The Continuous π-Calculus: A Process Algebra for Biochemical Modelling

Ian Stark and Marek Kwiatkowski

Laboratory for Foundations of Computer Science
School of Informatics
The University of Edinburgh

Friday 20 March 2009

Overview

The *continuous π-calculus* ($c\pi$) is a process algebra for modelling behaviour and variation in molecular systems.

It has a structured operational semantics that captures system behaviour as trajectories through a continuous process space, by generating familiar differential-equation models.

We have existing biochemical systems expressed in $c\pi$; the aim is to use this to investigate evolutionary properties of biochemical pathways.

Marek Kwiatkowski and Ian Stark.

The Continuous π-Calculus: A Process Algebra for Biochemical Modelling. In *Computational Methods in Systems Biology: Proc. CMSB 2008*

Lecture Notes in Computer Science 5307, pages 103–122. Springer 2008

Overview

Contents

- Systems Biology and Process Algebras
- The Continuous π-Calculus
- Example: Circadian Rhythms in *Synechococcus Elongatus*

Systems Biology

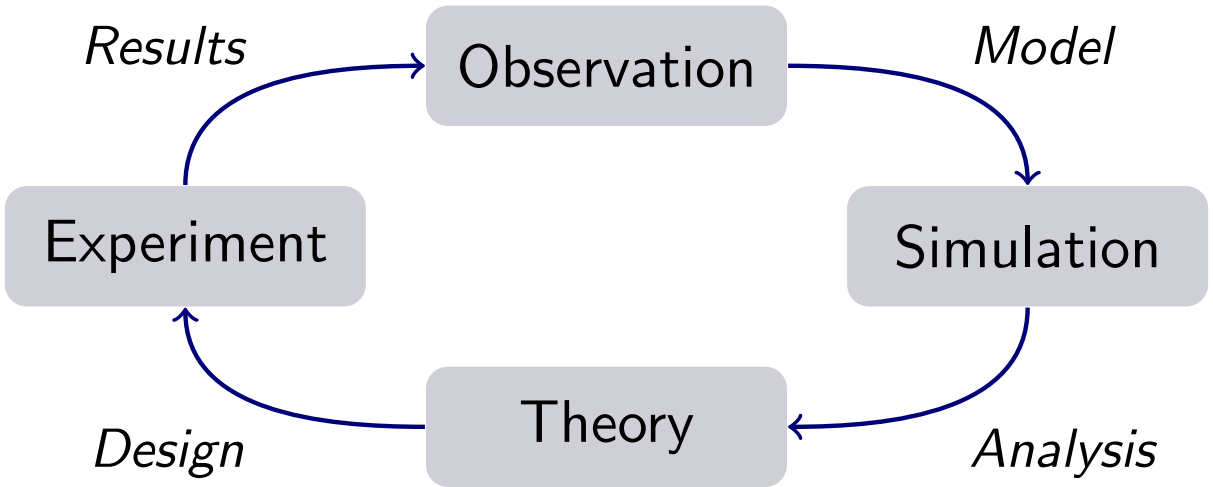
Biology is the study of living organisms; Systems Biology is the study of the dynamic processes that take place within those organisms.

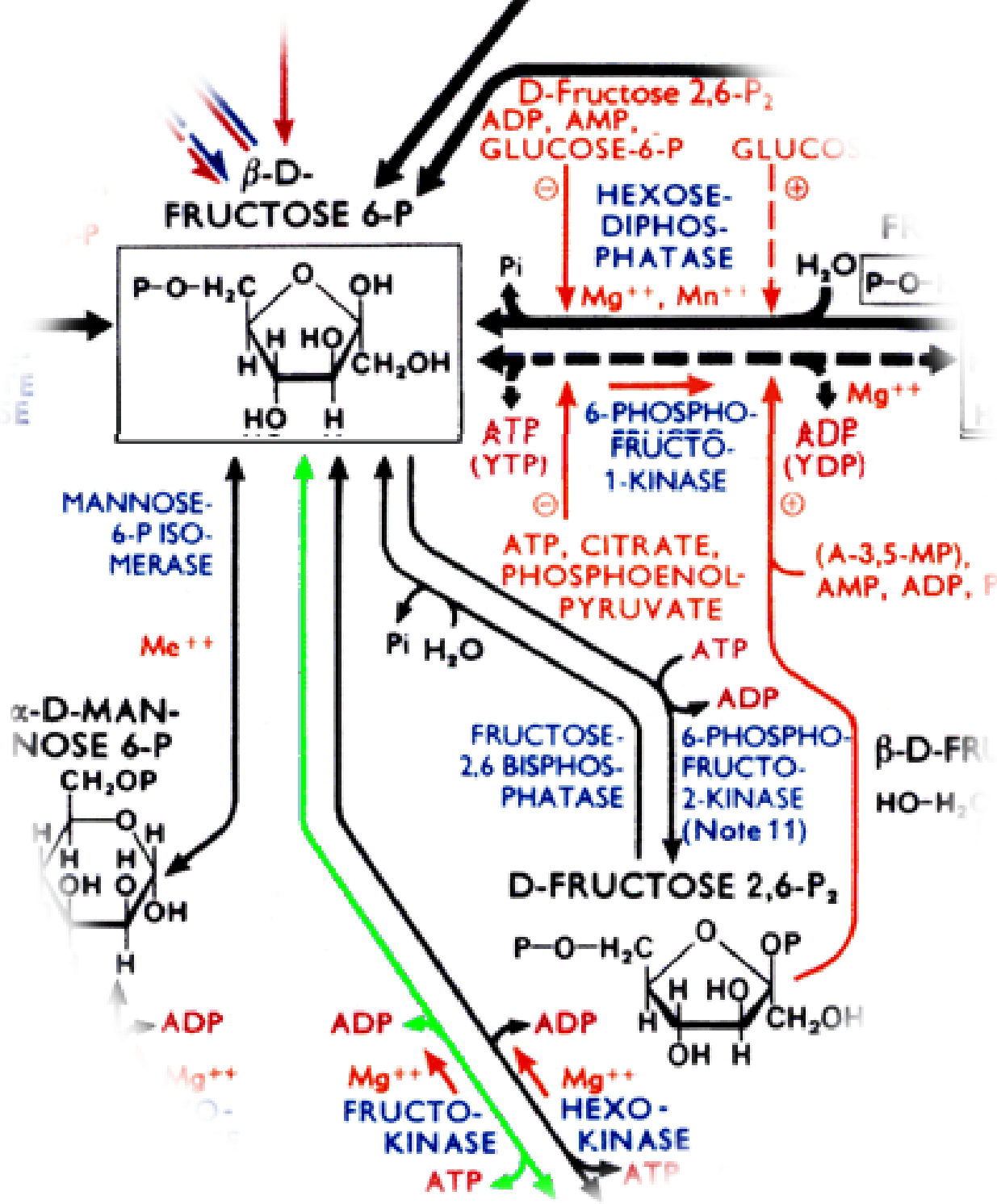
In particular:

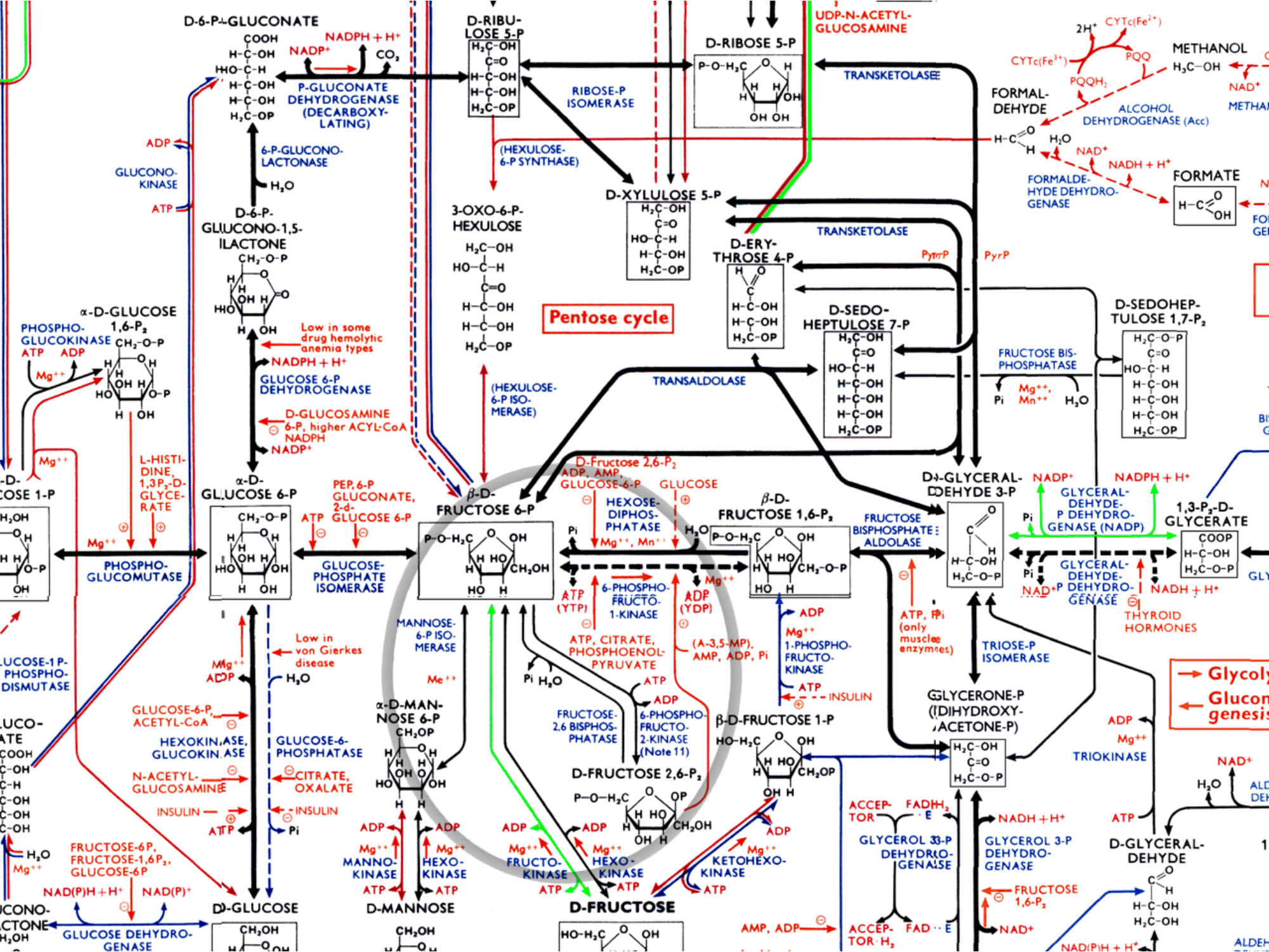
- Interaction between processes;
- Behaviour emerging from such interaction; and
- Integration of component behaviours.

