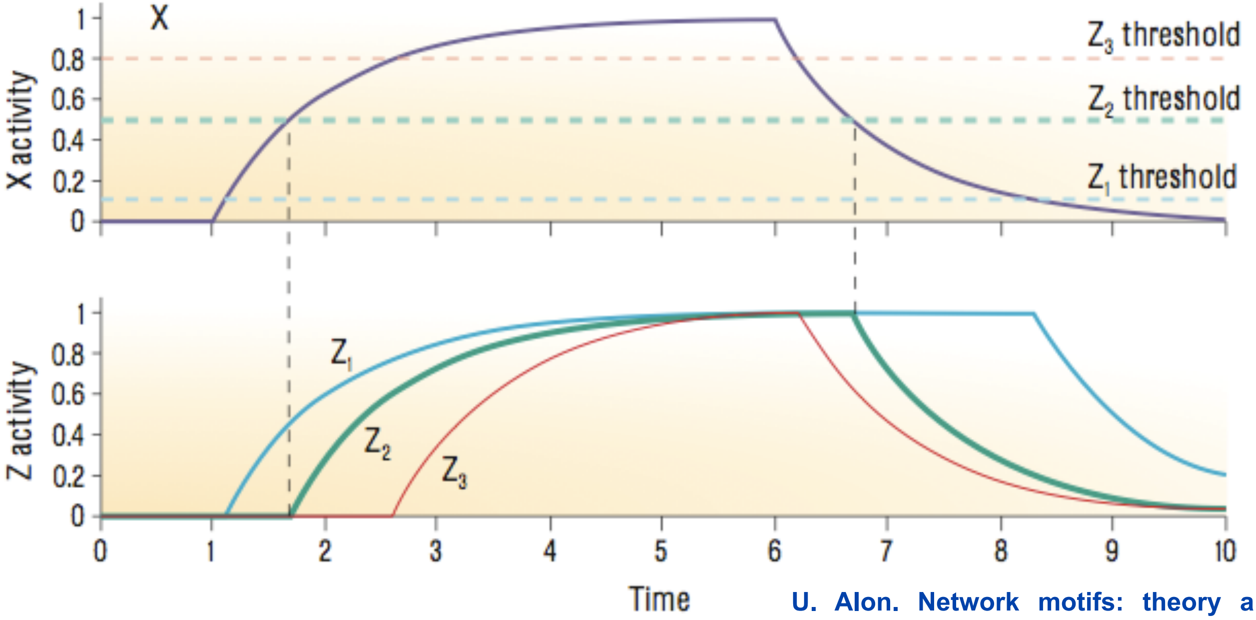
U. Alon. Network motifs: theory and experimental approaches. Nature Reviews Genetics (2007) vol. 8 (6) pp. 450-461



SIM is defined by one TF controlling a set of operons, with the same signs and no additional control.

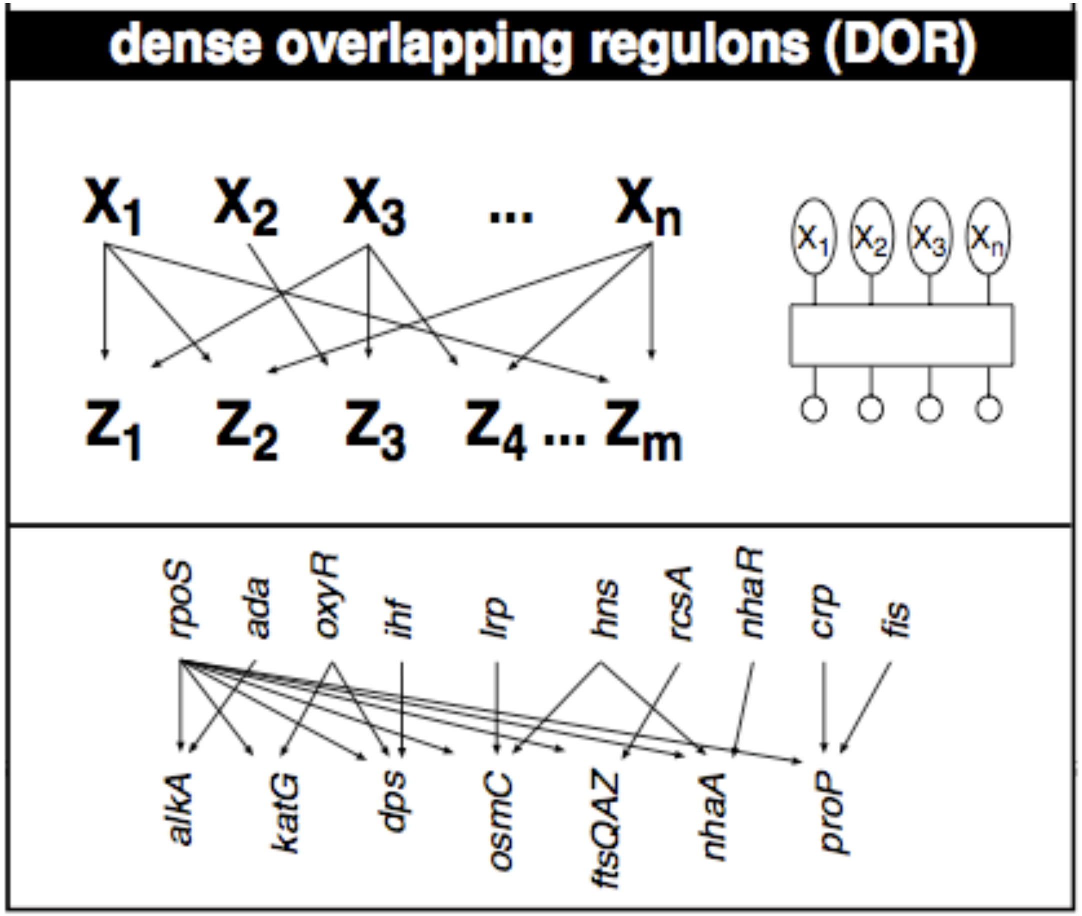
TFs in SIMs are mostly negative autoregulatory (70% in E. *coli*)

Shai S. Shen-Orr et al., Network motifs in the transcriptional regulation network of Escherichia coli. Nature Genetics 31, 64 - 68 (2002)



As the activity of the master regulator X changes in time, it crosses the different activation threshold of the genes in the SIM at different times, this prioritizing the activation of the operons

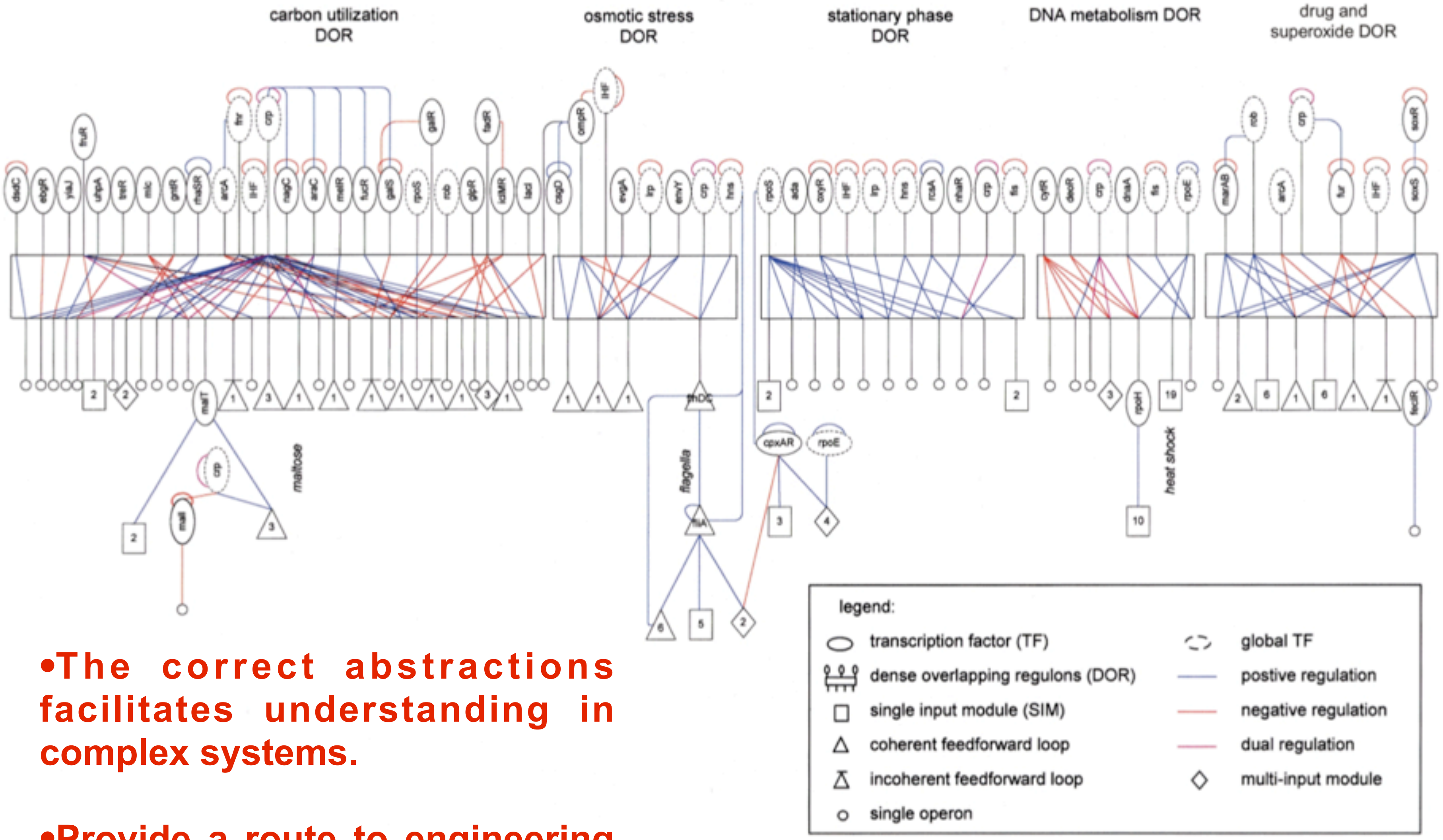
U. Alon. Network motifs: theory and experimental approaches. Nature Reviews Genetics (2007) vol. 8 (6) pp. 450-461



DORs are layers of dense sets of TFs affecting multiple operons.

To understand the specific function of these “gate-arrays” one needs to know the input functions (AND/OR) for each output gene. This data is not currently available in most cases.

Shai S. Shen-Orr et al., Network motifs in the transcriptional regulation network of Escherichia coli. Nature Genetics 31, 64 - 68 (2002)



- **The correct abstractions facilitates understanding in complex systems.**
- **Provide a route to engineering & programming cells.**

Shai S. Shen-Orr et al., Network motifs in the transcriptional regulation network of Escherichia coli. Nature Genetics 31, 64 - 68 (2002)

Centre For Bacterial Cell Biology

Centre for Synthetic Biology and the Bioeconomy

Outline

- The cell as a Computing Device for Biocomputing
- P Systems for Representing Cellular Computation
- Examples of Biocomputation
- Conclusions

P Systems

- Field of membrane computing initiated by Gheorghe Păun in 2000
- Inspired by the hierarchical membrane structure of eukaryotic cells
- A formal language: precisely defined and machine processable
- An executable biology methodology

Centre For Bacterial Cell Biology

Centre for Synthetic Biology and the Bioeconomy

Functional Entities

Container

- A boundary defining self/non-self (symmetry breaking).
- Maintain concentration gradients and avoid environmental damage.

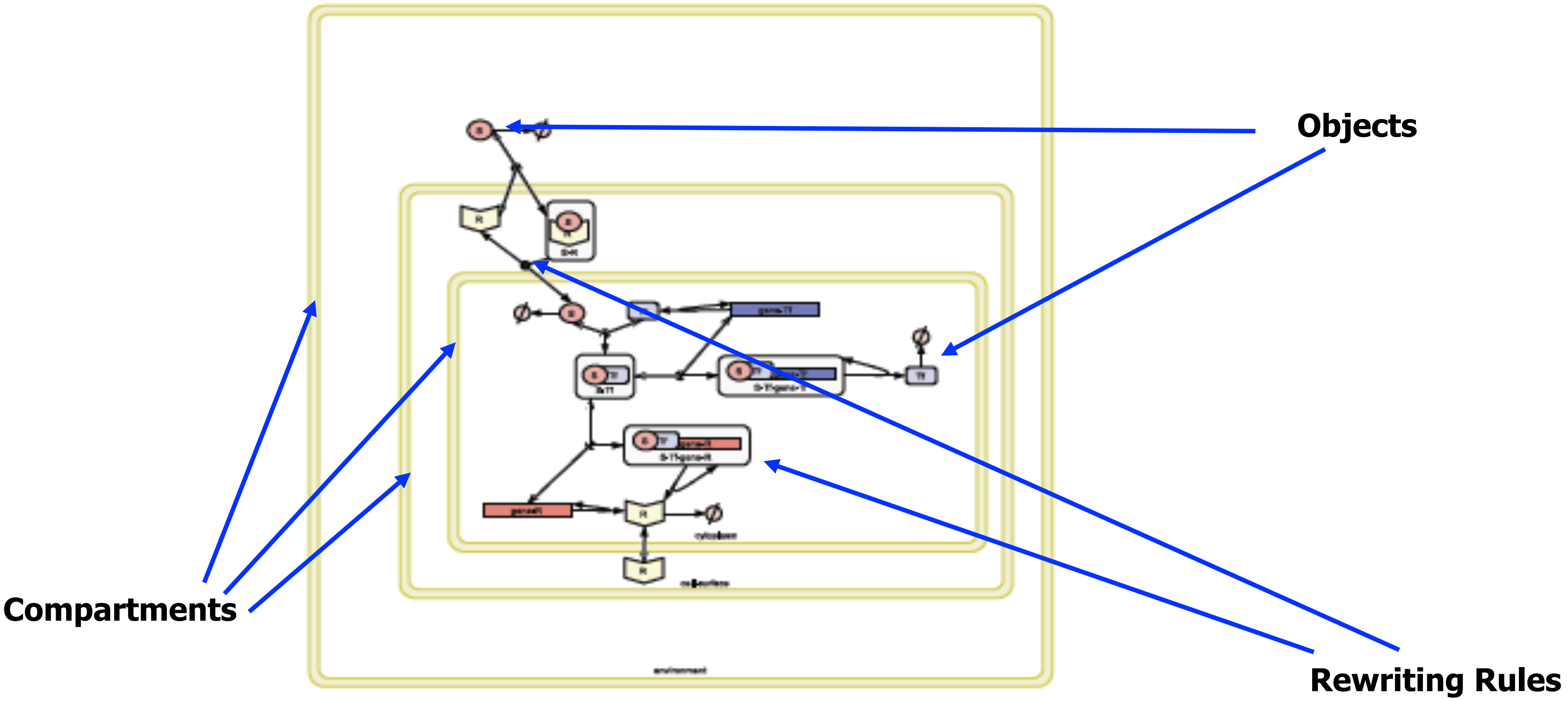
Metabolism

- Confining raw materials to be processed.
- Maintenance of internal structures (autopoiesis).

Information

- Sensing environmental signals / release of signals.
- Genetic information

Distributed and parallel rewritting systems in *compartmentalised hierarchical structures*.



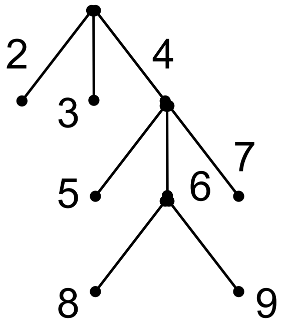
Computational universality and efficiency.

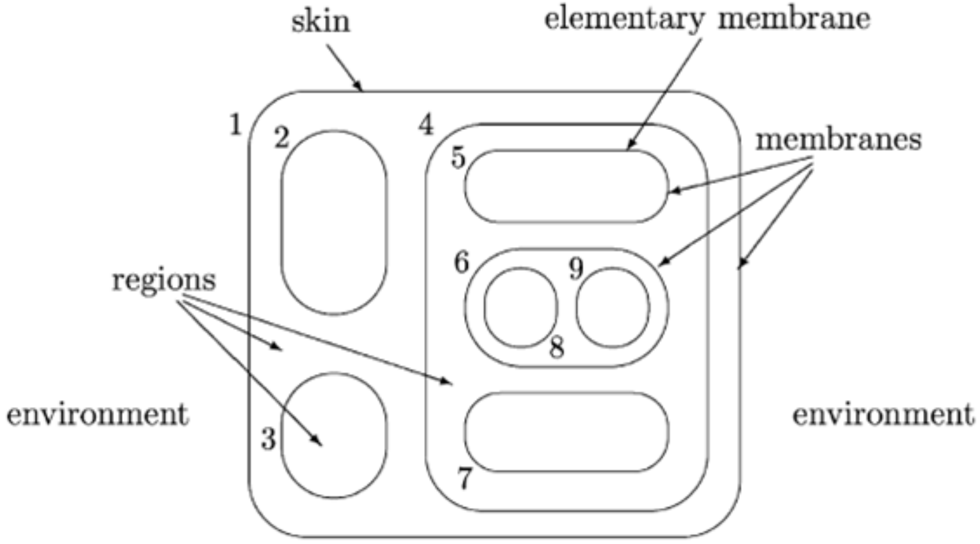
Modelling Framework

Cell-like P systems

Intuitive Visual representation as a Venn diagram with a unique superset and without intersected sets.

formally equivalent to a tree:





the classic P system diagram appearing in most papers (Păun)

- a string of matching parentheses: $[_1 [_2]_2 [_3]_3 [_4 [_5]_5 [_6 [_8]_8 [_9]_9]_6 [_7]_7]_4]_1$

P-Systems: Modelling Principles

Molecules → Letters

Structured Molecules → Strings

Molecular Species → Multisets of letters/strings

Membranes / organelles → Membrane

Biochemical activity → rules

Biochemical transport → Communication rules

Stochastic P Systems

$$\Pi = (O, L, \mu, M_1, M_2, \ldots, M_n, R_{l_1}, \ldots, R_{l_m})$$

- O is a finite alphabet representing objects;
- $L = \{l_1, \ldots, l_m\}$ is a finite alphabet of labels identifying compartments types.
- μ is a membrane structure containing $n \geq 1$ membranes labelled with elements from L.
- $M_i = (l_i, w_i, s_i)$ is the initial configuration of membrane i with $l_i \in L$, $w_i \in O^*$ a finite multiset of objects and s_i a finite set of strings over O.
- $R_{l_t} = \{r_1^{l_t}, \ldots, r_{k_{l_t}}^{l_t}\}$ is a finite set of rewriting rules associated with compartments of type $l_t \in L$.

Rewriting Rules

Rewriting rules on Multiset of OBJECTS

$$r_j^{l_t} \, : \; obj_1 \, [\, obj_2 \,]_l \xrightarrow{c_j^{l_t}} obj'_1 \, [\, obj'_2 \,]_l$$

with $obj_1, obj_2, obj'_1, obj'_2 \in O^*$ *some finite multisets of objects and* l *a label from* L. *A multiset of objects,* obj *is represented as* $obj = o_1 + o_2 + \cdots + o_m$ *with* $o_1, \ldots, o_m \in O$.

Rewriting rules on Multiset of STRINGS

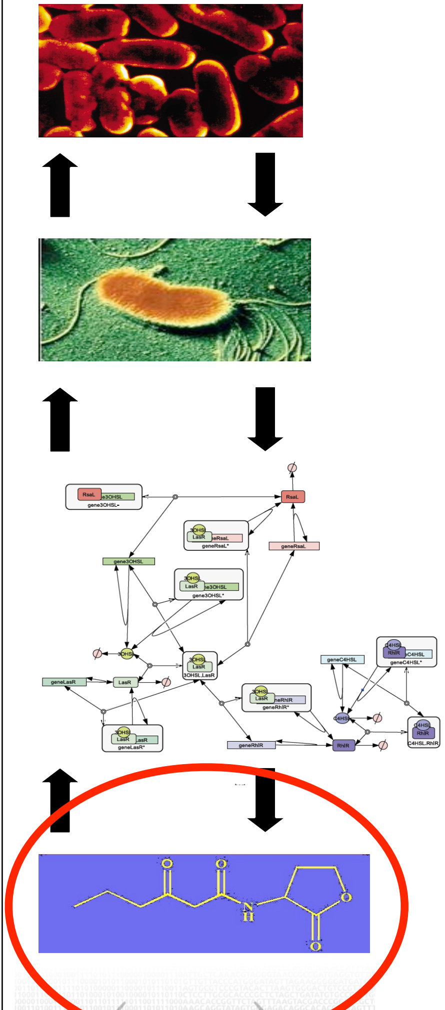
$$r_j^{l_t} \, : \; [\, obj_1 + str_1 \,]_l \xrightarrow{c_j^{l_t}} [\, obj'_1 + str'_1 + \cdots + str'_p \,]_l$$

A string str *is represented as* $str = \langle s_1 . s_2 . \cdots . s_i \rangle$ *where* $s_1, \ldots, s_i \in O$.

A stochastic constant $c_j^{l_t}$ is associated specifically with each rule.

used by Multi-volume Gillespie's algorithm

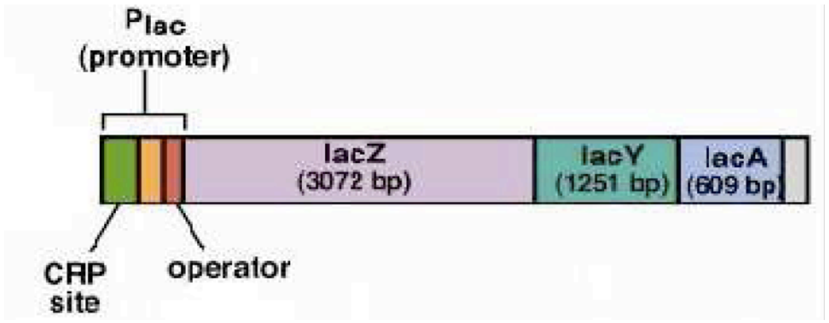
Molecular Species



- A molecular species can be represented using **individual objects**.

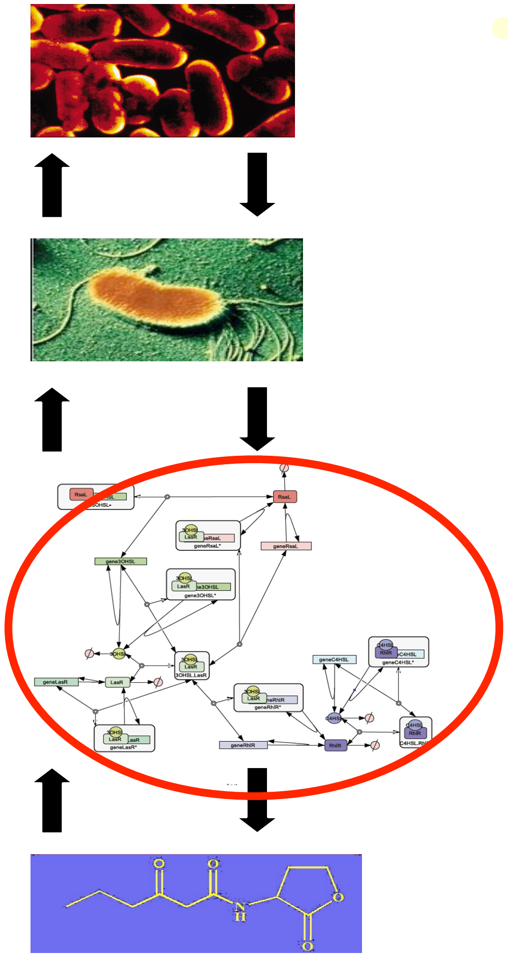
$LacI$

- A molecular species with relevant internal structure can be represented **using a string**.



$$\langle cap \times prom \times op \times lacZ_1 \cdots lacZ_{30} \times lacY_1 \cdots lacY_{12} \times lacA_1 \cdots lacA_6 \rangle$$

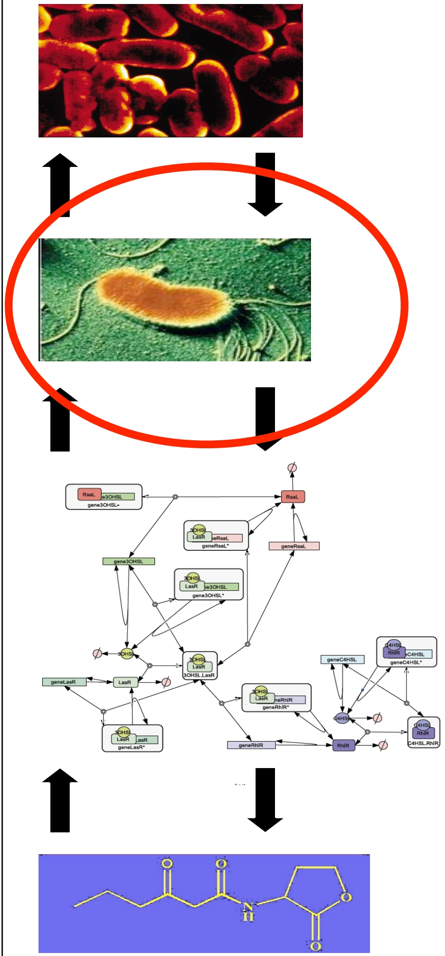
Molecular Interactions



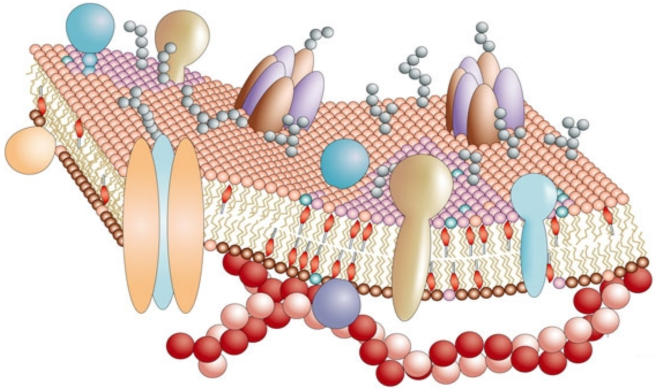
- **Comprehensive and relevant rule-based schema** for the most common molecular interactions taking place in living cells.