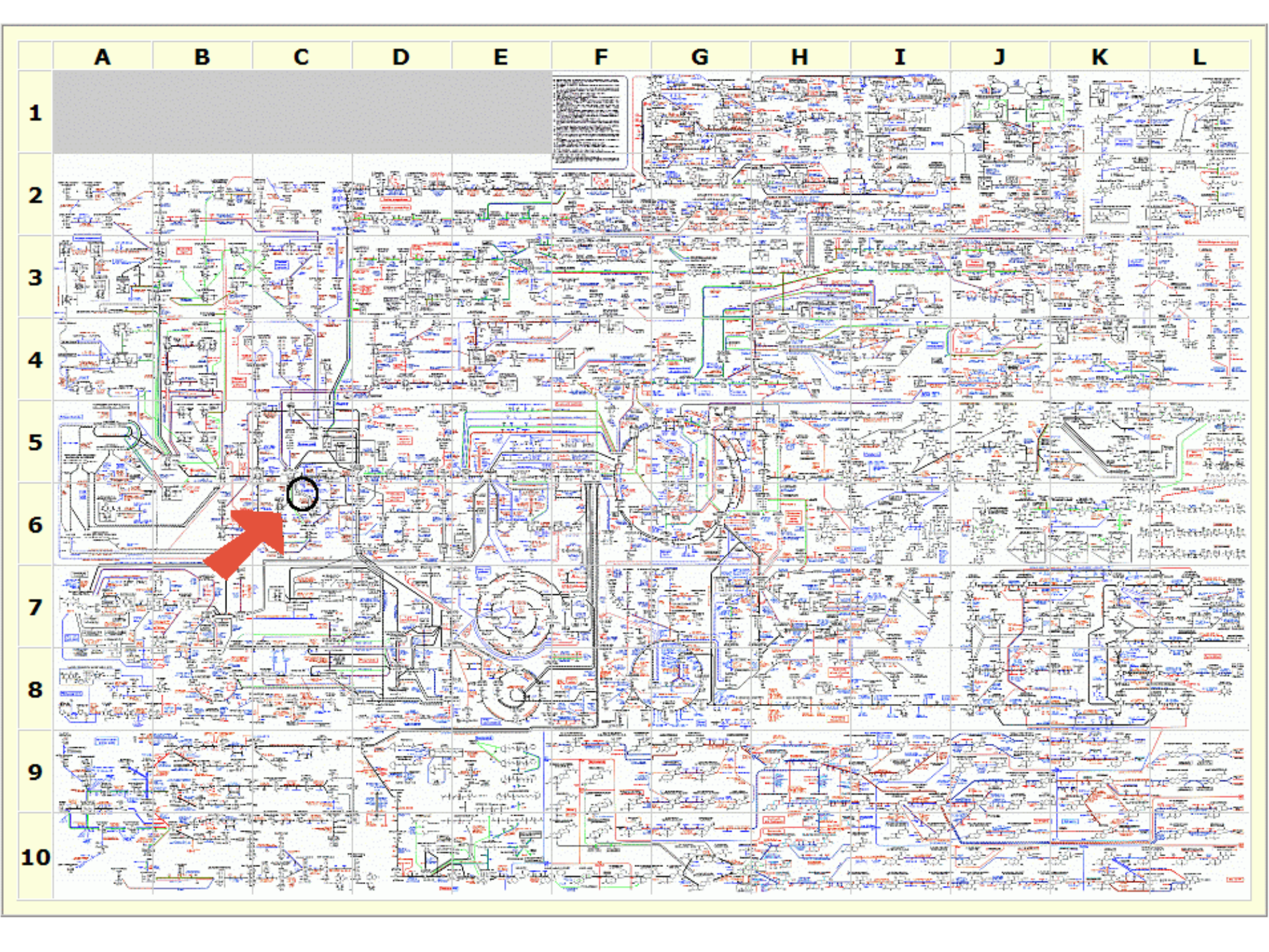
Systems Biology

Biology is the study of living organisms; Systems Biology is the study of the dynamic processes that take place within those organisms.




β-D-
FRUCTOSE 6-P
D-Fructose 2,6-P2
ADP, AMP,
GLUCOSE-6-P
HEXOSE-
DIPHOS-
PHATASE
Mg++, Mn++
Pi
H2O
ATP
(YTP)
6-PHOSPHO-
FRUCTO-
1-KINASE
ADP
(YDP)
Mg++
ATP, CITRATE,
PHOSPHOENOL-
PYRUVATE
(A-3,5-MP),
AMP, ADP,
MANNOSE-
6-P ISO-
MERASE
Me++
α-D-MAN-
NOSE 6-P
Pi H2O
ATP
ADP
FRUCTOSE-
2,6 BISPHOS-
PHATASE
6-PHOSPHO-
FRUCTO-
2-KINASE
(Note 11)
D-FRUCTOSE 2,6-P2
ADP
ADP
ADP
Mg++
FRUCTO-
KINASE
Mg++
HEXO-
KINASE
ATP
ATP


Pentose cycle
D-6-P-GLUCONATE
NADP+
NADPH + H+
CO2
P-GLUCONATE DEHYDROGENASE (DECARBOXY-LATING)
D-RIBU-LOSE 5-P
RIBOSE-P ISOMERASE
D-RIBOSE 5-P
TRANSKETOLASE
UDP-N-ACETYL-GLUCOSAMINE
6-P-GLUCONO-LACTONASE
H2O
ADP
ATP
GLUCONO-KINASE
D-6-P-GLUCONO-1,5-LACTONE
(HEXULOSE-6-P SYNTHASE)
3-OXO-6-P-HEXULOSE
D-XYLULOSE 5-P
TRANSKETOLASE
D-ERY-THROSE 4-P
PyrP
D-SEDO-HEPTULOSE 7-P
TRANSALDOLASE
(HEXULOSE-6-P ISO-MERASE)
FRUCTOSE BIS-PHOSPHATASE
Mg++, Mn++
Pi
D-SEDOHEP-TULOSE 1,7-P2
α-D-GLUCOSE 1,6-P2
PHOSPHO-GLUCOKINASE
α-D-GLUCOSE 6-P
Low in some drug hemolytic anemia types
NADPH + H+
GLUCOSE 6-P DEHYDROGENASE
D-GLUCOSAMINE 6-P, higher ACYL-CoA
NADPH
NADP+
L-HISTI-DINE, 1,3P2-D-GLYCE-RATE
PEP, 6-P GLUCONATE, 2-d-GLUCOSE 6-P
PHOSPHO-GLUCOMUTASE
GLUCOSE-PHOSPHATE ISOMERASE
β-D-FRUCTOSE 6-P
D-Fructose 2,6-P2
ADP, AMP, GLUCOSE-6-P
GLUCOSE
HEXOSE-DIPHOS-PHATASE
β-D-FRUCTOSE 1,6-P2
FRUCTOSE BISPHOSPHATE ALDOLASE
D-GLYCERAL-DEHYDE 3-P
GLYCERAL-DEHYDE-P DEHYDRO-GENASE (NADP)
1,3-P2-D-GLYCERATE
GLYCERAL-DEHYDE-P DEHYDRO-GENASE
THYROID HORMONES
ATP, Pi (only muscle enzymes)
6-PHOSPHO-FRUCTO-1-KINASE
ATP (YTP)
ADP (YDP)
ATP, CITRATE, PHOSPHOENOL-PYRUVATE
(A-3,5-MP), AMP, ADP, Pi
1-PHOSPHO-FRUCTO-KINASE
INSULIN
TRIOSE-P ISOMERASE
GLYCERONE-P (DIHYDROXY-ACETONE-P)
MANNOSE-6-P ISO-MERASE
α-D-MAN-NOSE 6-P
FRUCTOSE-2,6 BISPHOS-PHATASE
6-PHOSPHO-FRUCTO-2-KINASE (Note 11)
D-FRUCTOSE 2,6-P2
β-D-FRUCTOSE 1-P
TRIOKINASE
Low in von Gierkes disease
GLUCOSE-6-P, ACETYL-CoA
HEXOKINASE, GLUCOKINASE
GLUCOSE-6-PHOSPHATASE
N-ACETYL-GLUCOSAMINE
CITRATE, OXALATE
INSULIN
FRUCTOSE-6P, FRUCTOSE-1,6P2, GLUCOSE-6P
NAD(P)H + H+
NAD(P)+
D-GLUCOSE
GLUCOSE DEHYDRO-GENASE
D-MANNOSE
MANNO-KINASE
HEXO-KINASE
FRUCTO-KINASE
D-FRUCTOSE
KETOHEXO-KINASE
AMP, ADP
ACCEP-TOR H2
FAD
FADH2
GLYCEROL 3-P DEHYDRO-GENASE
FRUCTOSE 1,6-P2
D-GLYCERAL-DEHYDE
METHANOL
FORMAL-DEHYDE
ALCOHOL DEHYDROGENASE (Acc)
FORMALDE-HYDE DEHYDRO-GENASE
FORMATE
CYTc(Fe3+)
CYTc(Fe2+)
PQQ
PQQH2
2H+
Glycol
Glucon genesis


A
B
C
D
E
F
G
H
I
J
K
L
1
2
3
4
5
6
7
8
9
10

What can Computer Science do for Systems Biology?

Machines

Large Databases: Semistructured data; data integration; data mining

Large Simulations: Experiments *in silico*; parameter scans; folding search

Ideas

Language: Abstraction; modularity; semantics; formal models

Reasoning: Logics; behavioural description; model checking

Biochemical Simulation

Biologists routinely use one of two alternative approaches to computational modelling of biochemical systems:

- Stochastic simulation
  - Continuous time
  - Discrete behaviour: tracking individual molecules
  - Randomized
  - Gillespie's algorithm
- Ordinary Differential Equations
  - Continuous time
  - Continuous behaviour: chemical concentrations
  - Deterministic
  - Numerical ODE solutions

The classical approach is to use the mathematics directly as the target formal system; CS suggests the value of a mediating language.

Process Algebras in Systems Biology

- Petri nets
- π-calculus; stochastic π; BioSPI; SPiM
- Beta binders
- Ambients, bioAmbients
- Brane calculi; Bitonal systems
- PEPA, bioPEPA
- Kappa
- PRISM
- Pathway Logic
- ...

The Continuous π-Calculus

The *Continuous π-Calculus* ($c\pi$) is a process algebra for modelling behaviour and variation in molecular systems.

Based on the π-calculus, it introduces continuous variability in:

- rates of reaction;
- affinity between interacting names; and
- quantities of processes.

while retaining classic process-algebra features of:

- compositional semantics (modular, not monolithic);
- abstraction (separating language and semantics);
- specifying interaction (taking behaviour as it emerges).

Motivated by Fontana's work on evolutionary change, neutral spaces and the "topology of the possible".