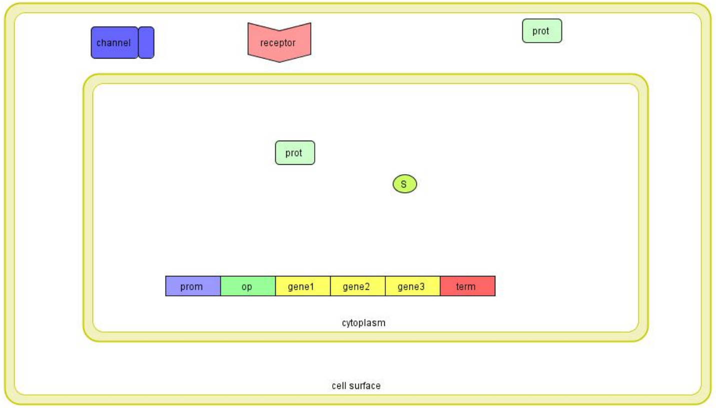
Transformation/Degradation
Complex Formation and Dissociation
Diffusion in / out
Binding and Debinding
Recruitment and Releasing
Transcription Factor Binding/Debinding
Transcription/Translation

Compartments / Cells

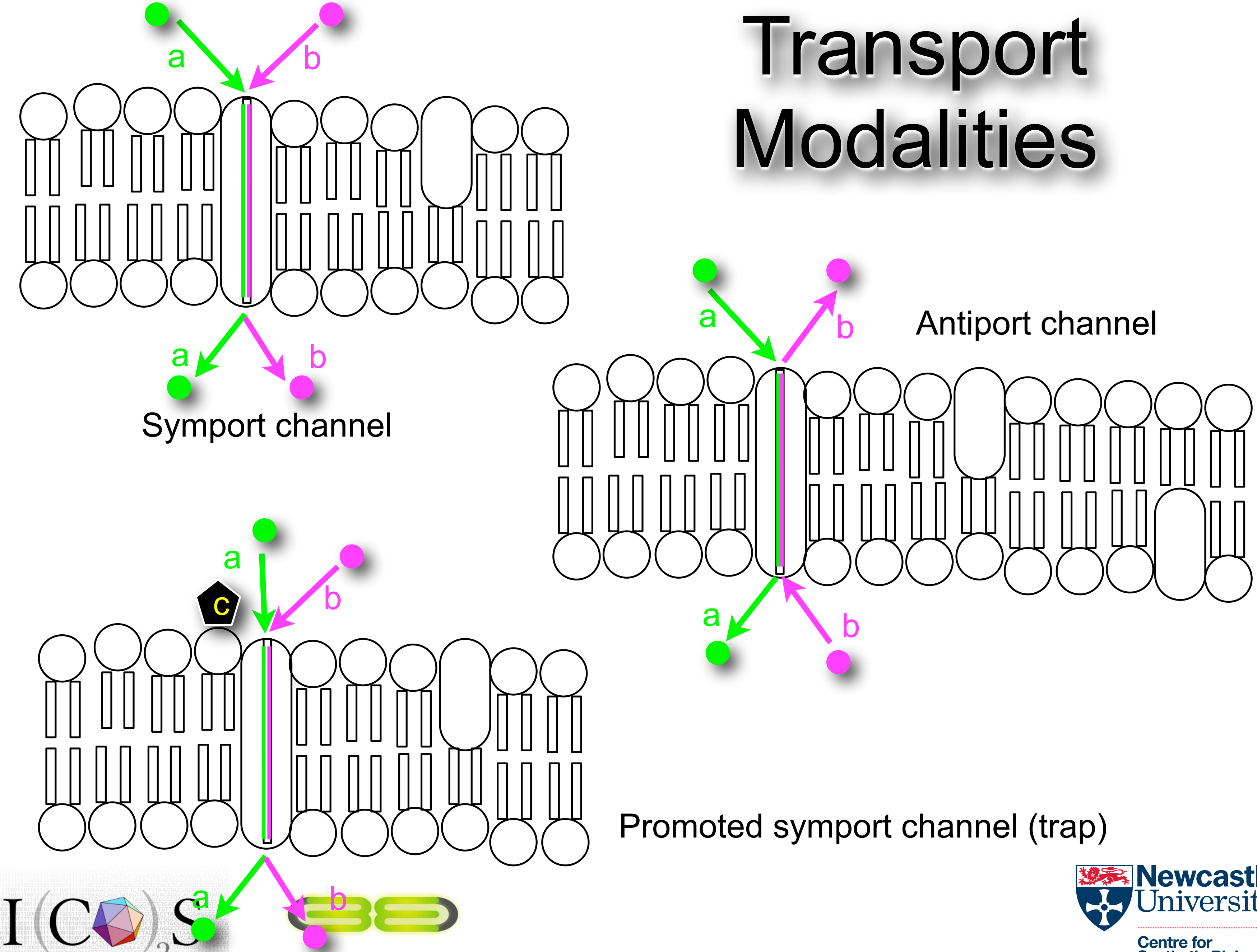


- **Compartments and regions are explicitly** specified using membrane structures.





Transport Modalities

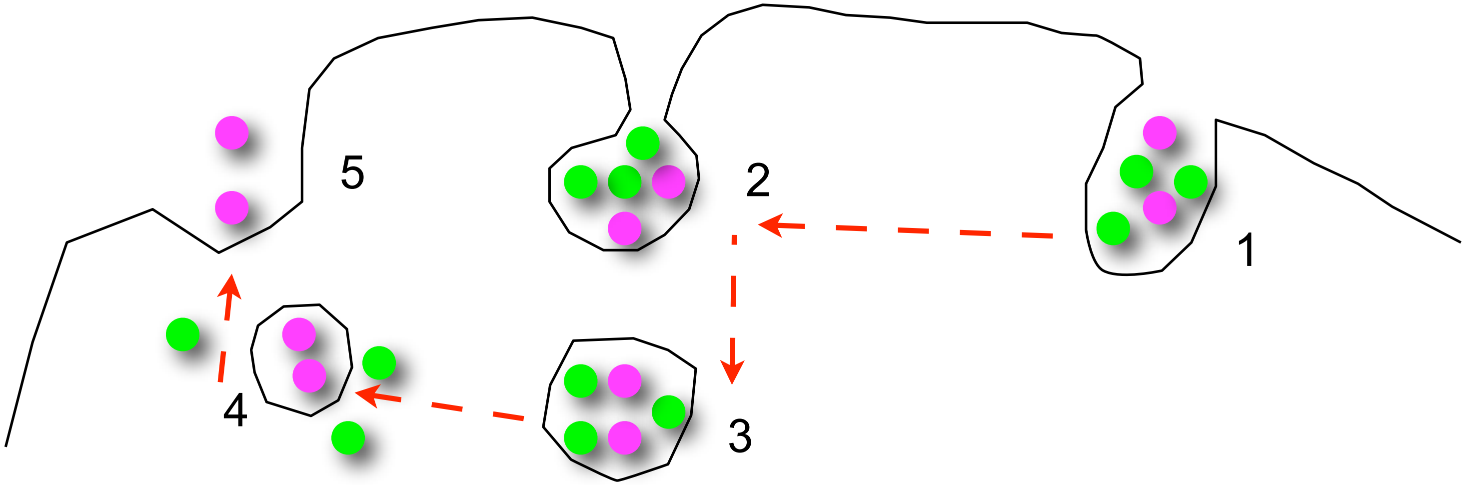


Symport channel

Antiport channel

Promoted symport channel (trap)

Transport Modalities

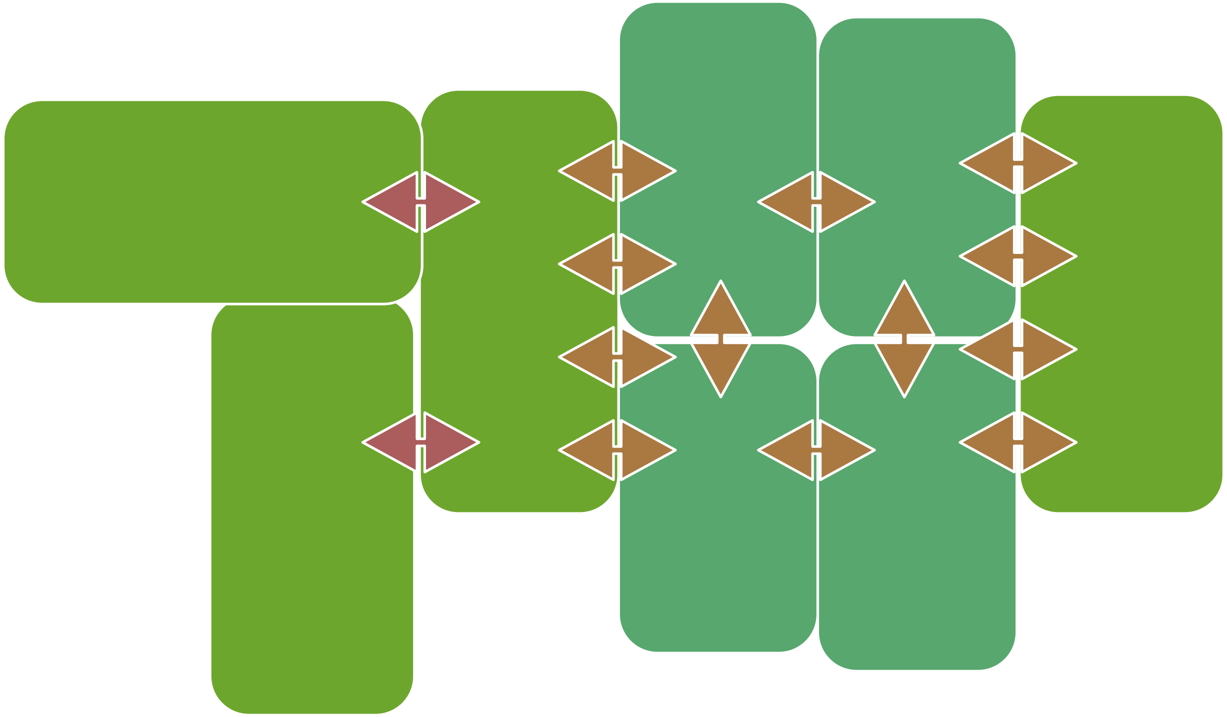


Endocitosys → Phagocitosys

Endocitosys → Pinocitosys

Exocitosys

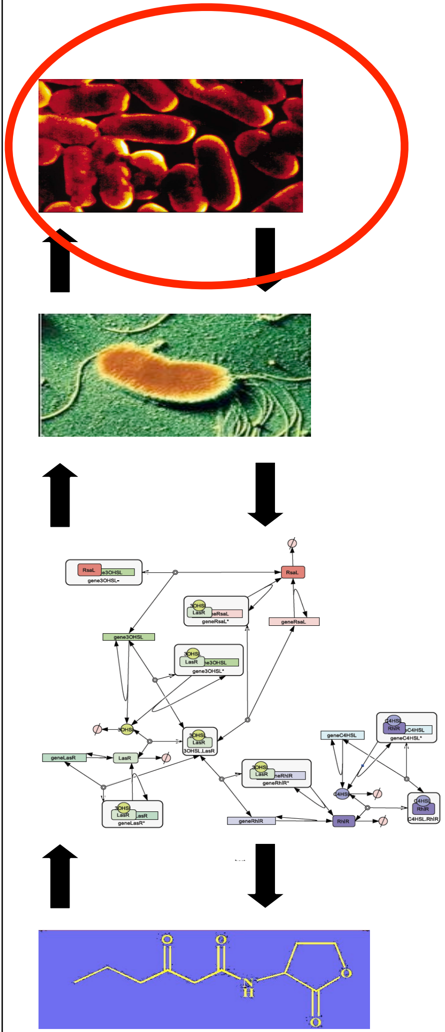
Transport Modalities



Highly specific:

cell specific & topology specific

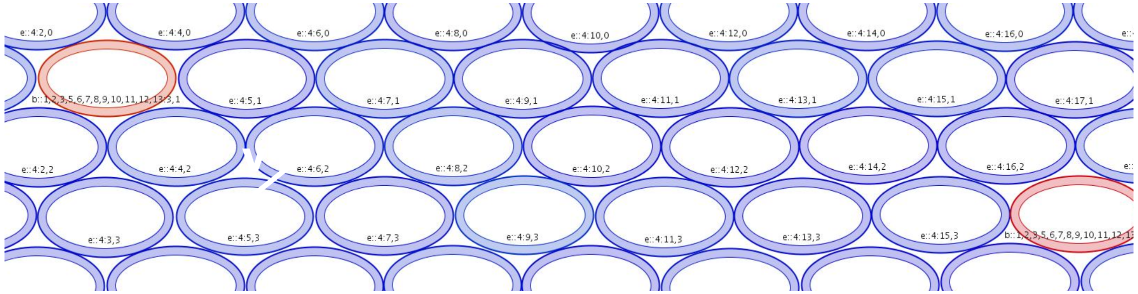
Colonies / Tissues



- Colonies and tissues are representing as **collection** of P systems distributed over a lattice.