Basics of $c\pi$

Continuous π has two levels of system description:

- Species
  - Individual molecules (proteins)
  - Transition system semantics
- Processes
  - Bulk population (concentration)
  - Differential equations

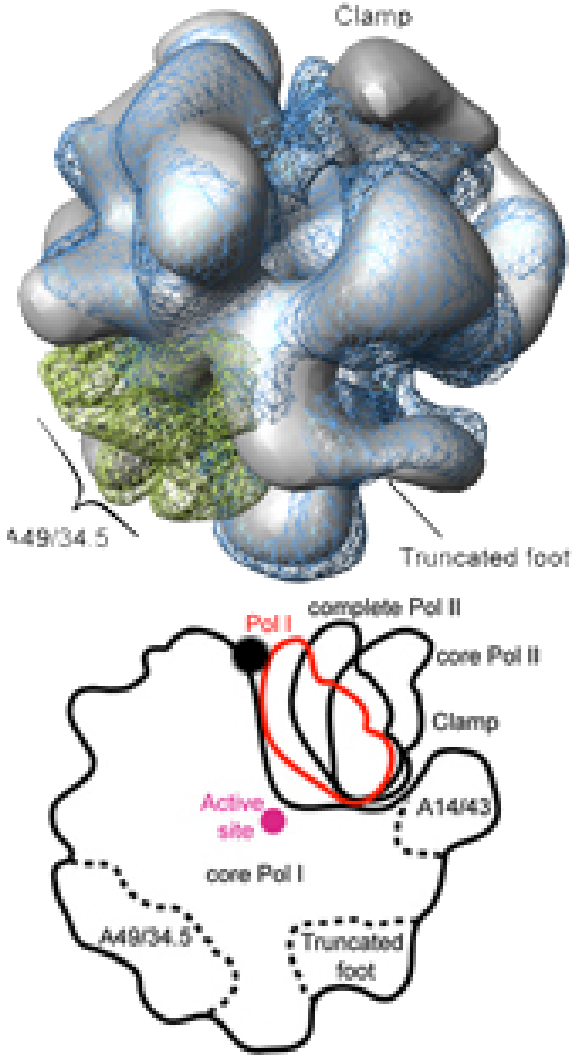
Process space arises as a real-valued vector space over species, with each point the state of a system and behaviours as trajectories through that.

Names in $c\pi$

As in standard π-calculus, *names* indicate a potential for interaction: for example, the docking sites on an enzyme where other molecules may attach.

These sites may interact with many different other sites, to different degrees.

This variation is captured by an *affinity network*: a graph setting out the interaction potential between different names.

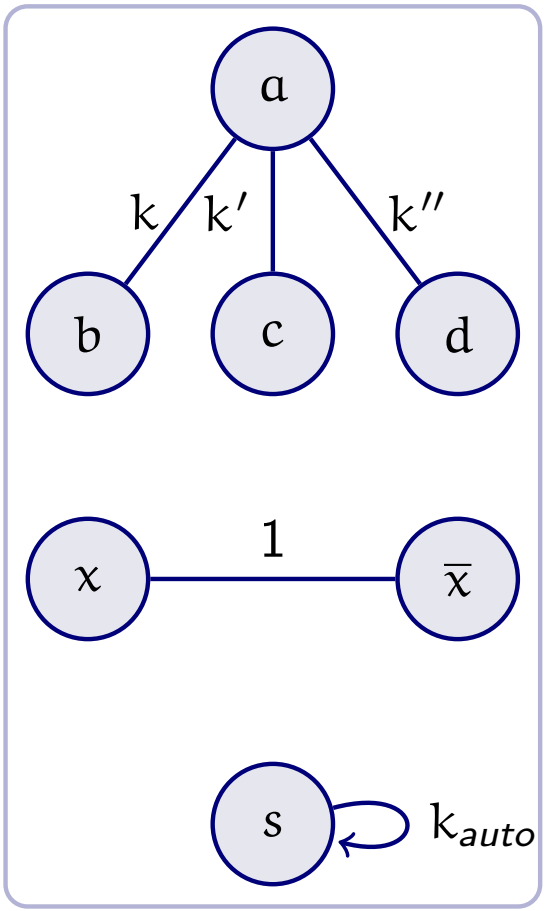


Names in $c\pi$

As in standard π-calculus, *names* indicate a potential for interaction: for example, the docking sites on an enzyme where other molecules may attach.

These sites may interact with many different other sites, to different degrees.

This variation is captured by an *affinity network*: a graph setting out the interaction potential between different names.

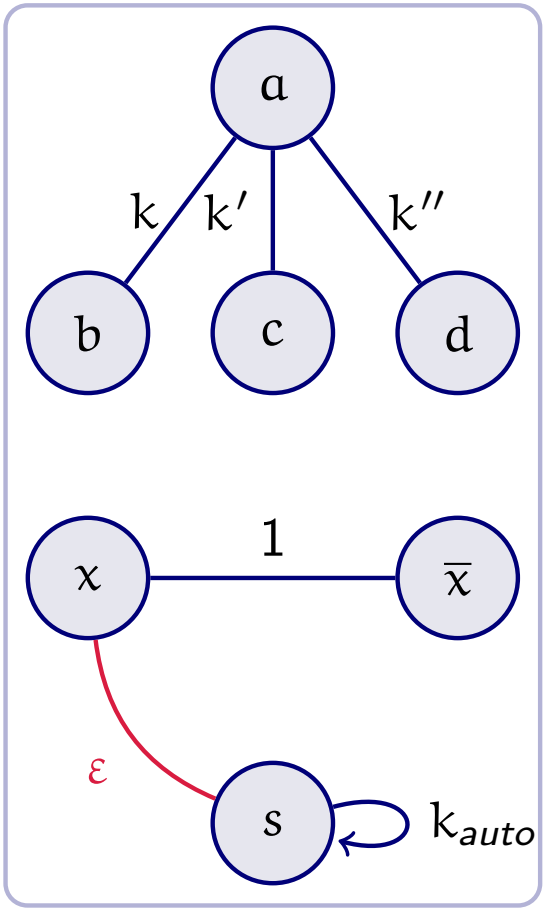


Names in $c\pi$

As in standard π-calculus, *names* indicate a potential for interaction: for example, the docking sites on an enzyme where other molecules may attach.

These sites may interact with many different other sites, to different degrees.

This variation is captured by an *affinity network*: a graph setting out the interaction potential between different names.



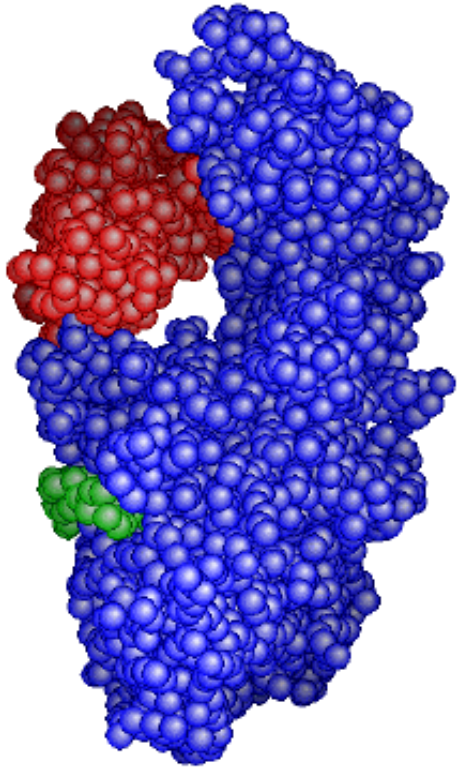
Restriction in $c\pi$

Name restriction $\nu x(A \mid B)$ captures molecular *complexes*, with local name x mediating further internal modification, or decomplexation.

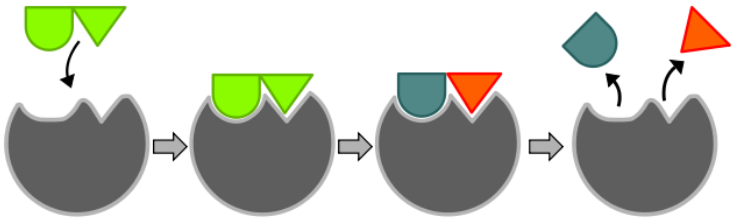
The binder can be a single local name $(\nu x.-)$, or several names with their own affinity network $(\nu M.-)$.

As in the classic π-calculus "cocktail party" model, interacting names can communicate further names, allowing further interactions.

In particular, we use name *extrusion* to model complex formation.



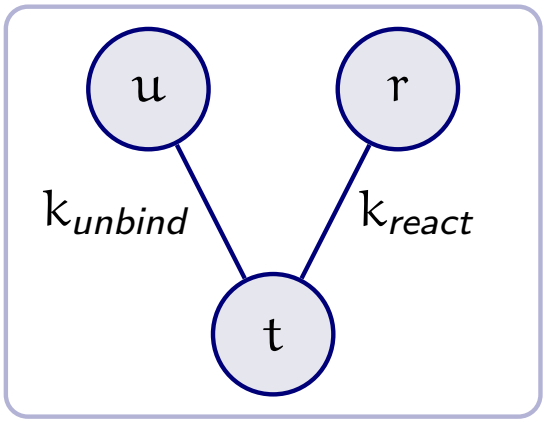
Example Species: Enzyme Catalysis

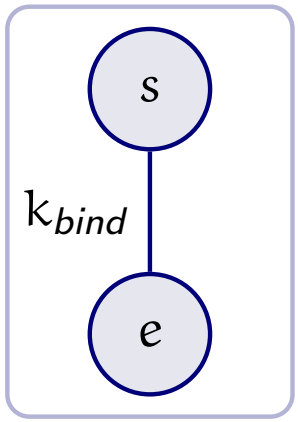


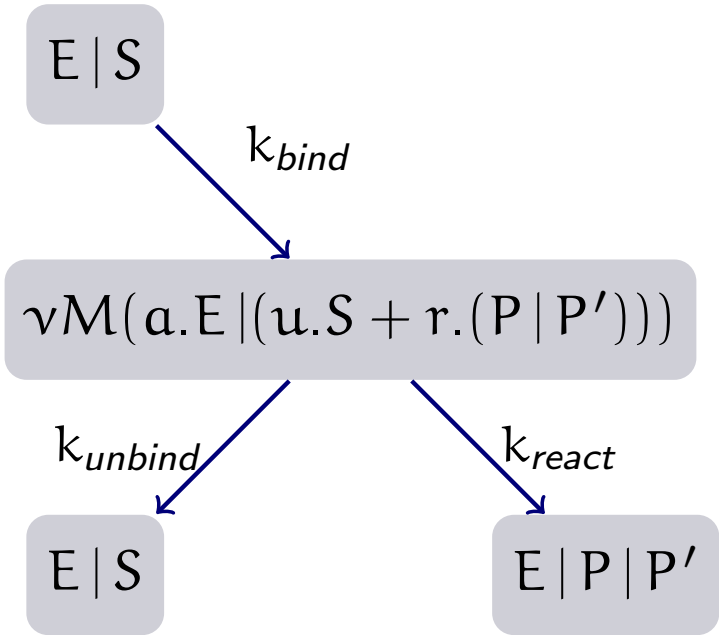
$$S = s(x, y).(x.S + y.(P|P'))$$

$$E = \nu M.e\langle u, r\rangle.t.E$$

$$P = P' = \tau@k_{degrade}.0$$







Species

Species $A, B ::= 0 \mid S(\vec{a}) \mid \Sigma a(\vec{b}; \vec{y}).A \mid \tau@k.A \mid A \mid B \mid \nu M.A$

Symmetric prefix $a(b, c; x, y).A$ for two-way communication.

Guarded sums $\Sigma_i \alpha_i.A$ or $\alpha.A + \alpha'.A'$ for alternative choices.

Silent transition $\tau@k.A$ for constitutive reactions at rate $k \in \mathbb{R}_{\geqslant 0}$.

Parallel composition $A \mid B$ within complexes.

Recursion via guarded species definitions $S(\vec{x}) = \ldots$

Set $\mathcal{S}$ of species up to structural congruence, and $\mathcal{S}^{\#}$ of *prime* species.