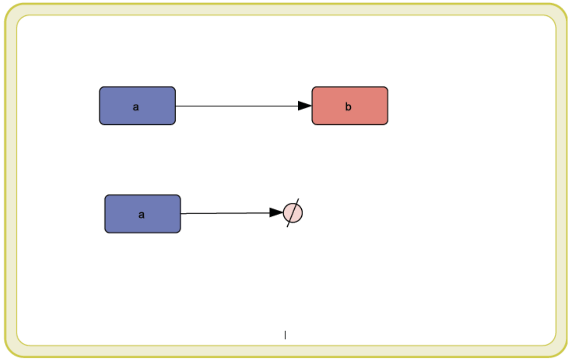
- Objects can travel around the lattice through **translocation** rules.

$$[\,obj\,]_l \overset{\mathbf{v}}{\rightleftharpoons} [\ \ ] \xrightarrow{c_4^l} [\ ]_l \overset{\mathbf{v}}{\rightleftharpoons} [\,obj\,]$$



Molecular Interactions Inside Compartments

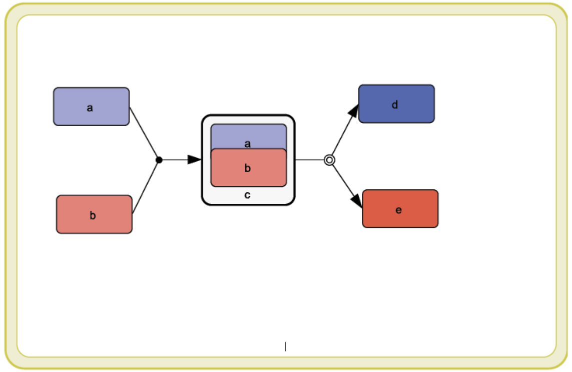
Transformation and degradation:



$$
\begin{aligned}
r_1 &: \quad [\, a \,]_l \xrightarrow{c} [\, b \,]_l \\
r_2 &: \quad [\, a \,]_l \xrightarrow{c} [\ ]_l
\end{aligned}
$$

$$p(r_i) = c \cdot |a| \qquad i = 1, 2$$

Complex formation and dissociation:



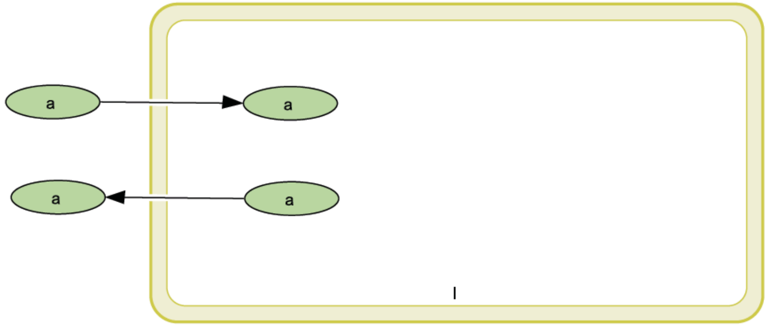
$$
\begin{aligned}
r_{cf} &: \quad [\, a + b \,]_l \xrightarrow{c_{cf}} [\, c \,]_l \\
r_{cd} &: \quad [\, c \,]_l \xrightarrow{c_{cd}} [\, d + e \,]_l
\end{aligned}
$$

$$
p(r_{cf}) = \begin{cases} c_{cf} \cdot |a||b| & a \neq b \\ c_{cf} \cdot \dfrac{|a|(|a| - 1)}{2} & a = b \end{cases}
$$

$$p(r_{cd}) = c_{cd} \cdot |c|$$

Passive Diffusion of Molecules

Diffusion in and out:



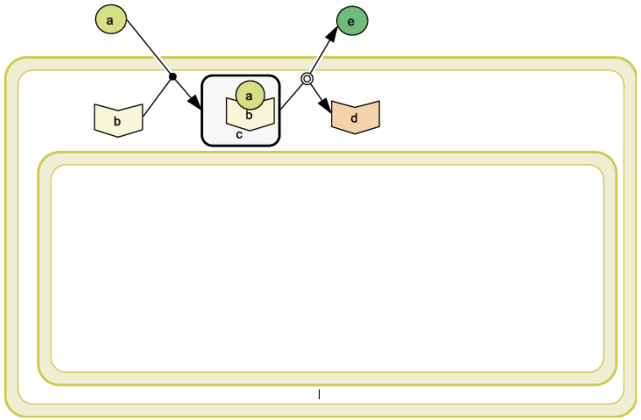
$$
\begin{array}{llll}
r_1 : & a\,[\ ]_l \xrightarrow{c_{in}} [\,a\,]_l & & p(r_1) = c_{in}|a| \\
r_2 : & [\,a\,]_l \xrightarrow{c_{out}} a\,[\ ]_l & & p(r_2) = c_{out}|a|
\end{array}
$$

Centre For Bacterial Cell Biology

Centre for
Synthetic Biology
and the Bioeconomy

Signal Sensing and Active Transport

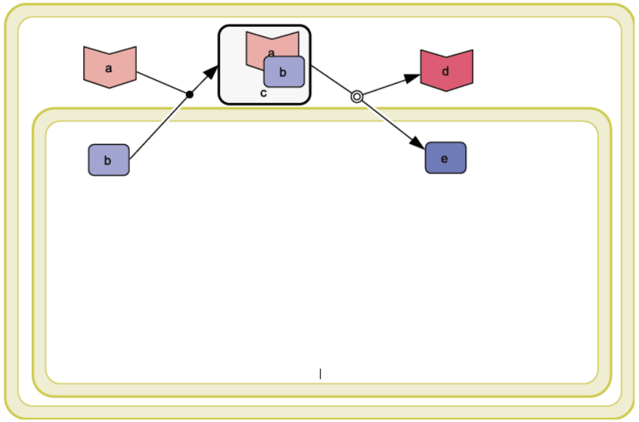
Binding and debinding:



$$r_1 : \quad a\,[\,b\,]_l \xrightarrow{c_{lb}} [\,c\,]_l$$
$$r_2 : \quad [\,c\,]_l \xrightarrow{c_{ld}} d\,[\,e\,]_l$$

$$p(r_1) = c_{lb}|a||b|$$
$$p(r_2) = c_{ld}|c|$$

Recruitment and releasing:

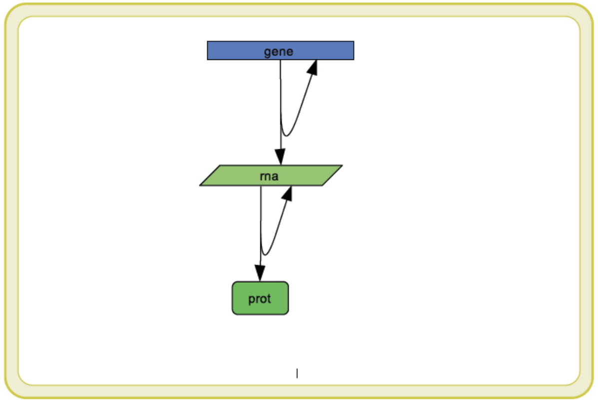


$$r_{cf} : a\,[\,b\,]_l \xrightarrow{c_{cf}} c\,[\quad]_l$$
$$r_{cd} : c\,[\quad]_l \xrightarrow{c_{cd}} d\,[\,e\,]_l$$

$$p(r_{rt}) = c_{rt}|a||b|$$
$$p(r_{rl}) = c_{rl}|c|$$

Specification of Transcriptional Regulatory Networks

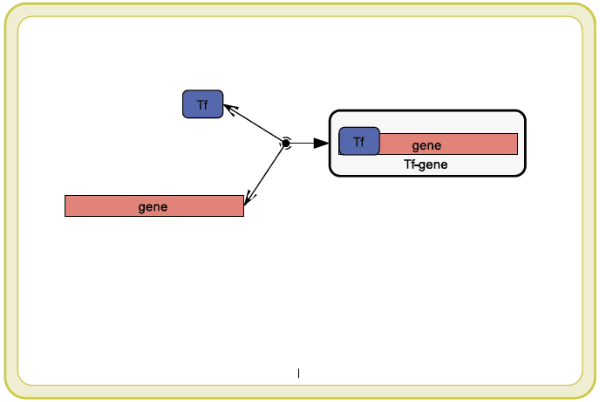
Transcription and Translation:



$$r_{tc} : [\ gene\ ]_l \xrightarrow{c_{tc}} [\ gene + rna\ ]_l$$
$$r_{tl} : [\ rna\ ]_l \xrightarrow{c_{tl}} [\ rna + prot\ ]_l$$

$$p(r_{tc}) = c_{tc}|gene|$$
$$p(r_{tl}) = c_{tl}|rna|$$

Transcription factor binding and debinding:

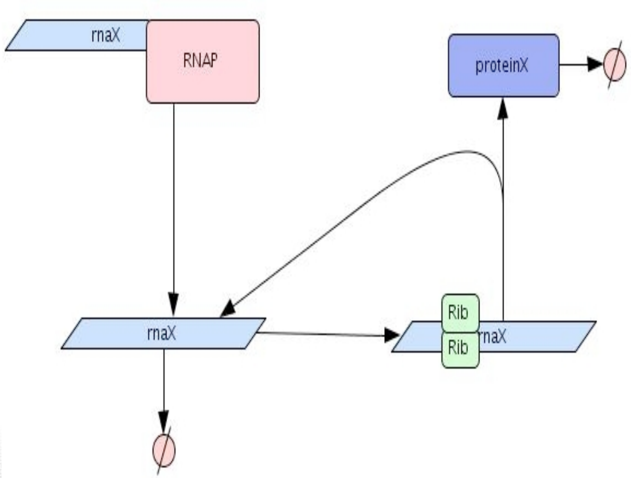


$$r_{gon} : [\ Tf + gene\ ]_l \xrightarrow{c_{gon}} [\ Tfgene\ ]_l$$
$$r_{goff} : [\ Tfgene\ ]_l \xrightarrow{c_{goff}} [\ Tf + gene\ ]_l$$

$$p(r_{gon}) = c_{gon}|Tf||gene|$$
$$p(r_{goff}) = c_{goff}|Tfgene|$$

Post-Transcriptional Processes

- For each protein in the system, post-transcriptional processes like translational initiation, messenger and protein degradation, protein dimerisation, signal sensing, signal diffusion etc are represented using modules of rules.
- Modules can have also as parameters the stochastic kinetic constants associated with the corresponding rules in order to allow us to explore possible mutations in the promoters and ribosome binding sites in order to optimise the behaviour of the system.



$$PostTransc(\{X\},\{c_1,\ldots,c_5\}) = \begin{cases} [\, rnaX\cdot RNAP\,]_b \xrightarrow{c_1} [\, rnaX\,]_b \\ [\, rnaX\,]_b \xrightarrow{c_2} [\ ]_b \\ [\, rnaX\,]_b \xrightarrow{c_3} [\, rnaX + proteinX\cdot Rib\,]_b \\ [\, proteinX\cdot Rib\,]_b \xrightarrow{c_4} [\, proteinX\,]_b \\ [\, proteinX\,]_b \xrightarrow{c_5} [\ ]_b \end{cases}$$

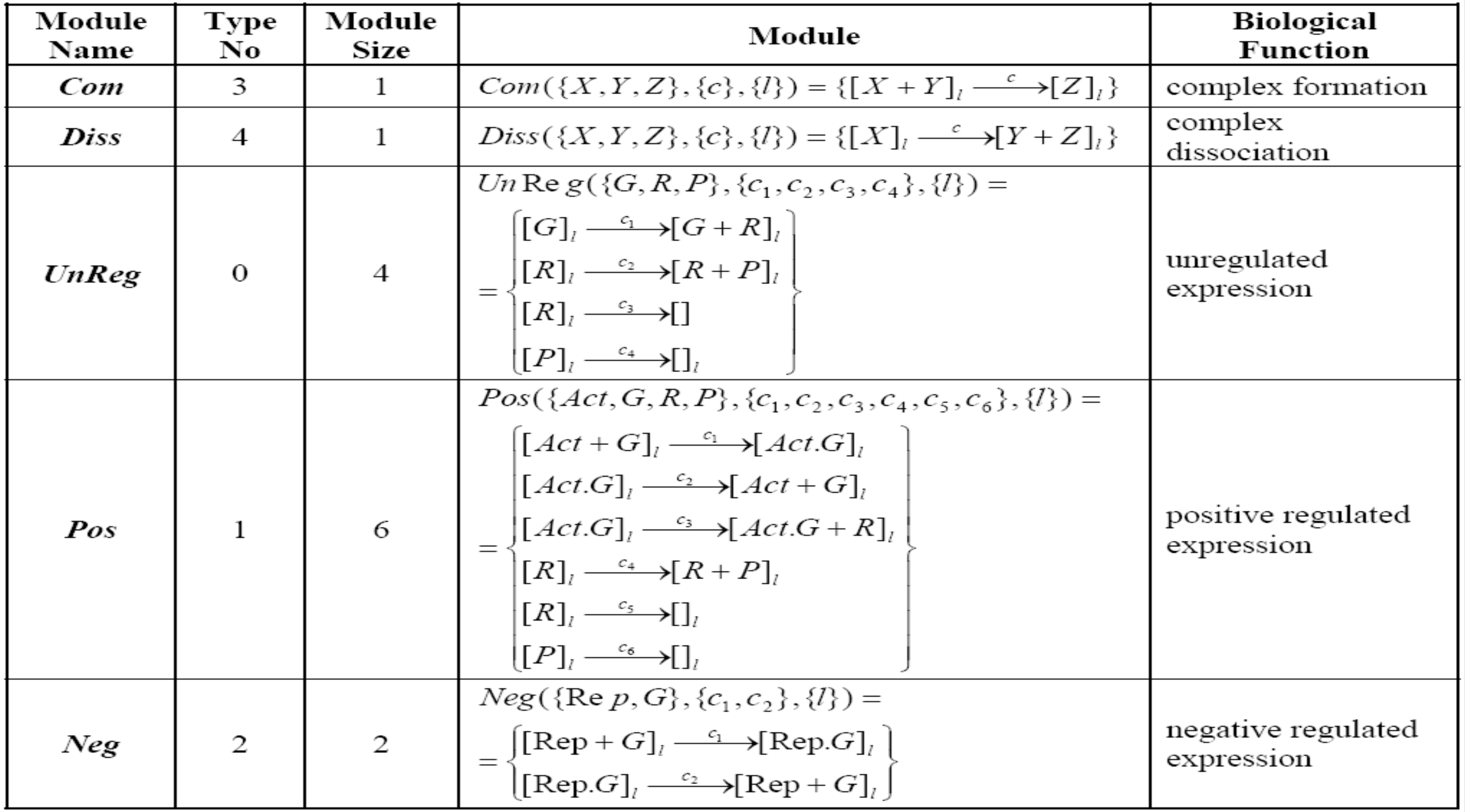
Scalability through Modularity

Cellular functions arise from **orchestrated interactions between motifs** consisting of many molecular interacting species.