Operational Semantics for Species

The behaviour of a species is given by transitions:

$$A \xrightarrow{a} (\vec{b};\vec{y})B \qquad \text{Potential interaction}$$

$$A \xrightarrow{\tau@k} B \qquad \text{Immediate action}$$

$$A \xrightarrow{\tau\langle x,y\rangle} B \qquad \text{Internal action}$$

Here $(\vec{b};\vec{y})B$ is a *concretion* representing potential interaction; the result of actual interaction is given by pseudo application:

$$(\vec{a};\vec{x})A \circ (\vec{b};\vec{y})B \;=\; A\{\vec{b}/\vec{x}\} \mid B\{\vec{a}/\vec{y}\}$$

Rules for deriving transitions give a structural operational semantics:

$$\frac{A \xrightarrow{a} F \quad B \xrightarrow{b} G}{A \mid B \xrightarrow{\tau\langle a,b\rangle} F \circ G} \qquad \frac{A \xrightarrow{\tau\langle a,b\rangle} B \quad a,b \in M}{\nu M.A \xrightarrow{\tau@M(a,b)} B} \qquad \ldots$$

Processes

Processes	$P, Q ::= 0 \mid c \cdot A \mid P \parallel Q$
Component	species $c \cdot A$ at concentration $c \in \mathbb{R}_{\geqslant 0}$.
Mixture	of processes $P \parallel Q$.

We can identify processes, up to structural congruence, with elements of *process space* $\mathcal{P} = \mathbb{R}^{\mathcal{S}^\#}$.

Species embed in process space $\langle - \rangle : \mathcal{S} \to \mathcal{P}$ at unit concentration.

Operational Semantics for Processes

Immediate behaviour $\frac{dP}{dt} \in \mathbb{R}^{\mathcal{S}^\#}$ vector in process space

Interaction potential $\partial P \in \mathbb{R}^{\mathcal{S}\times\mathcal{N}\times\mathcal{C}} = \mathcal{D}$ interaction space

Space $\mathcal{D}$ has basis $\langle A \xrightarrow{a} F \rangle$ for species A, name a, concretion F.

Interaction tensor $\owedge : \mathcal{D} \times \mathcal{D} \to \mathcal{P}$

Bilinear function generated by

$$\langle A \xrightarrow{a} F \rangle \owedge \langle B \xrightarrow{b} G \rangle = Aff(a, b)(\langle F \circ G \rangle - \langle A \rangle - \langle B \rangle)$$

Process Semantics

$\frac{dP}{dt}$: Immediate behaviour

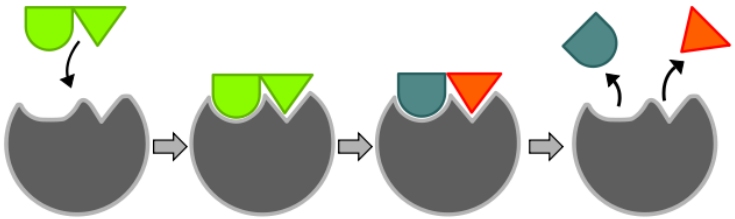
- Vector field $\frac{d}{dt}$ over process space $\mathcal{P}$
- Equivalent to an ODE system

∂P: Interaction potential

- Element of $\mathbb{R}^{\mathcal{S}\times\mathcal{N}\times\mathcal{C}}$
- Equivalent to transition system

$$\partial(P \parallel Q) = \partial P + \partial Q$$
$$\frac{d(P \parallel Q)}{dt} = \frac{dP}{dt} + \frac{dQ}{dt} + \partial P \oplus \partial Q$$

Example Process: Enzyme Catalysis



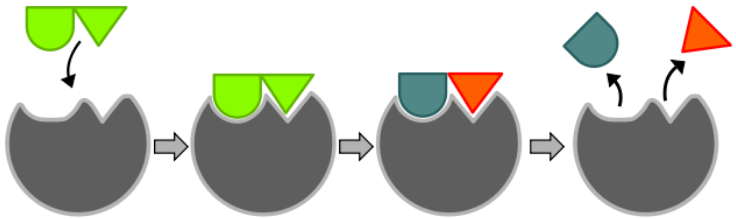
$$S = s(x, y).(x.S + y.(P|P'))$$

$$E = \nu M.e\langle u, r\rangle.t.E$$

$$P = P' = \tau @ k_{degrade}.0$$

$$c_S \cdot S \parallel c_E \cdot E$$

Example Process: Enzyme Catalysis



$$S = s(x, y).(x.S + y.(P|P'))$$

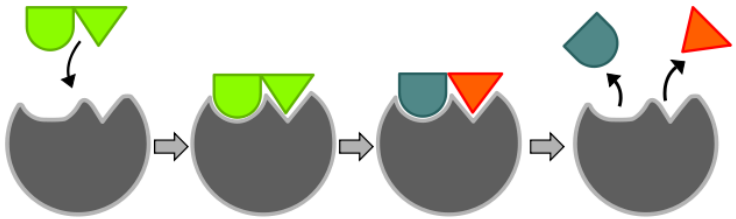
$$E = \nu M.e\langle u, r\rangle.t.E$$

$$P = P' = \tau@k_{degrade}.0$$

$$c_S \cdot S \,\|\, c_E \cdot E$$

```
enzyme.cpi
...
species E() = {
  site t, u, r;
...
```

Example Process: Enzyme Catalysis



$$S = s(x, y).(x.S + y.(P|P'))$$

$$E = \nu M.e\langle u, r\rangle.t.E$$

$$P = P' = \tau@k_{degrade}.0$$

$$c_S \cdot S \,\|\, c_E \cdot E$$

Cpi tool

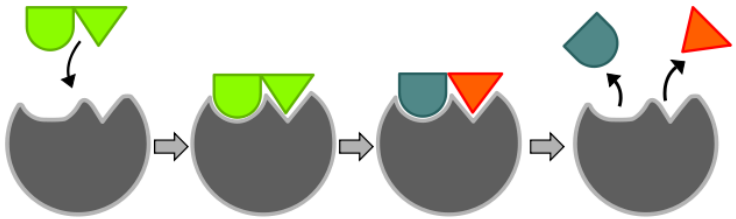
```
enzyme.cpi
...
species E() = {
  site t, u, r;
...
```

ODEs

$$x_2' = -k_1 x_4 x_2 + \ldots$$

$$\vdots$$

Example Process: Enzyme Catalysis



$$S = s(x, y).(x.S + y.(P|P'))$$

$$E = \nu M.e\langle u, r\rangle.t.E$$

$$P = P' = \tau@k_{degrade}.0$$

$$c_S \cdot S \,\|\, c_E \cdot E$$

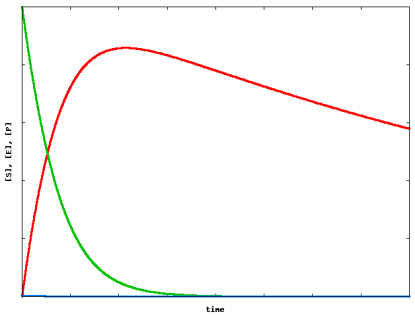
Cpi tool *Octave*

```
enzyme.cpi
...
species E() = {
  site t, u, r;
...
```

ODEs

$$x_2' = -k_1 x_4 x_2 + \ldots$$

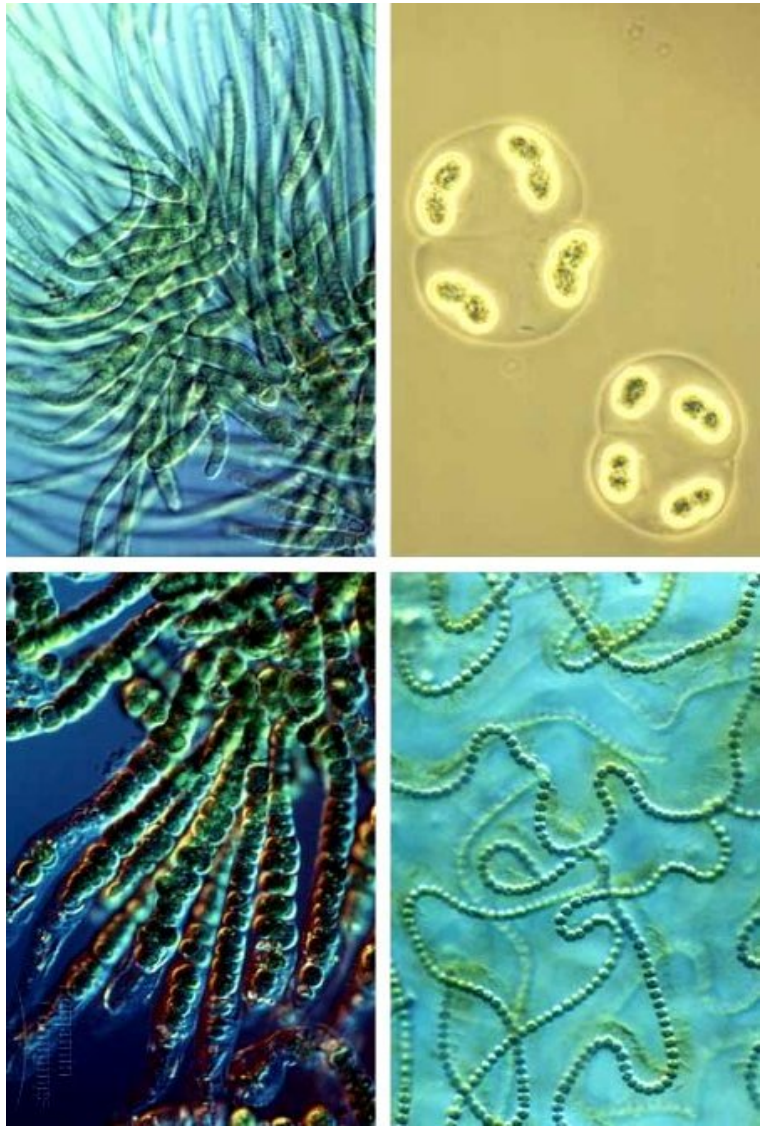
$$\vdots$$



Example: Synechococcus Elongatus

Synechococcus is a genus of cyanobacteria (blue-green algae): single-celled photosynthesising plankton that provide a foundation for the aquatic food chain.

S. Elongatus is a species of Synechococcus that is particularly abundant: some estimates suggest that it contributes 25% of marine nutrient primary production.