A ***P System model*** is a **set of rules** representing molecular interactions motifs that appear in many cellular systems.

F. J. Romero-Campero, J. Twycross, M. Camara, M. Bennett, M. Gheorghe, and N. Krasnogor. Modular assembly of cell systems biology models using p systems. *International Journal of Foundations of Computer Science*, 20(3):427-442, 2009.

Basic P System Modules Used



Module Name	Type No	Module Size	Module	Biological Function
Com	3	1	$Com(\{X,Y,Z\},\{c\},\{l\}) = \{[X+Y]_l \xrightarrow{c} [Z]_l\}$	complex formation
Diss	4	1	$Diss(\{X,Y,Z\},\{c\},\{l\}) = \{[X]_l \xrightarrow{c} [Y+Z]_l\}$	complex dissociation
UnReg	0	4	$Un\mathrm{Re}\,g(\{G,R,P\},\{c_1,c_2,c_3,c_4\},\{l\}) = \left\{\begin{matrix}[G]_l \xrightarrow{c_1} [G+R]_l \\ [R]_l \xrightarrow{c_2} [R+P]_l \\ [R]_l \xrightarrow{c_3} [\,] \\ [P]_l \xrightarrow{c_4} [\,]_l\end{matrix}\right\}$	unregulated expression
Pos	1	6	$Pos(\{Act,G,R,P\},\{c_1,c_2,c_3,c_4,c_5,c_6\},\{l\}) = \left\{\begin{matrix}[Act+G]_l \xrightarrow{c_1} [Act.G]_l \\ [Act.G]_l \xrightarrow{c_2} [Act+G]_l \\ [Act.G]_l \xrightarrow{c_3} [Act.G+R]_l \\ [R]_l \xrightarrow{c_4} [R+P]_l \\ [R]_l \xrightarrow{c_5} [\,]_l \\ [P]_l \xrightarrow{c_6} [\,]_l\end{matrix}\right\}$	positive regulated expression
Neg	2	2	$Neg(\{\mathrm{Re}\,p,G\},\{c_1,c_2\},\{l\}) = \left\{\begin{matrix}[\mathrm{Rep}+G]_l \xrightarrow{c_1} [\mathrm{Rep}.G]_l \\ [\mathrm{Rep}.G]_l \xrightarrow{c_2} [\mathrm{Rep}+G]_l\end{matrix}\right\}$	negative regulated expression

Characterisation/Encapsulation of Cellular Parts: Gene Promoters

- ❑ A **modeling language** for the design of synthetic bacterial colonies.
- ❑ A ***module***, set of rules describing the molecular interactions involving a cellular part, provides **encapsulation and abstraction**.
- ❑ Collection or **libraries of reusable cellular parts and reusable models.**

PluxOR1({X},{c1, c2, c3, c4, c5, c6, c7, c8, c9},{l}) = {

type: promoter

sequence: ACCTGTAGGATCGTACAGGTTTACGCAAGAA
ATGGTTTGTATAGTCGAATACCTCTGGCGGTGATA

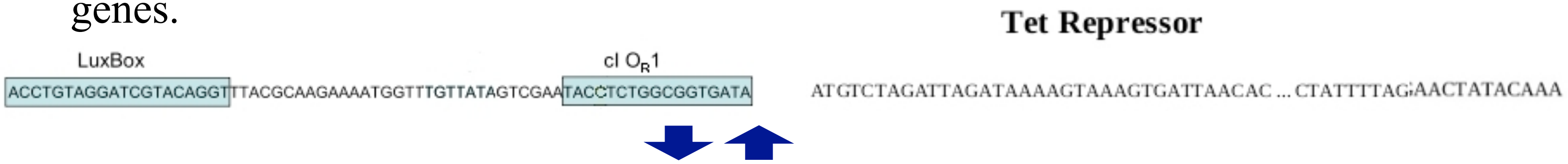
rules:
r1(c1): [LuxR2 + PluxPR.X] => [PluxPR.LuxR2.X]
r2(c2): [PluxPR.LuxR2.X] => [LuxR2 + PluxPR.X]
...
r5(c5): [CI2 + PluxPR.X] => [PluxPR.CI2.X]
r6(c6): [PluxPR.CI2.X] => [CI2 + PluxPR.X]
...
r9(c9): [PluxPR.LuxR2.X] => [PluxPR.LuxR2.X + RNAP.X]

}

E. Davidson (2006) The Regulatory Genome, Gene Regulation Networks in Development and Evolution, Elsevier

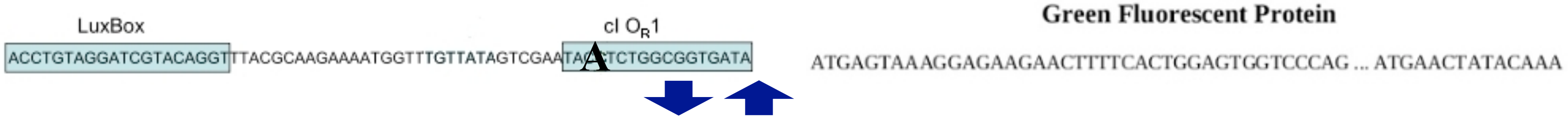
Module Variables: Recombinant DNA, Directed Evolution, Chassis selection

- ☐ **Recombinant DNA:** Objects variables can be instantiated with the name of specific genes.



PluxOR1({X=GFP})

- ☐ **Directed evolution:** Variables for stochastic constants can be instantiated with specific values.



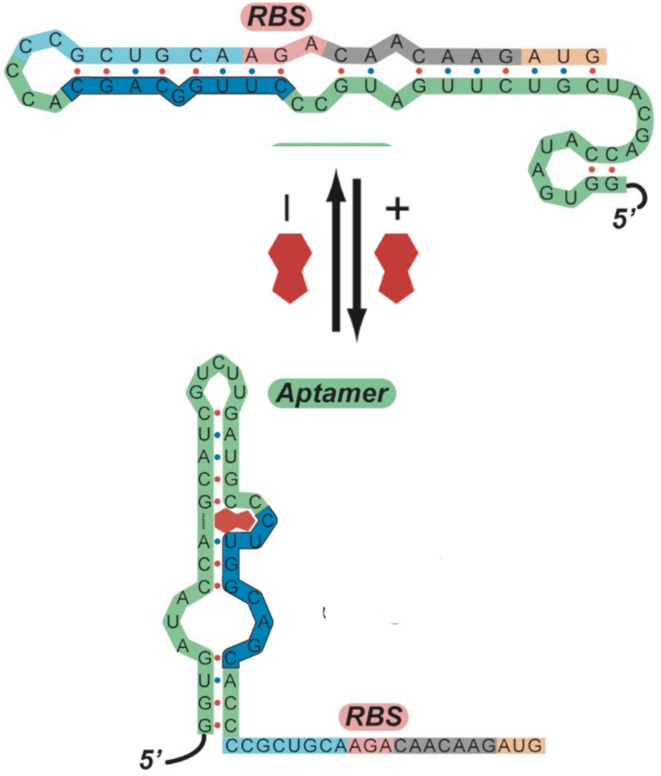
PluxOR1({X=GFP},{...,c4=10,...})

- ☐ **Chassis Selection:** The variable for the label can be instantiated with the name of a chassis.

PluxOR1({X=GFP},{...,c4=10,...},{l=DH5α })

Characterisation/Encapsulation of Cellular Parts: Riboswitches

☐ A **riboswitch** is a piece of RNA that *folds* in different ways depending on the presence or absence of specific molecules regulating translation.



ToppRibo({X},{c1, c2, c3, c4, c5, c6},{l}) = {

type: riboswitch

sequence:GGTGATACCAGCATCGTCTTGATGCCCTTGG
CAGCACCCCGCTGCAAGACAACAAGATG

rules:
r1(c1): [RNAP.ToppRibo.X] => [ToppRibo.X]
r2(c2): [ToppRibo.X] => [ ]
r3(c3): [ToppRibo.X + theop] => [ToppRibo*.X]
r4(c4): [ToppRibo*.X] => [ToppRibo.X + theop]
r5(c5): [ToppRibo*.X] => [ ]
r6(c6): [ToppRibo*.X] => [ToppRibo*.X + Rib.X]

}

Characterisation/Encapsulation of Cellular Parts: Degradation Tags

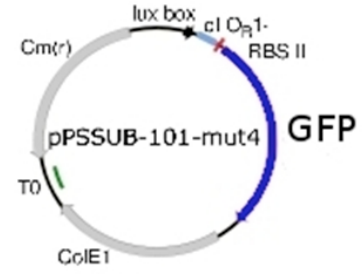
☐ **Degradation tags** are amino acid sequences recognised by proteases. Once the corresponding DNA sequence is fused to a gene the half life of the protein is reduced considerably.

```
degLVA({X},{c1, c2},{l}) = {

    type:  degradation tag

    sequence: CAGCAAACGACGAAAACTACGCTTTAGTAGCT

    rules:
        r1(c1): [ Rib.X.degLVA ] => [ X.degLVA ]
        r2(c2): [ X.degLVA  ] => [  ]
}
```

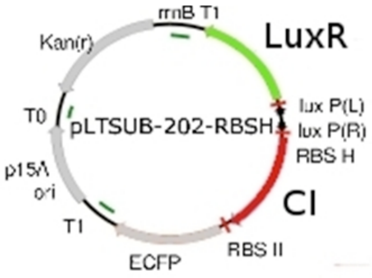
Higher Order Modules: Building Synthetic Gene Circuits

PluxOR1	ToppRibo	geneX	degLVA

```
3OC6_repressible_sensor({X}) = {
      PluxOR1({X=ToppRibo.geneX.degLVA},{...},{l=DH5α})
      ToppRibo({X=geneX.degLVA},{...},{l=DH5α})
      degLVA({X},{...},{l=DH5α})
}
```



```
Plux({X=ToppRibo.geneCI.degLVA},{...},{l=DH5α})
ToppRibo({X=geneCI.degLVA},{...},{l=DH5α})
degLVA({CI},{...},{l=DH5α})

 PtetR({X=ToppRibo.geneLuxR.degLVA},{...},{l=DH5α})
 Weiss_RBS({X=LuxR},{...},{l=DH5α})
  Deg({X=LuxR},{...},{l=DH5α})
```



Stochastic P Systems Are ***Executable*** Programs

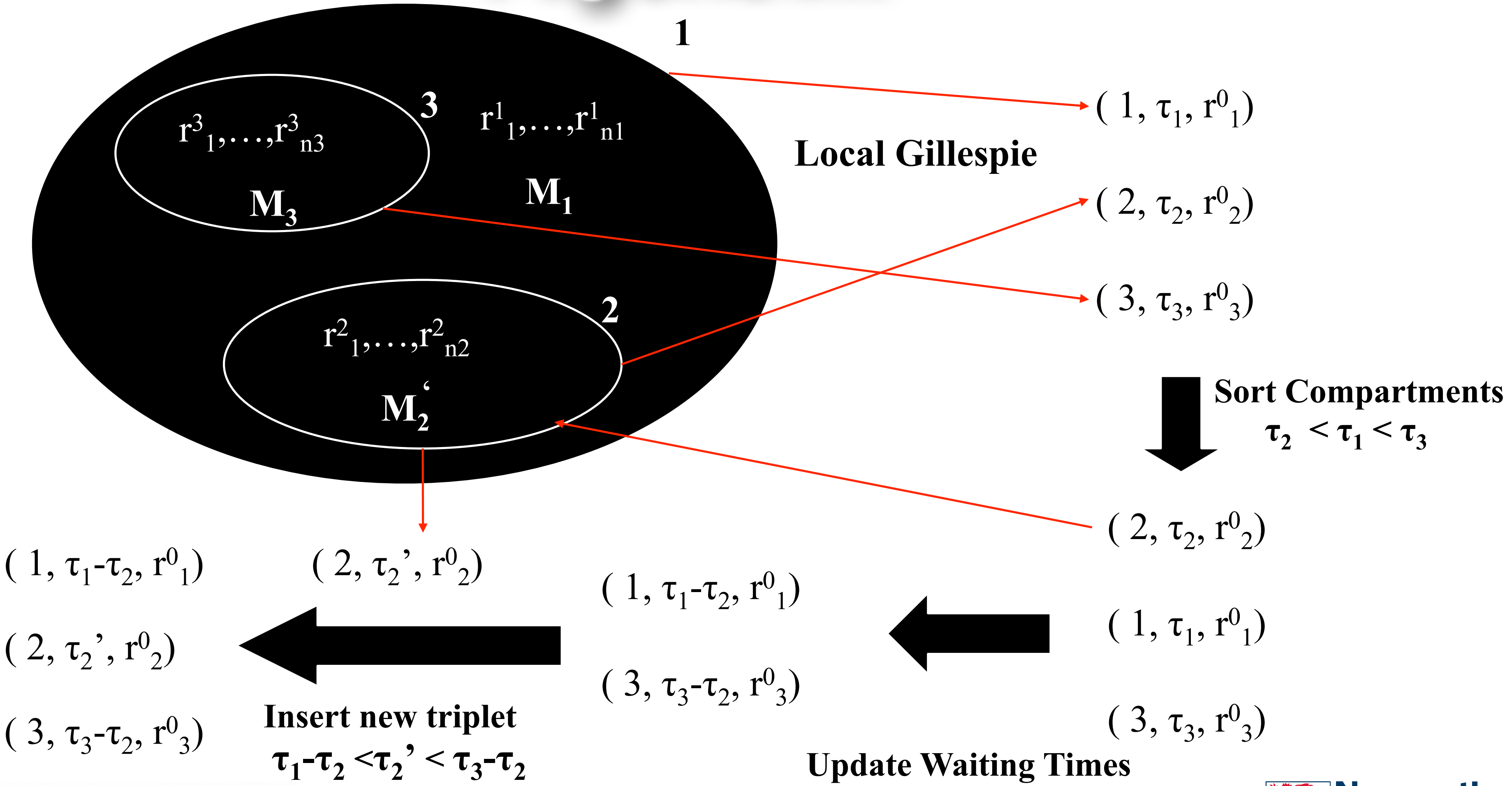
The virtual machine running these programs is a "Gillespie Algorithm (SSA)". It generates trajectories of a stochastic system:

A stochastic constant is associated with each rule.

A propensity is computed for each rule by multiplying the stochastic constant by the number of distinct possible combinations of the elements on the left hand side of the rule.

F. J. Romero-Campero, J. Twycross, M. Camara, M. Bennett, M. Gheorghe, and N. Krasnogor. Modular assembly of cell systems biology models using p systems. International Journal of Foundations of Computer Science, 2009

Multicompartmental Gillespie Algorithm



1

3

$r^3_{1},\ldots,r^3_{n3}$

M_3

$r^1_{1},\ldots,r^1_{n1}$

M_1

2

$r^2_{1},\ldots,r^2_{n2}$

M_2'

Local Gillespie

$(1, \tau_1, r^0_1)$

$(2, \tau_2, r^0_2)$

$(3, \tau_3, r^0_3)$

Sort Compartments

$\boldsymbol{\tau_2 < \tau_1 < \tau_3}$

$(2, \tau_2, r^0_2)$

$(1, \tau_1, r^0_1)$

$(3, \tau_3, r^0_3)$

Update Waiting Times

$(1, \tau_1-\tau_2, r^0_1)$

$(3, \tau_3-\tau_2, r^0_3)$

$(2, \tau_2', r^0_2)$

Insert new triplet

$\boldsymbol{\tau_1-\tau_2 < \tau_2' < \tau_3-\tau_2}$

$(1, \tau_1-\tau_2, r^0_1)$

$(2, \tau_2', r^0_2)$

$(3, \tau_3-\tau_2, r^0_3)$

Outline

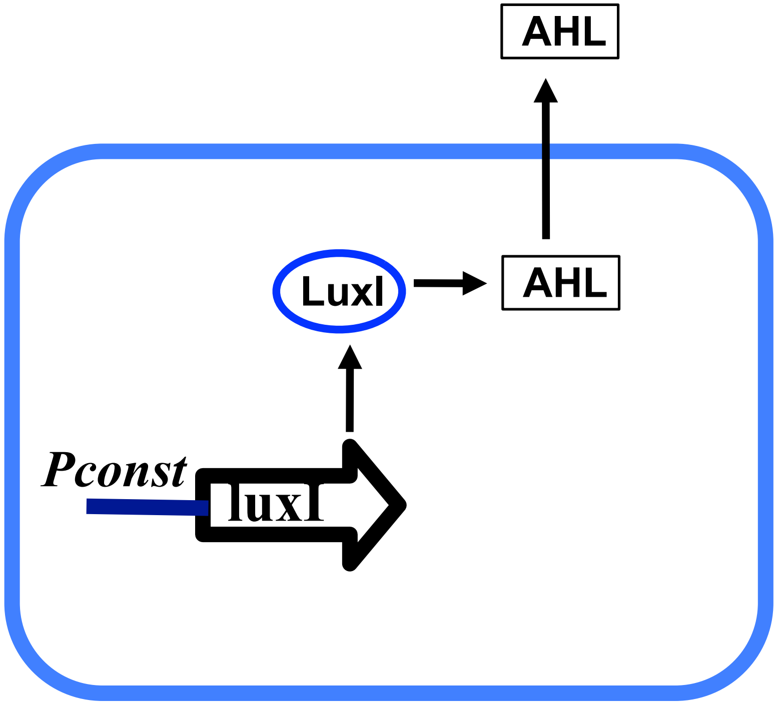
- The cell as a Computing Device for Biocomputing
- P Systems for Representing Cellular Computation
- Examples of Biocomputation
- Conclusions

An example: A Pulse Generator

- **Two different bacterial strains** carrying specific synthetic gene regulatory networks are used.
- The first strain produces a diffusible **signal AHL**.
- The second strain possesses a *synthetic gene regulatory* network which produces a **pulse of GFP** after AHL sensing within a range of values (Band Pass).

S. Basu, R. Mehreja, et al. (2004) Spatiotemporal control of gene expression with pulse generating networks, PNAS, 101, 6355-6360

Sender Cells



SenderCell()=

{

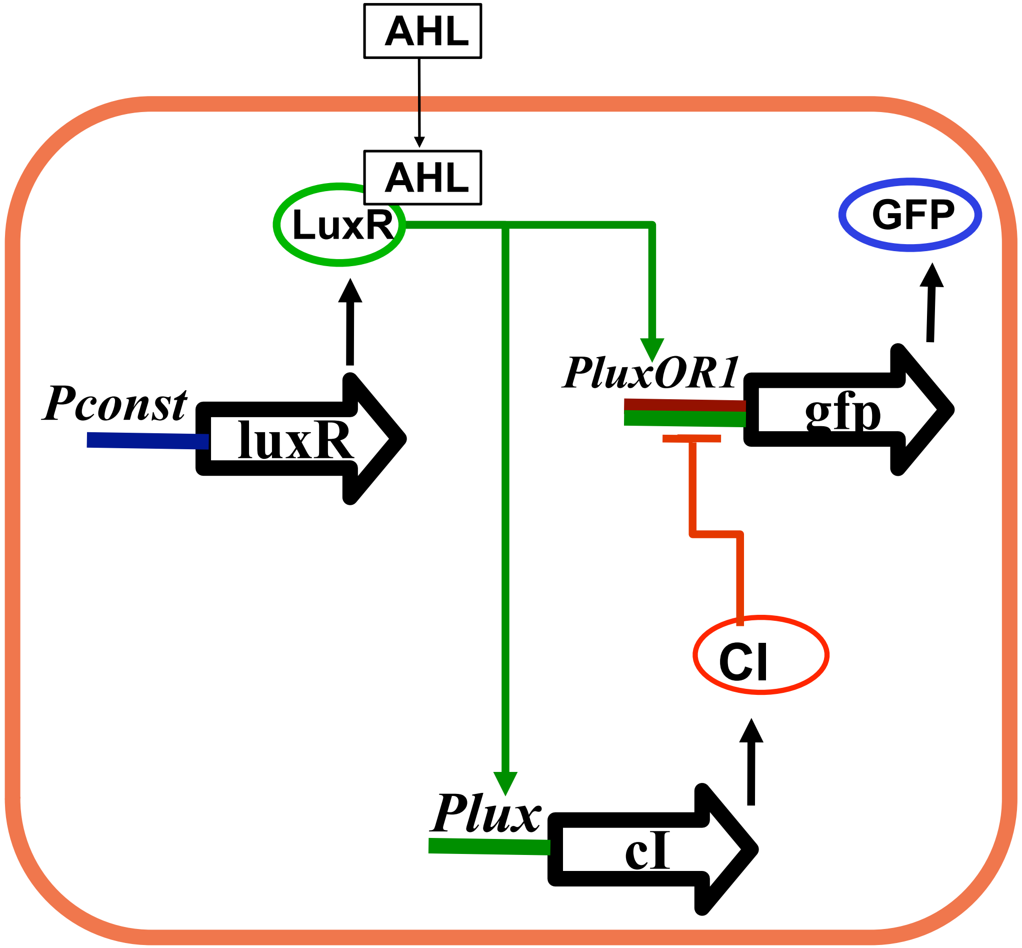
Pconst({X = luxI },…)

PostTransc({X=LuxI},{c_1=3.2,…}

Diff({X=AHL},{c=0.1})

}

Pulse Generating Cells



PulseGenerator()=

{

Pconst({X=luxR},…)

PluxOR1({X=gfp},…)

Plux({X=cI},…)

…

…

Diff({X=AHL},…)