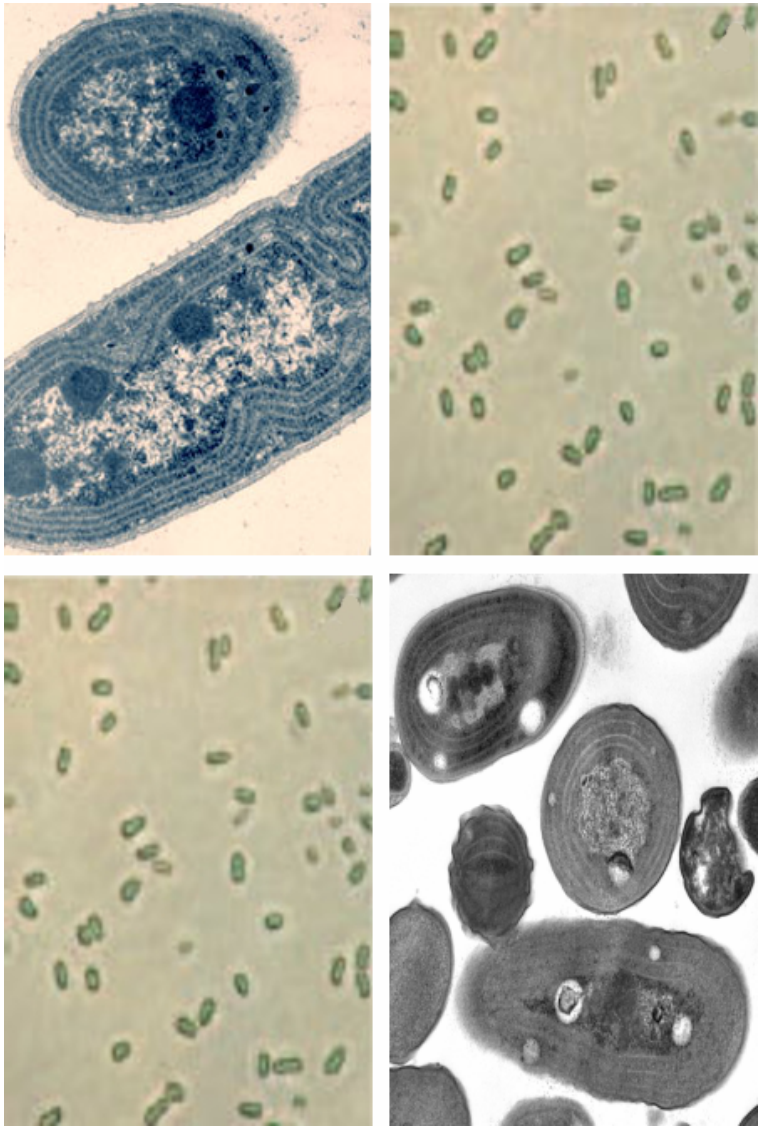


Circadian Clock in S. Elongatus

S. Elongatus has an internal clock, that turns genes on and off through day and night.

The cycling mechanism does not require gene transcription, and will operate in a test tube (*in vitro*).

Although it is entrained by light, it will also run for weeks without external stimulus.



Tomita, Nakajima, Kondo, Iwasaki.
No transcription-translation feedback in circadian rhythm of KaiC phosphorylation.
Science **307**(5707) (2005) 251–254

Proposed Mechanism

The S. Elongatus clock requires three proteins: KaiA, KaiB and KaiC (for *kaiten*). One proposed mechanism is the following:

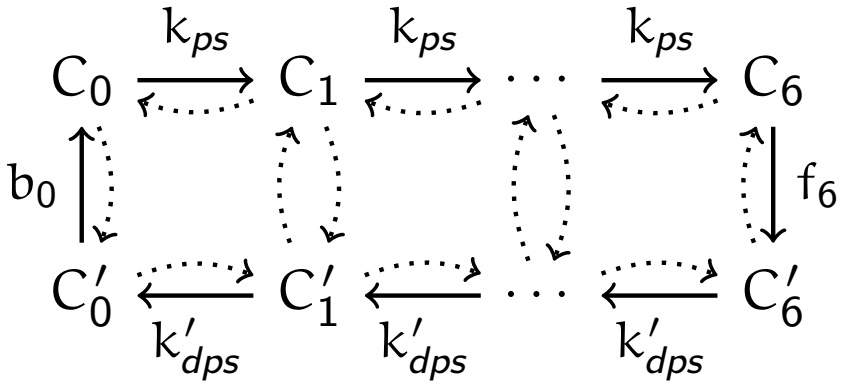
- KaiC forms hexamers, with six phosphorylation sites.
- KaiC also has two conformations; it preferentially phosphorylates in one and dephosphorylates in the other,
- KaiA catalyses phosphorylation of the first (active) conformation.
- KaiB dimers stabilise the second (inactive) conformation.
- A KaiB dimer bound to KaiC will bind a further two KaiA, removing them from other possible interactions.
- Cyclic phosphorylation of individual KaiC gives the basic mechanism; interaction with varying levels of KaiA and KaiB coordinates this across the cell.

van Zon, Lubensky, Altena, ten Wolde.

An allosteric model of circadian KaiC phosphorylation.

PNAS **104**(18) (2007) 7420–7425

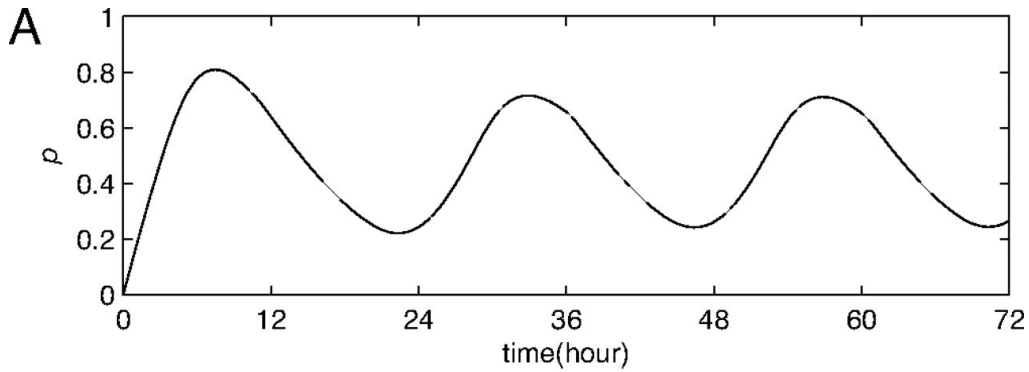
ODE Model



Active forms

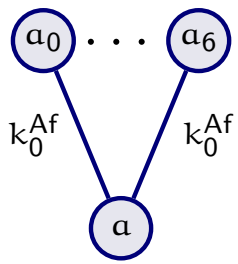
Inactive forms

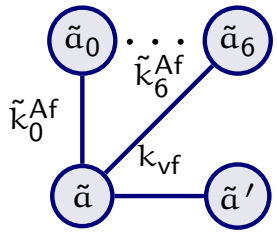
van Zon et al. give an ODE model of this mechanism, and show that it cycles. They conjecture that *differential affinities* are a key feature.

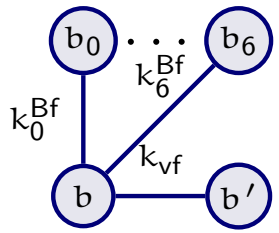


Continuous π Model

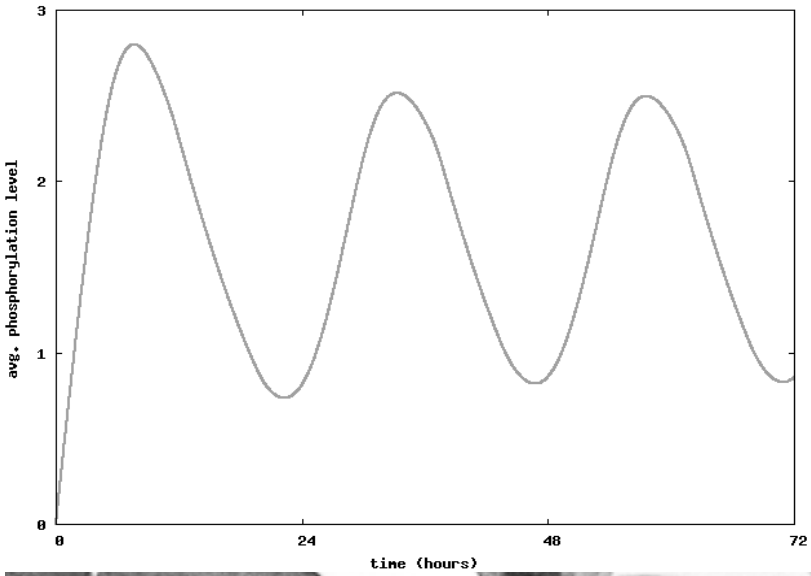
$$
\begin{aligned}
C_i &= (\nu M_i)(\tau@k_{ps}.C_{i+1} + \tau@f_i.\tilde{C}_i + \tau@k_{dps}.C_{i-1} + a_i\langle act_i\rangle.(u_i.C_i + r_i.C_{i+1})) \\
\tilde{C}_i &= \tau@\tilde{k}_{ps}.\tilde{C}_{i+1} + \tau@b_i.C_i + \tau@\tilde{k}_{dps}.\tilde{C}_{i-1} + b_i.b'.B\tilde{C}_i \\
B\tilde{C}_i &= \tau@\tilde{k}_{ps}.B\tilde{C}_{i+1} + \tau@k_i^{Bb}.(\tilde{C}_i \mid B \mid B) + \tau@\tilde{k}_{dps}.B\tilde{C}_{i-1} + \tilde{a}_i.\tilde{a}'.AB\tilde{C}_i \\
AB\tilde{C}_i &= \tau@\tilde{k}_{ps}.AB\tilde{C}_{i+1} + \tau@\tilde{k}_i^{Ab}.(B\tilde{C}_i \mid A \mid A) + \tau@\tilde{k}_{dps}.AB\tilde{C}_{i-1} \\
A &= a(x).x.A + \tilde{a}.0 \\
B &= b.0 \\
\\
\mathbf{P} &= \mathbf{c_A}\cdot\mathbf{A} \parallel \mathbf{c_B}\cdot\mathbf{B} \parallel \mathbf{c_C}\cdot\mathbf{C}_0
\end{aligned}
$$