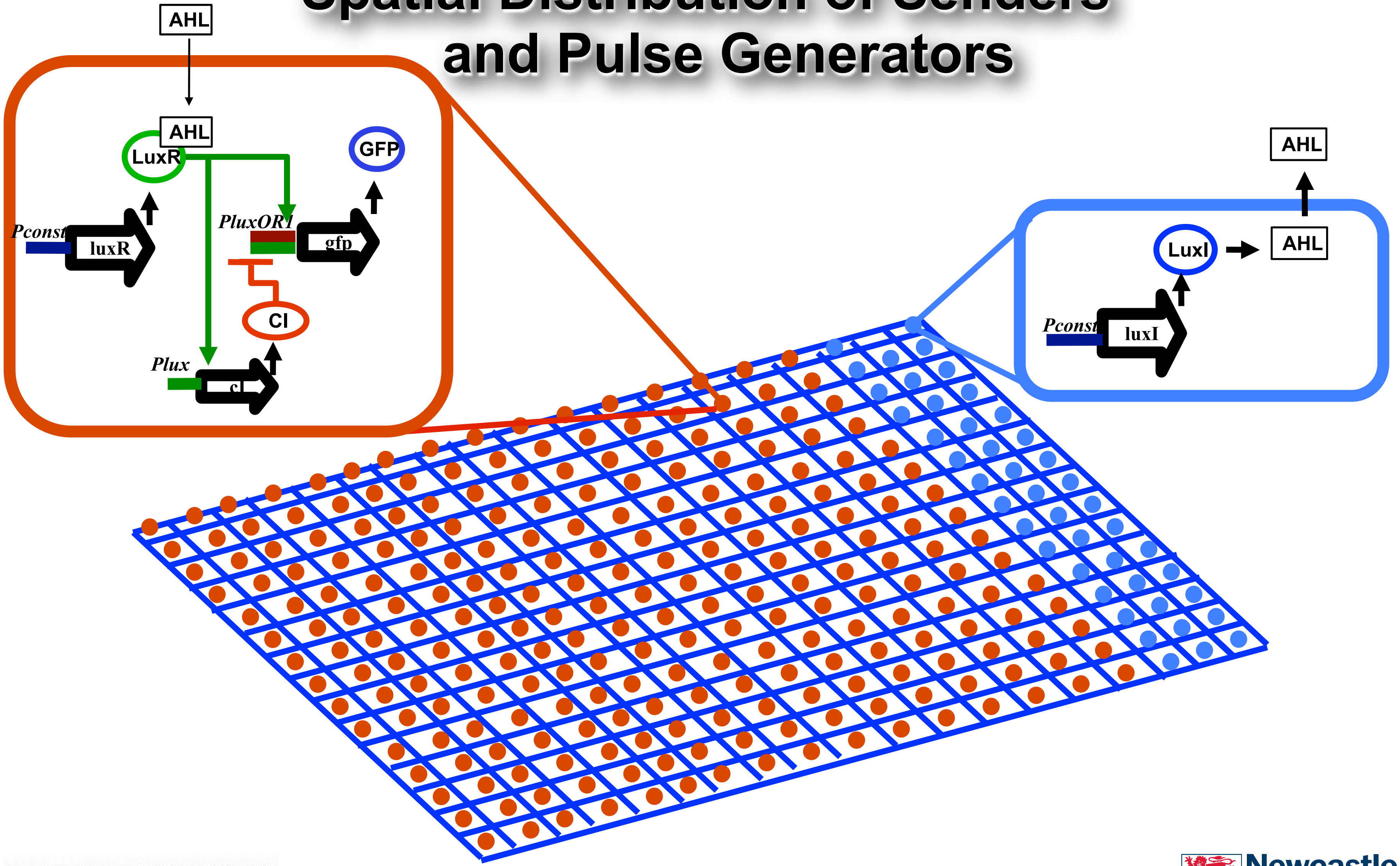
}

Centre For Bacterial Cell Biology

Newcastle University

Centre for Synthetic Biology and the Bioeconomy

Spatial Distribution of Senders and Pulse Generators



Pulse propagation - simulation I

Simulation I

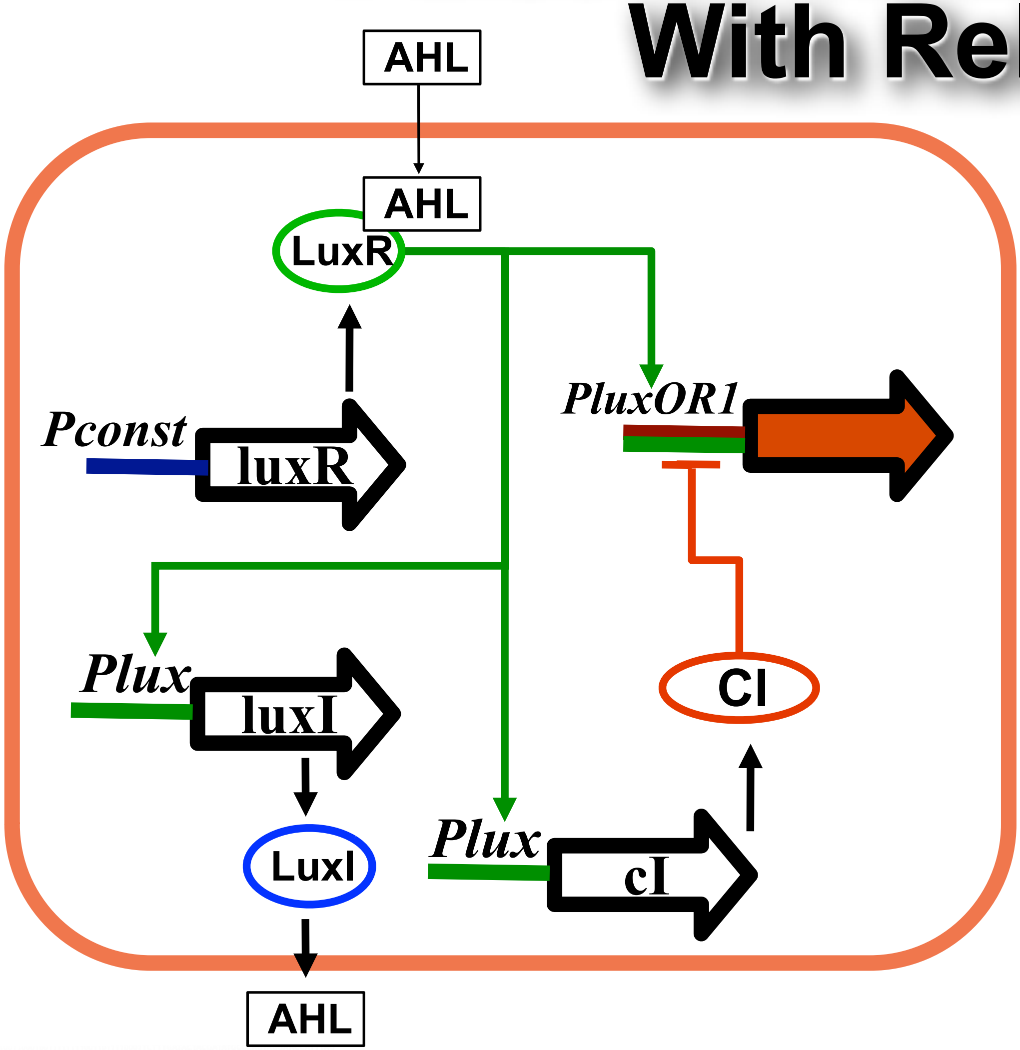
PulseGenerator2.htm

Centre For Bacterial Cell Biology

Newcastle University

Centre for Synthetic Biology and the Bioeconomy

Pulse Generating Cells With Relay



PulseGenerator(**X**) =

{

Pconst({X=luxR},…)

PluxOR1({**X**},…) ,

Plux({X=cI},…) ,

…

Diff({X=AHL},…) ,

Plux({X=luxI},…)

}

Centre For Bacterial Cell Biology

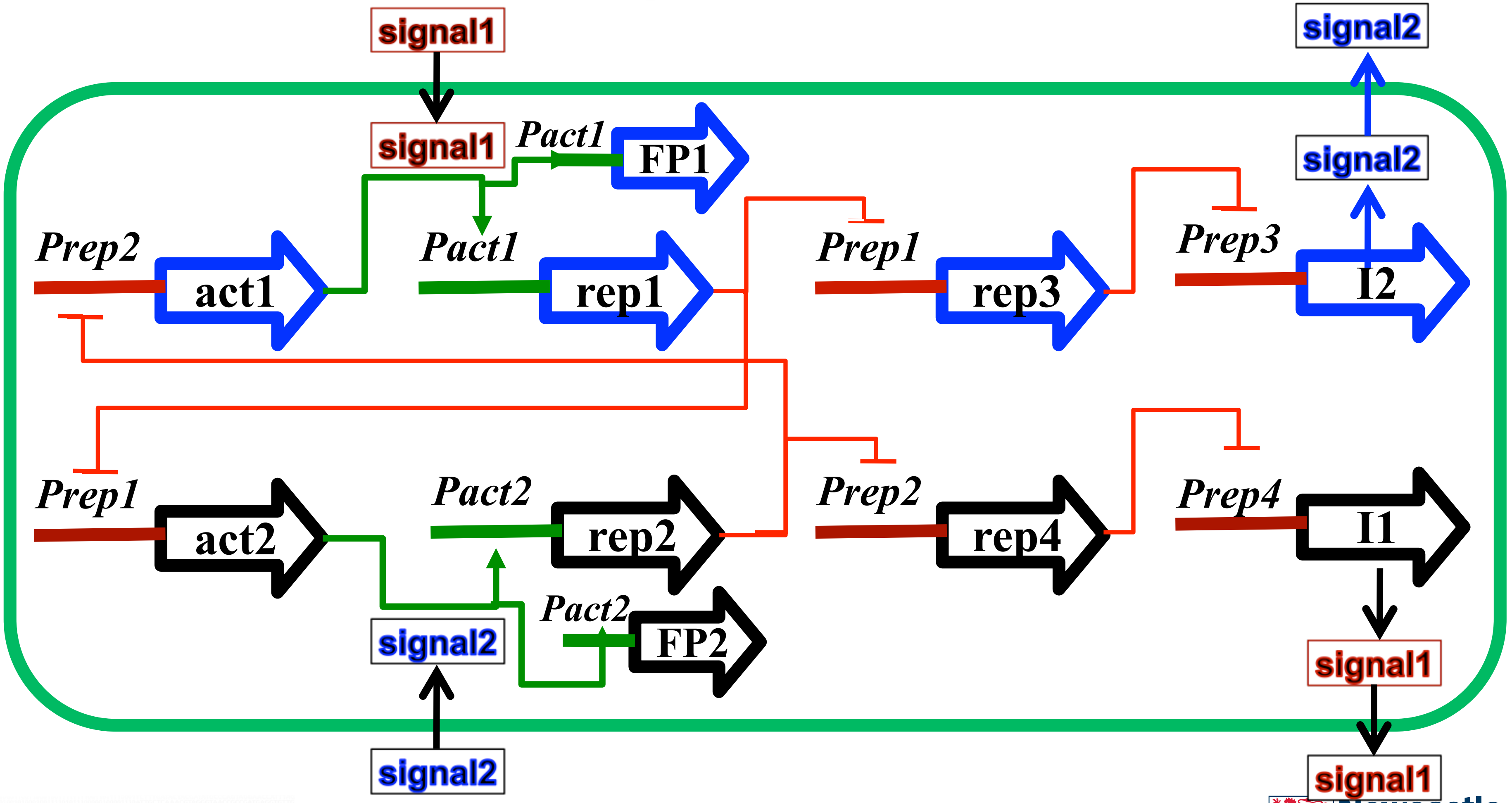
Centre for Synthetic Biology and the Bioeconomy

Pulse propagation & Rely-simulation II

Simulation II

PulseGenerator3.htm

A Signal Translator for Pattern Formation

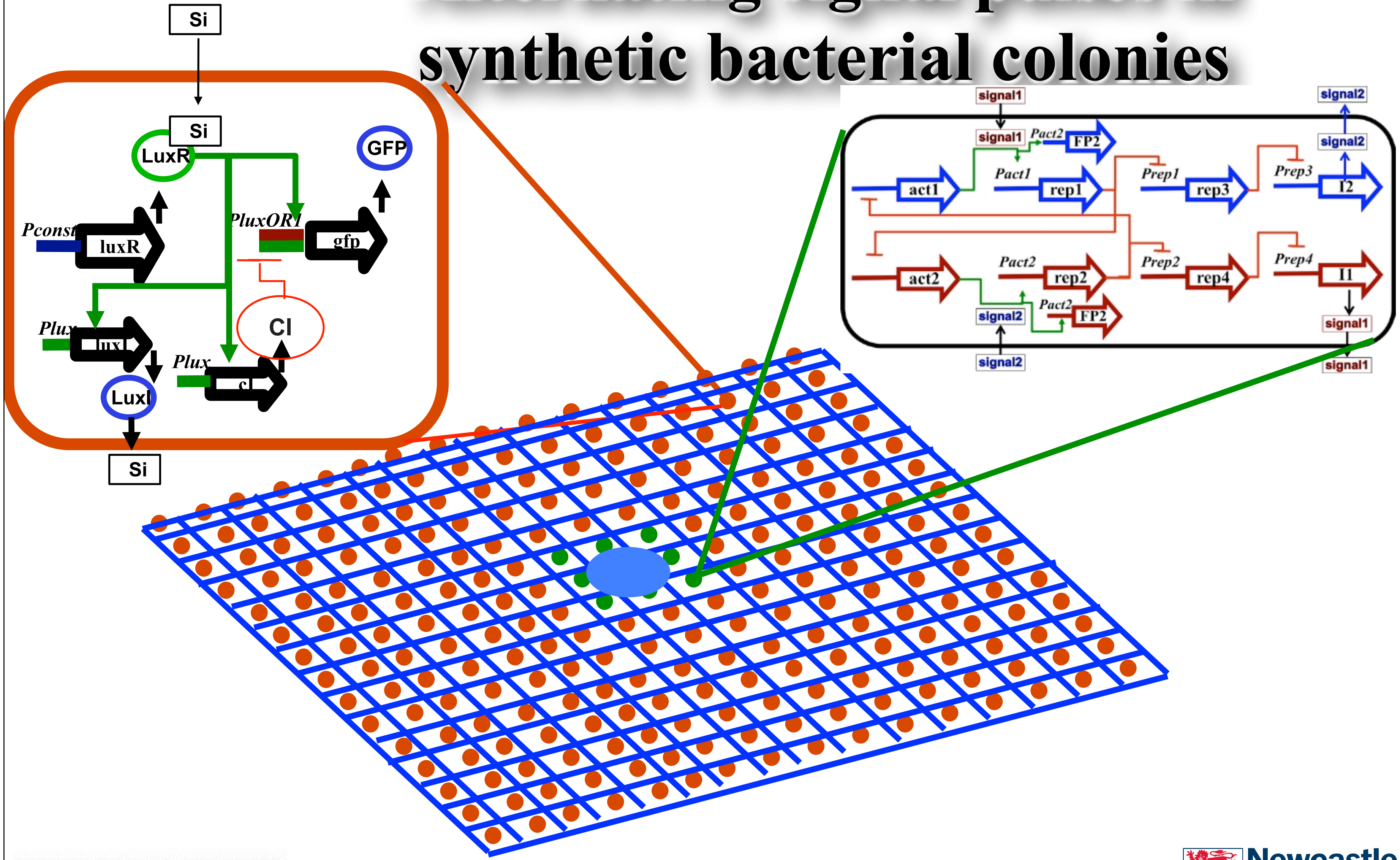


Centre For Bacterial Cell Biology

Newcastle University

Centre for Synthetic Biology and the Bioeconomy

Alternating signal pulses in synthetic bacterial colonies



Centre For Bacterial Cell Biology

Newcastle University

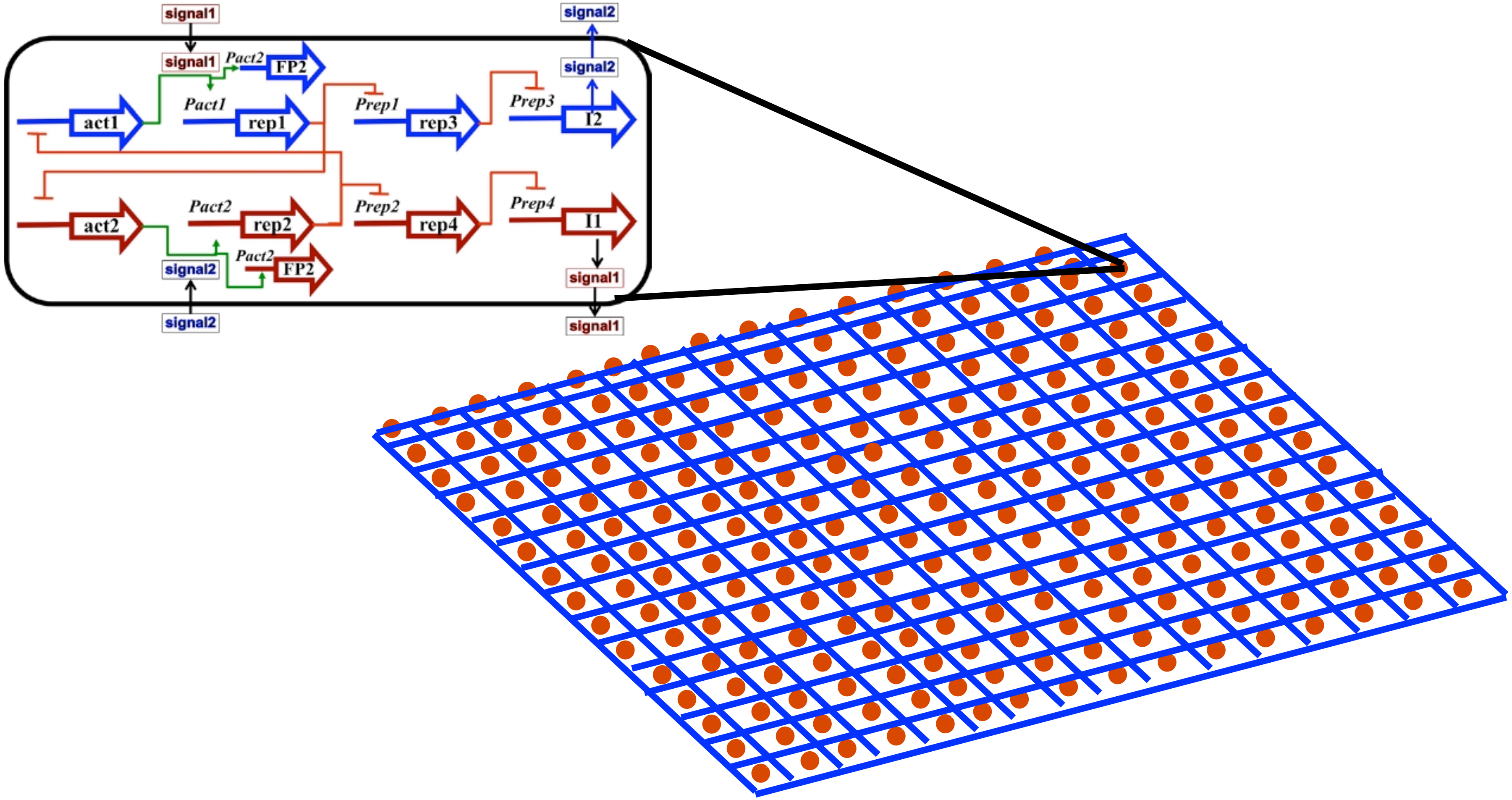
Centre for Synthetic Biology and the Bioeconomy