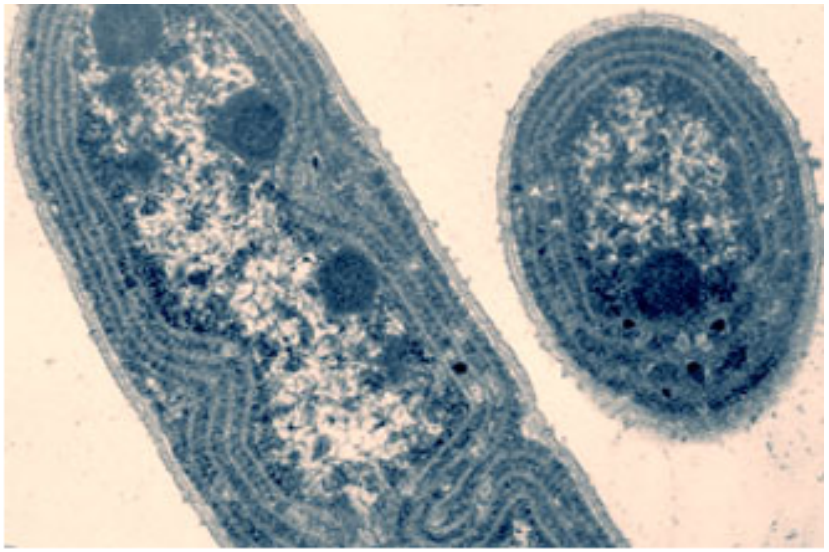


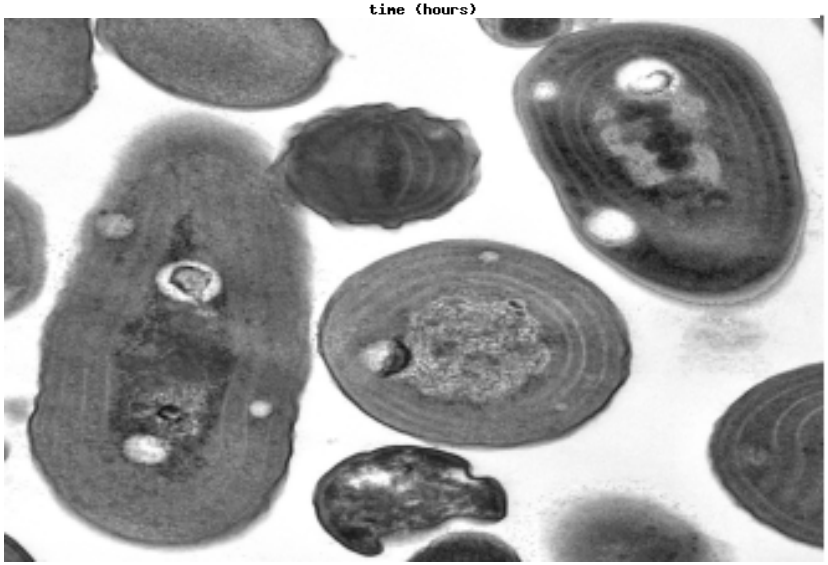


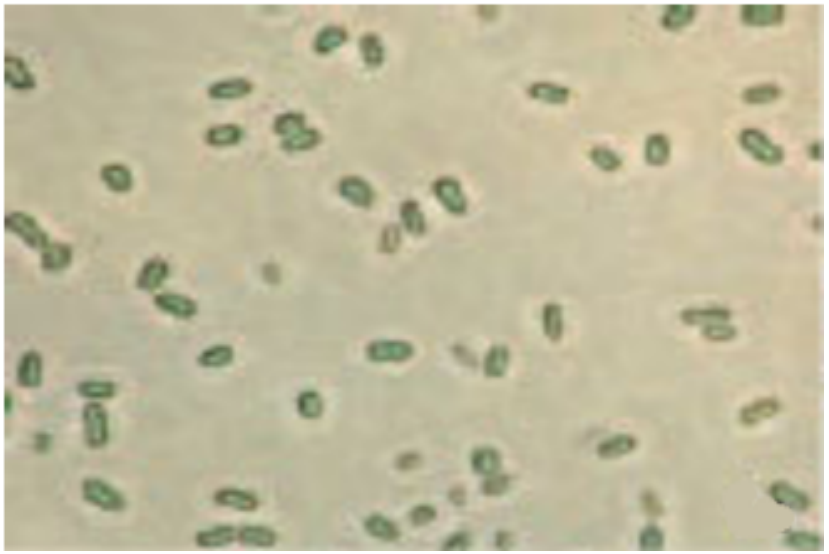


Running π



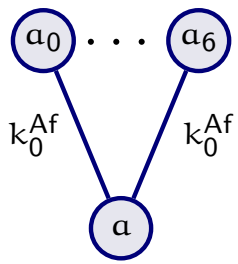


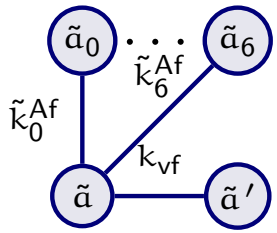


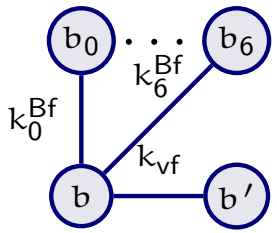


Modification: Remove autonomous phosphorylation

$$
\begin{aligned}
C_i &= (\nu M_i)(\tau@k_{ps}.C_{i+1} + \tau@f_i.\tilde{C}_i + \tau@k_{dps}.C_{i-1} + a_i\langle act_i\rangle.(u_i.C_i + r_i.C_{i+1})) \\
\tilde{C}_i &= \tau@\tilde{k}_{ps}.\tilde{C}_{i+1} + \tau@b_i.C_i + \tau@\tilde{k}_{dps}.\tilde{C}_{i-1} + b_i.b'.B\tilde{C}_i \\
B\tilde{C}_i &= \tau@\tilde{k}_{ps}.B\tilde{C}_{i+1} + \tau@k_i^{Bb}.(\tilde{C}_i \mid B \mid B) + \tau@\tilde{k}_{dps}.B\tilde{C}_{i-1} + \tilde{a}_i.\tilde{a}'.AB\tilde{C}_i \\
AB\tilde{C}_i &= \tau@\tilde{k}_{ps}.AB\tilde{C}_{i+1} + \tau@\tilde{k}_i^{Ab}.(B\tilde{C}_i \mid A \mid A) + \tau@\tilde{k}_{dps}.AB\tilde{C}_{i-1} \\
A &= a(x).x.A + \tilde{a}.0 \\
B &= b.0 \\
\\
\mathbf{P} &= \mathbf{c_A}\cdot\mathbf{A} \parallel \mathbf{c_B}\cdot\mathbf{B} \parallel \mathbf{c_C}\cdot\mathbf{C}_0
\end{aligned}
$$

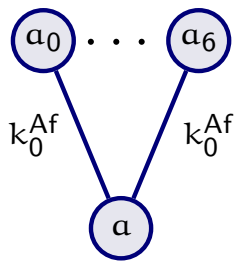


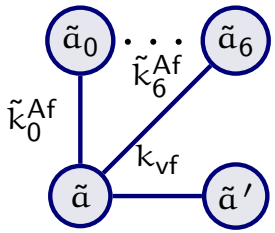


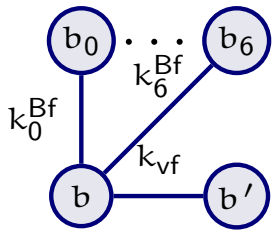


Modification: Remove autonomous phosphorylation

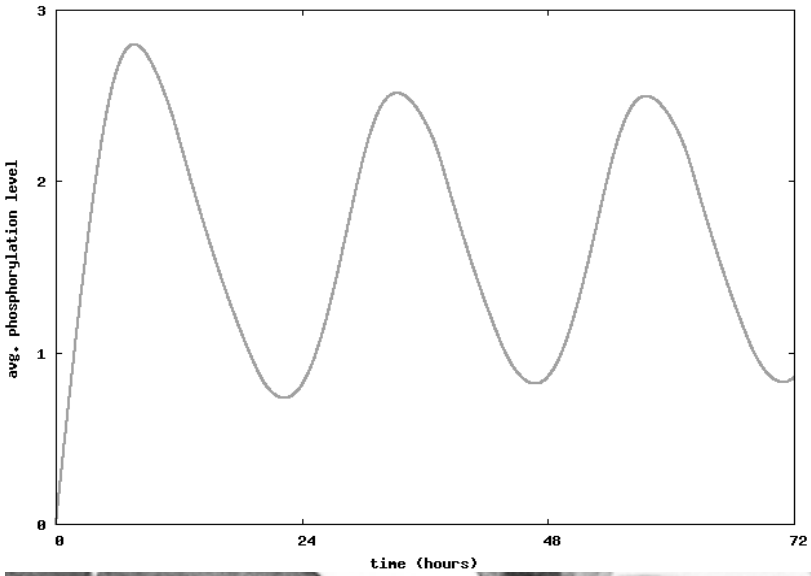
$$
\begin{aligned}
C_i &= (\nu M_i)(\tau@k_{ps}.C_{i+1} + \tau@f_i.\tilde{C}_i + \tau@k_{dps}.C_{i-1} + a_i\langle act_i\rangle.(u_i.C_i + r_i.C_{i+1})) \\
\tilde{C}_i &= \tau@\tilde{k}_{ps}.\tilde{C}_{i+1} + \tau@b_i.C_i + \tau@\tilde{k}_{dps}.\tilde{C}_{i-1} + b_i.b'.B\tilde{C}_i \\
B\tilde{C}_i &= \tau@\tilde{k}_{ps}.B\tilde{C}_{i+1} + \tau@k_i^{Bb}.(\tilde{C}_i \mid B \mid B) + \tau@\tilde{k}_{dps}.B\tilde{C}_{i-1} + \tilde{a}_i.\tilde{a}'.AB\tilde{C}_i \\
AB\tilde{C}_i &= \tau@\tilde{k}_{ps}.AB\tilde{C}_{i+1} + \tau@\tilde{k}_i^{Ab}.(B\tilde{C}_i \mid A \mid A) + \tau@\tilde{k}_{dps}.AB\tilde{C}_{i-1} \\
A &= a(x).x.A + \tilde{a}.0 \\
B &= b.0 \\
\\
\mathbf{P} &= \mathbf{c_A}\cdot\mathbf{A} \parallel \mathbf{c_B}\cdot\mathbf{B} \parallel \mathbf{c_C}\cdot\mathbf{C}_0
\end{aligned}
$$