Alternating signal pulses in synthetic bacterial colonies

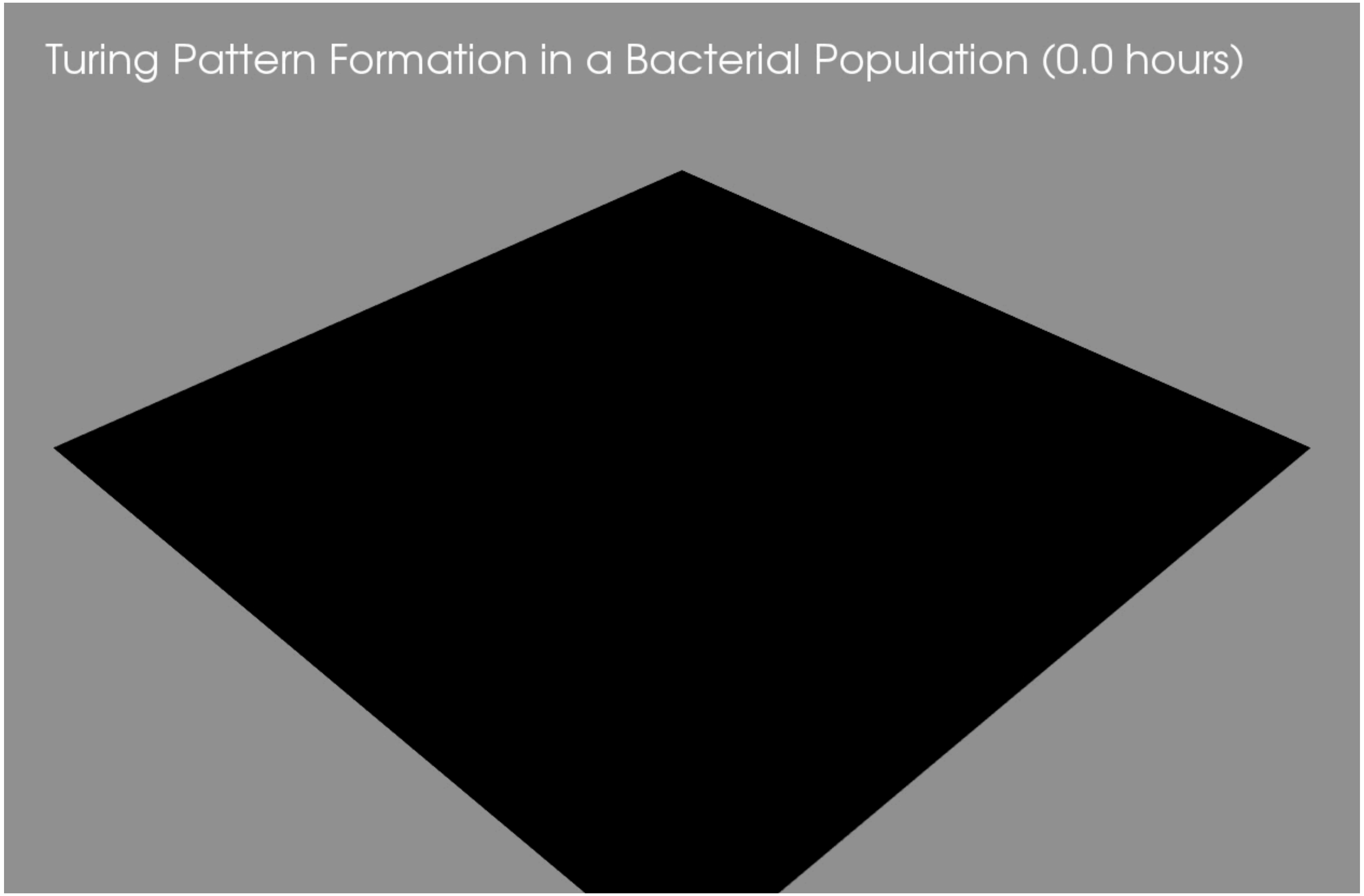
Simulation III

AlternatingPulses.htm

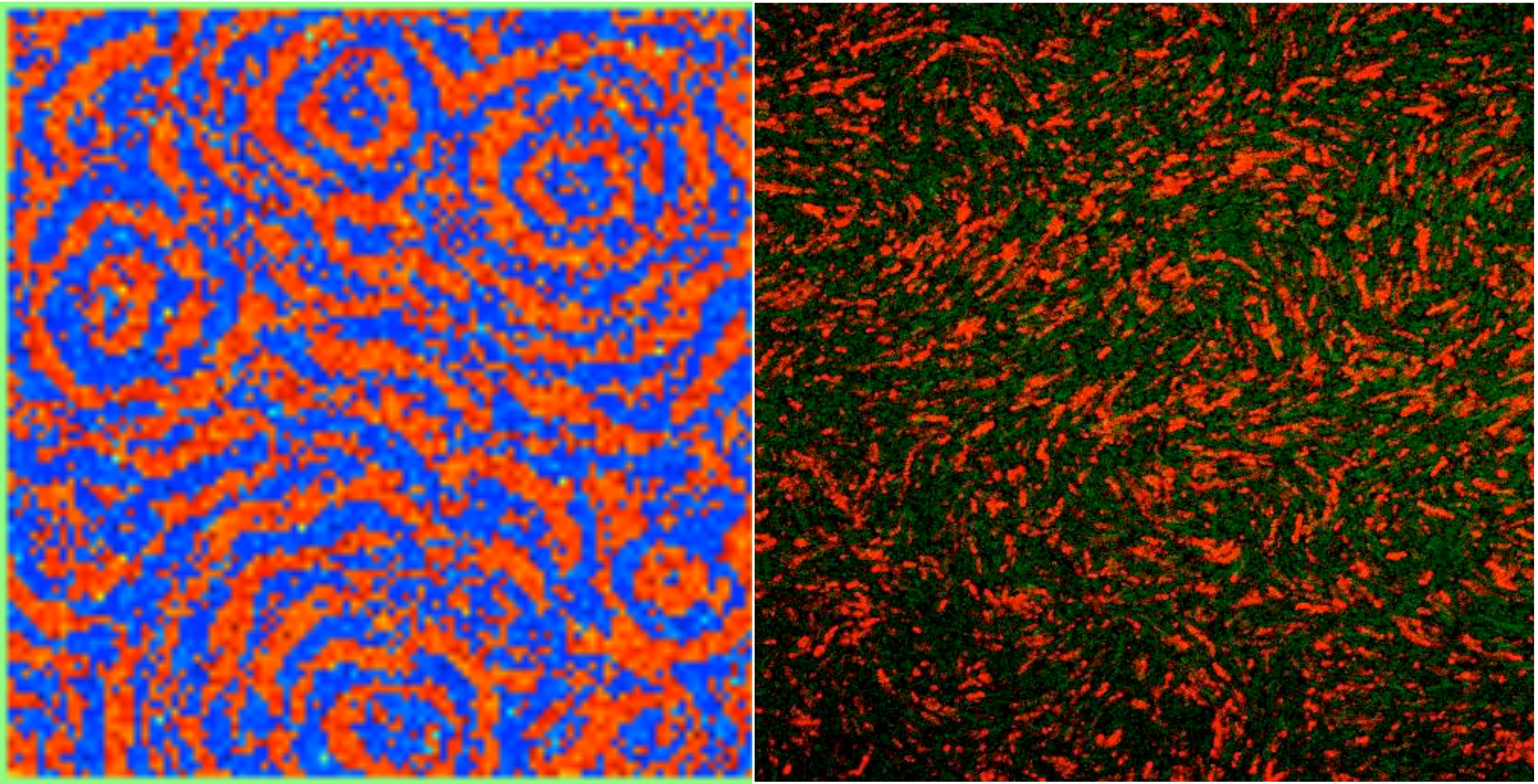
Uniform Spatial Distribution of Signal Translators for Pattern Formation



Turing Patterns in Bacterial Colonies


Turing Pattern Formation in a Bacterial Population (0.0 hours)

In Silico & In Vivo



Outline

- The cell as a Computing Device for Biocomputing
- P Systems for Representing Cellular Computation
- Examples of Biocomputation
- Conclusions

TAKE HOME MESSAGE

- Living cells as stochastic & asynchronous bio-processors that adapt and generate their own hardware on-demand
- Information processing is organised via interconnected networks (genes, signaling, metabolic, etc)
- P systems are a handy way of specifying *discrete and stochastic rule-based compartmental models for cellular computation*.
- Modularity in P systems as a design principle for synthetic networks that enables *reusability, hierarchical abstraction and standardisation*.
- Automated explorations (evolutionary search) on models' structure and parameters.
- Computer Aided analysis of *modular and alternative designs (e.g. synthetic network functionality)*.

Centre For Bacterial Cell Biology

Newcastle University

Centre for Synthetic Biology and the Bioeconomy

Other Sources

F. J. Romero-Campero, J. Twycross, M. Camara, M. Bennett, M. Gheorghe, and N. Krasnogor. Modular assembly of cell systems biology models using p systems. International Journal of Foundations of Computer Science, 2009.

F.J. Romero-Camero and N. Krasnogor. An approach to biomodel engineering based on p systems. In Proceedings of Computation In Europe (CIE 2009), 2009.

J. Smaldon, N. Krasnogor, M. Gheorghe, and A. Cameron. Liposome logic. In Proceedings of the 2009 Genetic and Evolutionary Computation Conference (GECCO 2009), 2009

F. Romero-Campero, H.Cao, M. Camara, and N. Krasnogor. Structure and parameter estimation for cell systems biology models. In Maarten Keijzer et.al, editor, Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2008), pages 331-338. ACM Publisher, 2008. This paper won the Best Paper award at the Bioinformatics track.

J. Smaldon, J. Blake, D. Lancet, and N. Krasnogor. A multi-scaled approach to artificial life simulation with p systems and dissipative particle dynamics. In Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2008). ACM Publisher, 2008.

Other Sources

Păun, Gh. Computing with membranes. Journal of Computer and System Sciences 61 (2000) 108-143

P Systems Web Page http://psystems.disco.unimib.it/

Bianco L. Membrane Models of Biological Systems PhD thesis 2007

Bernardini F, Gheorghe M, Krasnogor N, Terrazas G. Membrane Computing - Current Results and Future Problems. CiE 2005 49-53

Bernardini F, Gheorghe M, Krasnogor N. Quorum sensing P systems. Theoretical Computer Science 371 (2007) 20-33

Miguel Nicolau, Marc Schoenauer. Evolving Specific Network Statistical Properties using a Gene Regulatory Network Model. In ECCS 2008, 5th European Conference on Complex Systems, 2008.

Miguel Nicolau, Marc Schoenauer. Evolving Scale-Free Topologies using a Gene Regulatory Network Model. CEC 2008, IEEE Congress on Evolutionary Computation, pp. 3748-3755, IEEE Press, 2008.

Other Sources

A. Ridwan. A parallel implementation of Gillespie's Direct Method. Proc. of the International Conference on Computational Science, p.284-291, Krakow, Poland, June 2004.

G. C. Ewing et al. Akaroa2: Exploiting network computing by distributing stochastic simulation. Proc. of the European Simulation Multiconference, p. 175-181, Warsaw, June 1999.

P. Hellekalek. Don't trust parallel Monte Carlo! ACM SIGSIM Simulation Digest, 28(1):82-89, 1998.

M. Schwehm. Parallel stochastic simulation of whole cell models. Proc. of the Second International Conference on Systems Biology, p.333-341, CalTech, C.A., November 2001

Any Questions?