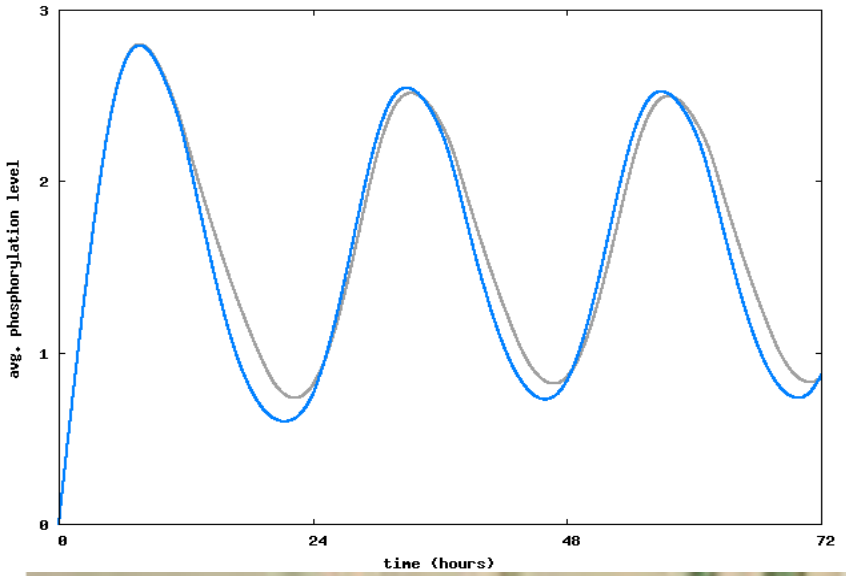


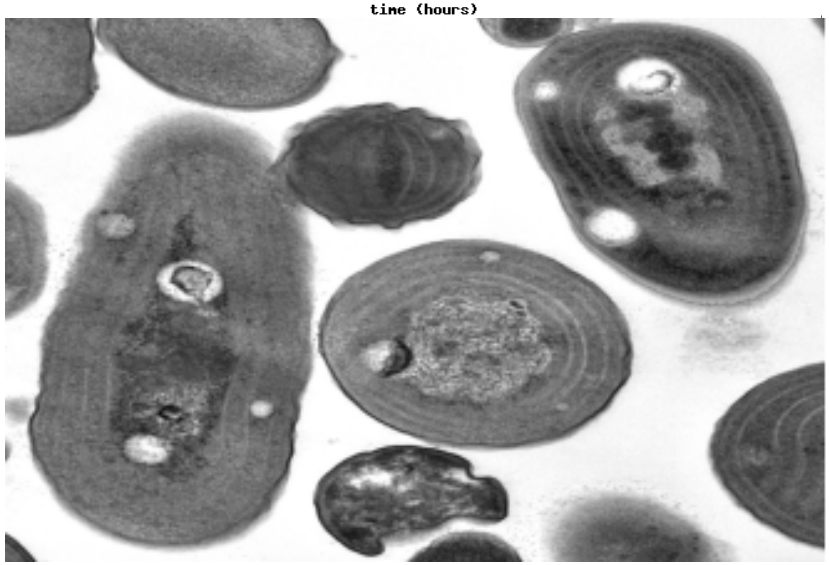


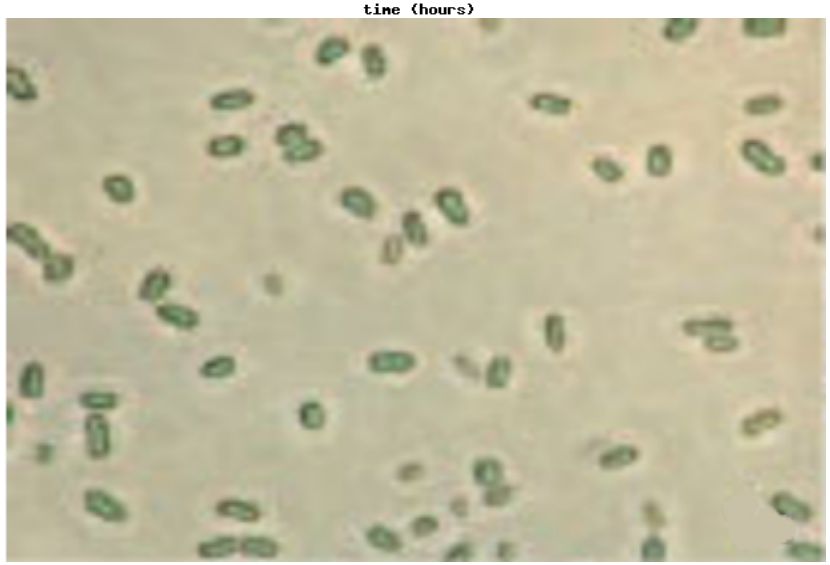


Modification: Remove autonomous phosphorylation



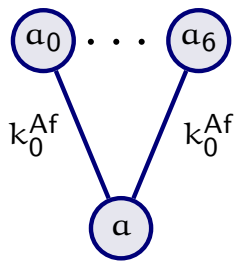


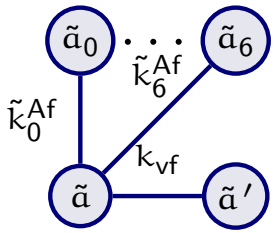


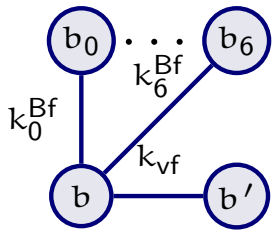


Modification: Weaken KaiA binding

$$
\begin{aligned}
C_i &= (\nu M_i)(\tau@k_{ps}.C_{i+1} + \tau@f_i.\tilde{C}_i + \tau@k_{dps}.C_{i-1} + a_i\langle act_i\rangle.(u_i.C_i + r_i.C_{i+1})) \\
\tilde{C}_i &= \tau@\tilde{k}_{ps}.\tilde{C}_{i+1} + \tau@b_i.C_i + \tau@\tilde{k}_{dps}.\tilde{C}_{i-1} + b_i.b'.B\tilde{C}_i \\
B\tilde{C}_i &= \tau@\tilde{k}_{ps}.B\tilde{C}_{i+1} + \tau@k_i^{Bb}.(\tilde{C}_i \mid B \mid B) + \tau@\tilde{k}_{dps}.B\tilde{C}_{i-1} + \tilde{a}_i.\tilde{a}'.AB\tilde{C}_i \\
AB\tilde{C}_i &= \tau@\tilde{k}_{ps}.AB\tilde{C}_{i+1} + \tau@\tilde{k}_i^{Ab}.(B\tilde{C}_i \mid A \mid A) + \tau@\tilde{k}_{dps}.AB\tilde{C}_{i-1} \\
A &= a(x).x.A + \tilde{a}.0 \\
B &= b.0 \\
\\
\mathbf{P} &= \mathbf{c_A{\cdot}A} \parallel \mathbf{c_B{\cdot}B} \parallel \mathbf{c_C{\cdot}C_0}
\end{aligned}
$$

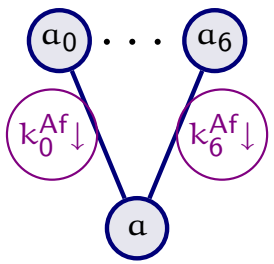


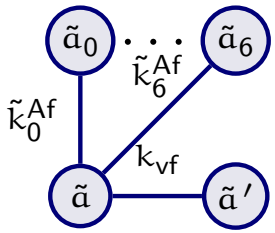


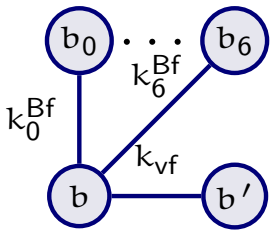


Modification: Weaken KaiA binding

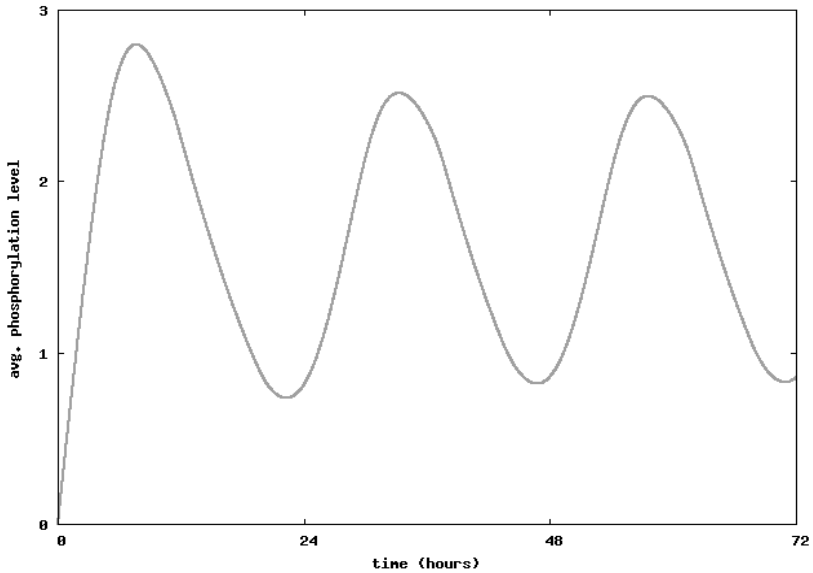
$$
\begin{aligned}
C_i &= (\nu M_i)(\tau@k_{ps}.C_{i+1} + \tau@f_i.\tilde{C}_i + \tau@k_{dps}.C_{i-1} + a_i\langle act_i\rangle.(u_i.C_i + r_i.C_{i+1})) \\
\tilde{C}_i &= \tau@\tilde{k}_{ps}.\tilde{C}_{i+1} + \tau@b_i.C_i + \tau@\tilde{k}_{dps}.\tilde{C}_{i-1} + b_i.b'.B\tilde{C}_i \\
B\tilde{C}_i &= \tau@\tilde{k}_{ps}.B\tilde{C}_{i+1} + \tau@k_i^{Bb}.(\tilde{C}_i \mid B \mid B) + \tau@\tilde{k}_{dps}.B\tilde{C}_{i-1} + \tilde{a}_i.\tilde{a}'.AB\tilde{C}_i \\
AB\tilde{C}_i &= \tau@\tilde{k}_{ps}.AB\tilde{C}_{i+1} + \tau@\tilde{k}_i^{Ab}.(B\tilde{C}_i \mid A \mid A) + \tau@\tilde{k}_{dps}.AB\tilde{C}_{i-1} \\
A &= a(x).x.A + \tilde{a}.0 \\
B &= b.0 \\
\\
\mathbf{P} &= \mathbf{c_A{\cdot}A} \parallel \mathbf{c_B{\cdot}B} \parallel \mathbf{c_C{\cdot}C_0}
\end{aligned}
$$

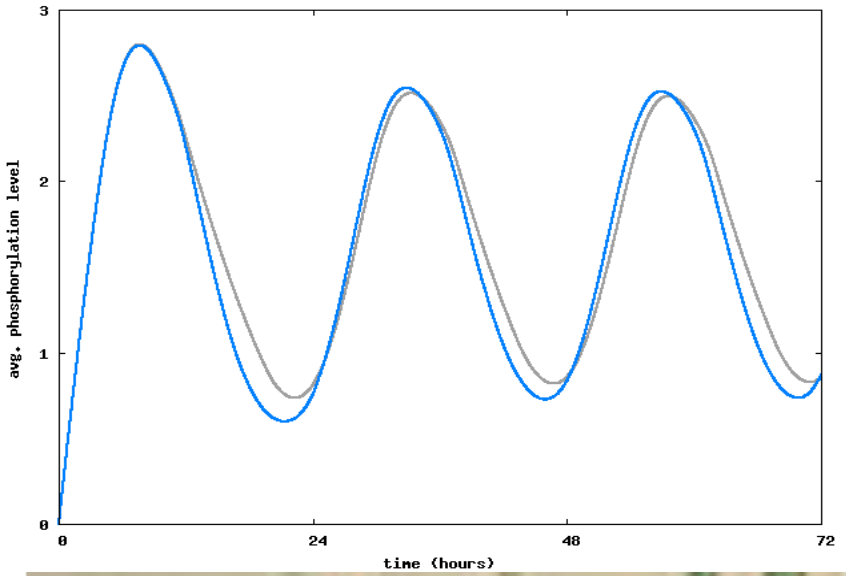


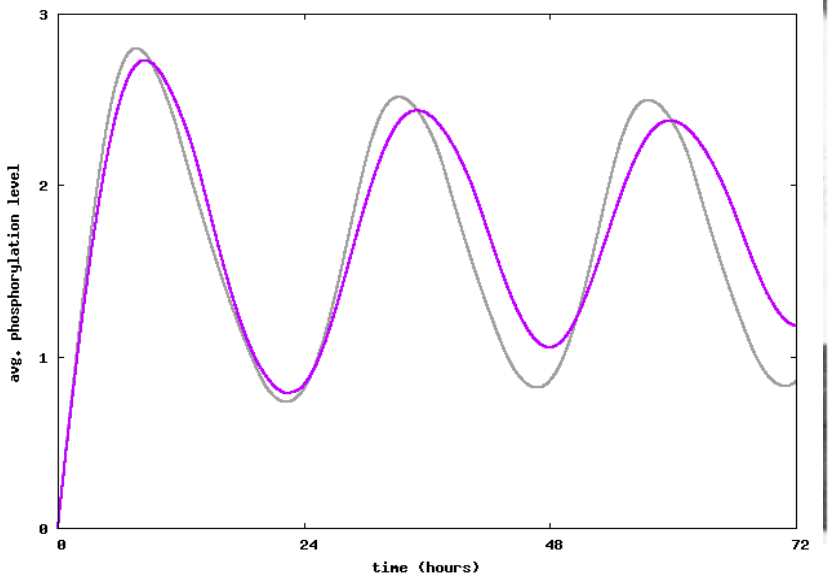


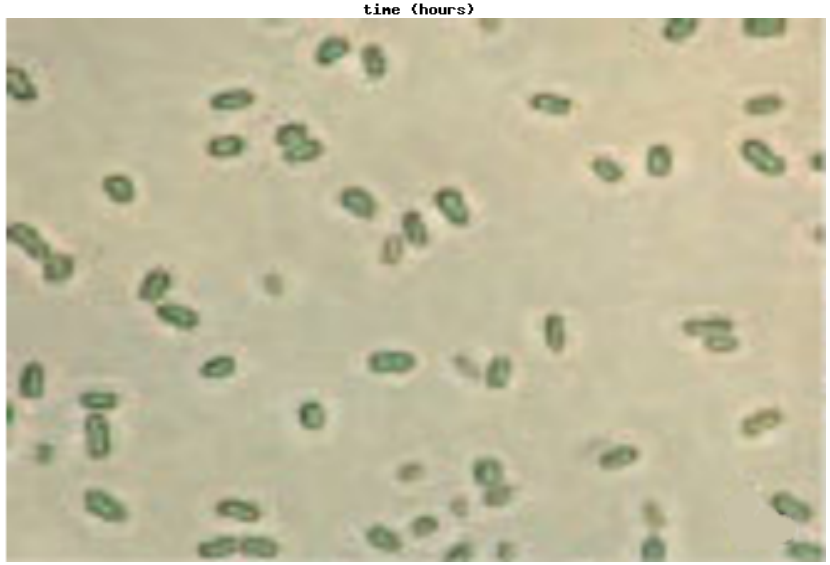


Modification: Weaken KaiA binding



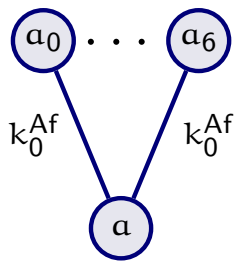


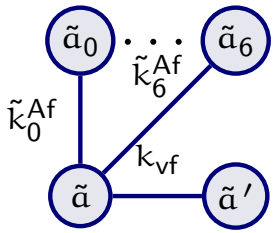


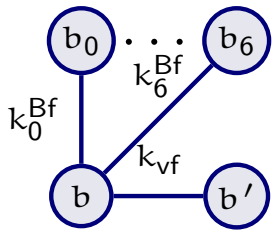


Modification: KaiA-KaiB dimers

$$
\begin{aligned}
C_i &= (\nu M_i)(\tau@k_{ps}.C_{i+1} + \tau@f_i.\tilde{C}_i + \tau@k_{dps}.C_{i-1} + a_i\langle act_i\rangle.(u_i.C_i + r_i.C_{i+1})) \\
\tilde{C}_i &= \tau@\tilde{k}_{ps}.\tilde{C}_{i+1} + \tau@b_i.C_i + \tau@\tilde{k}_{dps}.\tilde{C}_{i-1} + b_i.b'.B\tilde{C}_i \\
B\tilde{C}_i &= \tau@\tilde{k}_{ps}.B\tilde{C}_{i+1} + \tau@k_i^{Bb}.(\tilde{C}_i \mid B \mid B) + \tau@\tilde{k}_{dps}.B\tilde{C}_{i-1} + \tilde{a}_i.\tilde{a}'.AB\tilde{C}_i \\
AB\tilde{C}_i &= \tau@\tilde{k}_{ps}.AB\tilde{C}_{i+1} + \tau@\tilde{k}_i^{Ab}.(B\tilde{C}_i \mid A \mid A) + \tau@\tilde{k}_{dps}.AB\tilde{C}_{i-1} \\
A &= a(x).x.A + \tilde{a}.0 \\
B &= b.0 \\
\\
\mathbf{P} &= \mathbf{c_A}\cdot\mathbf{A} \parallel \mathbf{c_B}\cdot\mathbf{B} \parallel \mathbf{c_C}\cdot\mathbf{C}_0
\end{aligned}
$$

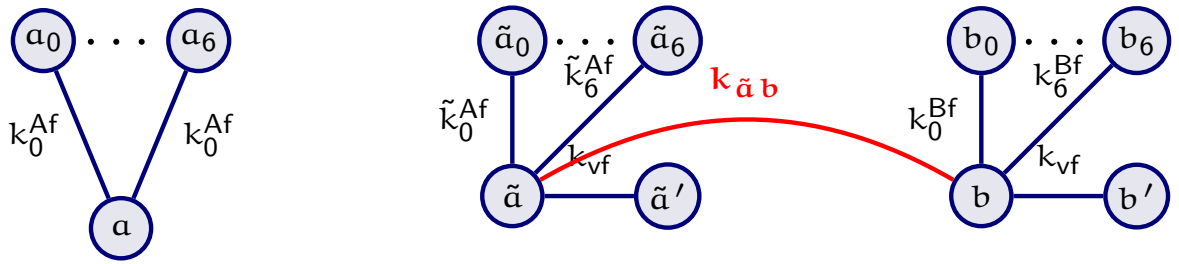




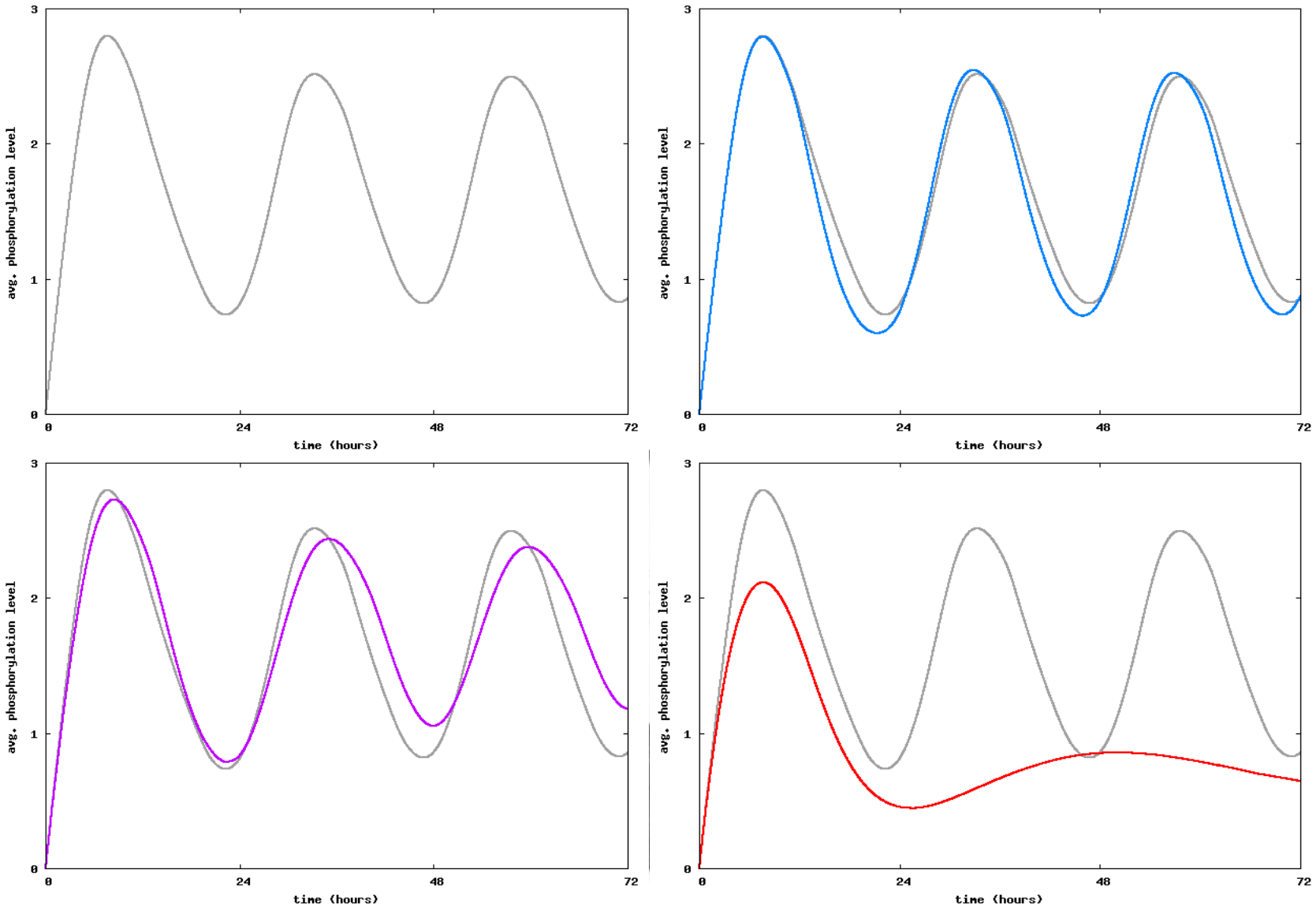


Modification: KaiA-KaiB dimers

$$
\begin{aligned}
C_i &= (\nu M_i)(\tau@k_{ps}.C_{i+1} + \tau@f_i.\tilde{C}_i + \tau@k_{dps}.C_{i-1} + a_i\langle act_i\rangle.(u_i.C_i + r_i.C_{i+1})) \\
\tilde{C}_i &= \tau@\tilde{k}_{ps}.\tilde{C}_{i+1} + \tau@b_i.C_i + \tau@\tilde{k}_{dps}.\tilde{C}_{i-1} + b_i.b'.B\tilde{C}_i \\
B\tilde{C}_i &= \tau@\tilde{k}_{ps}.B\tilde{C}_{i+1} + \tau@k_i^{Bb}.(\tilde{C}_i \mid B \mid B) + \tau@\tilde{k}_{dps}.B\tilde{C}_{i-1} + \tilde{a}_i.\tilde{a}'.AB\tilde{C}_i \\
AB\tilde{C}_i &= \tau@\tilde{k}_{ps}.AB\tilde{C}_{i+1} + \tau@\tilde{k}_i^{Ab}.(B\tilde{C}_i \mid A \mid A) + \tau@\tilde{k}_{dps}.AB\tilde{C}_{i-1} \\
A &= a(x).x.A + \tilde{a}.0 \\
B &= b.0 \\
\\
\mathbf{P} &= \mathbf{c_A}\cdot\mathbf{A} \parallel \mathbf{c_B}\cdot\mathbf{B} \parallel \mathbf{c_C}\cdot\mathbf{C}_0
\end{aligned}
$$



Modification: KaiA-KaiB dimers



Review

- Continuous π-calculus
  - Modular description of biomolecular systems
  - Compositional semantics in real vector spaces
  - Flexible interaction structure
- S. Elongatus circadian clock
  - Protein-protein interaction *in vitro*
  - Candidate mechanism oscillates
  - Behaviour under system variation

Future Work

- Behavioural analysis
  - Continuous temporal logic $P \vdash G_{\leqslant t}(\phi)$ and $Q \vdash F^{c \cdot a}_{\leqslant t} G(\psi)$
  - Model checking
  - Similarity metric
- System Evolution
  - Evolutionary trajectories
  - Variation, evolvability
  - Robustness and neutrality
- Alternative Semantics
  - Markov chains
  - Stochastic simulation
  - Hybrid models, protein/DNA interaction

References

Marek Kwiatkowski and Ian Stark.
The Continuous π-Calculus: A Process Algebra for Biochemical Modelling. In *Computational Methods in Systems Biology: Proc. CMSB 2008*
Lecture Notes in Computer Science 5307, pages 103–122. Springer 2008

Tomita, Nakajima, Kondo, Iwasaki.
No transcription-translation feedback in circadian rhythm of KaiC phosphorylation.
Science **307**(5707) (2005) 251–254

van Zon, Lubensky, Altena, ten Wolde.
An allosteric model of circadian KaiC phosphorylation.
PNAS **104**(18) (2007) 7420–7425