Exploring Variation in Biochemical Pathways with the Continuous pi-Calculus

Ian Stark Marek Kwiatkowski Chris Banks

Laboratory for Foundations of Computer Science
SynthSys: Synthetic & Systems Biology
The University of Edinburgh

Evolutionary Ecology
Institute of Integrative Biology
Department of Environmental Sciences
EAWAG & ETH Zürich

Thursday 16 June 2012

SynthSys
SYNTHETIC & SYSTEMS BIOLOGY

THE UNIVERSITY OF EDINBURGH

Summary

The *continuous pi-calculus* (c-pi) is a process algebra for modelling behaviour and variation in biomolecular systems: *e.g.* enzyme activation and inhibition; circadian clocks; signalling pathways.

Expressions in c-pi represent mixtures of chemical reagents, and can be compiled to conventional ODE models for fast numerical simulation.

With a language of potential changes in c-pi processes we systematically explore evolutionary neighbourhoods of a specific signalling pathway, and observe instances of robustness, neutrality and evolvability.

A complementary temporal logic for behaviour in context gives a language to classify these variations in behaviour.

Marek Kwiatkowski and Ian Stark.
On Executable Models of Molecular Evolution. In *Proc. 8th International Workshop on Computational Systems Biology WCSB 2011*, pp. 105–108.

The Continuous pi-Calculus

Continuous pi is a name-passing process algebra for modelling behaviour and variation in molecular systems.

Based on Milner's pi-calculus, it introduces real-valued variability in:

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

Although sharing an approach common to process algebras for biomodelling, some features are distinctive. For example, by comparison with the stochastic pi-calculus:

- ODEs are the primary mode of execution, not stochastic simulation
- Continuous concentrations of chemicals replace discrete individuals
- End-to-end channels are replaced by multiple competing names

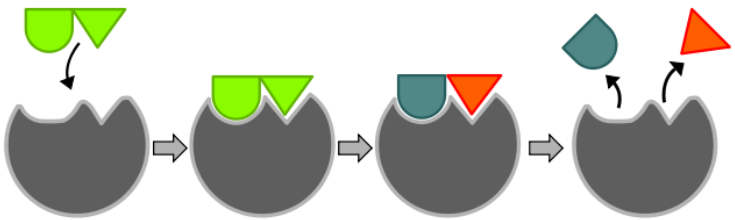
Basics of Continuous pi

Continuous pi 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.

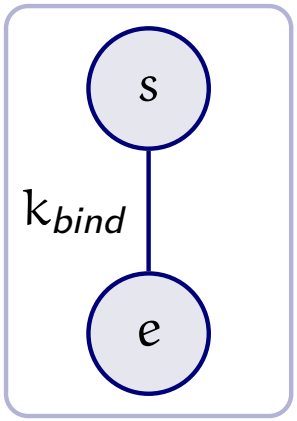
Example: Enzyme Catalysis

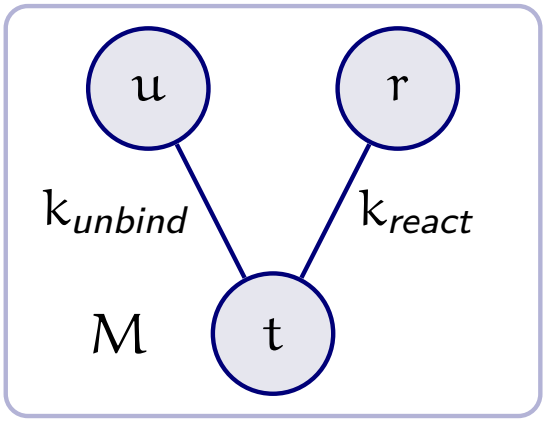


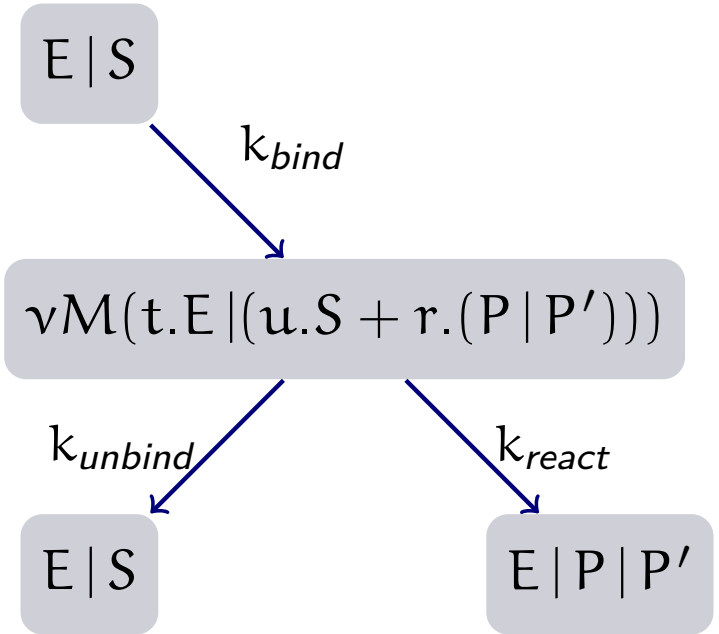
$$S = s(x, y).(x.S + y.(P|P'))$$

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

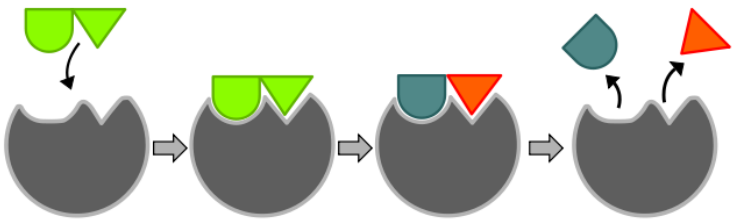
$$P = P' = \tau@k_{degrade}.0$$







Example: Enzyme Catalysis



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

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

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

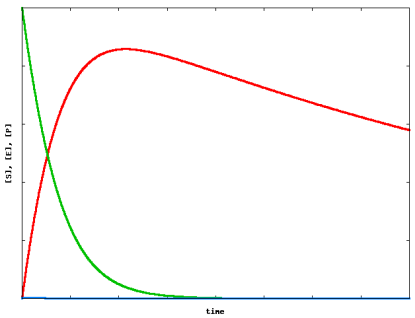
Cpi tool → *Octave*

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

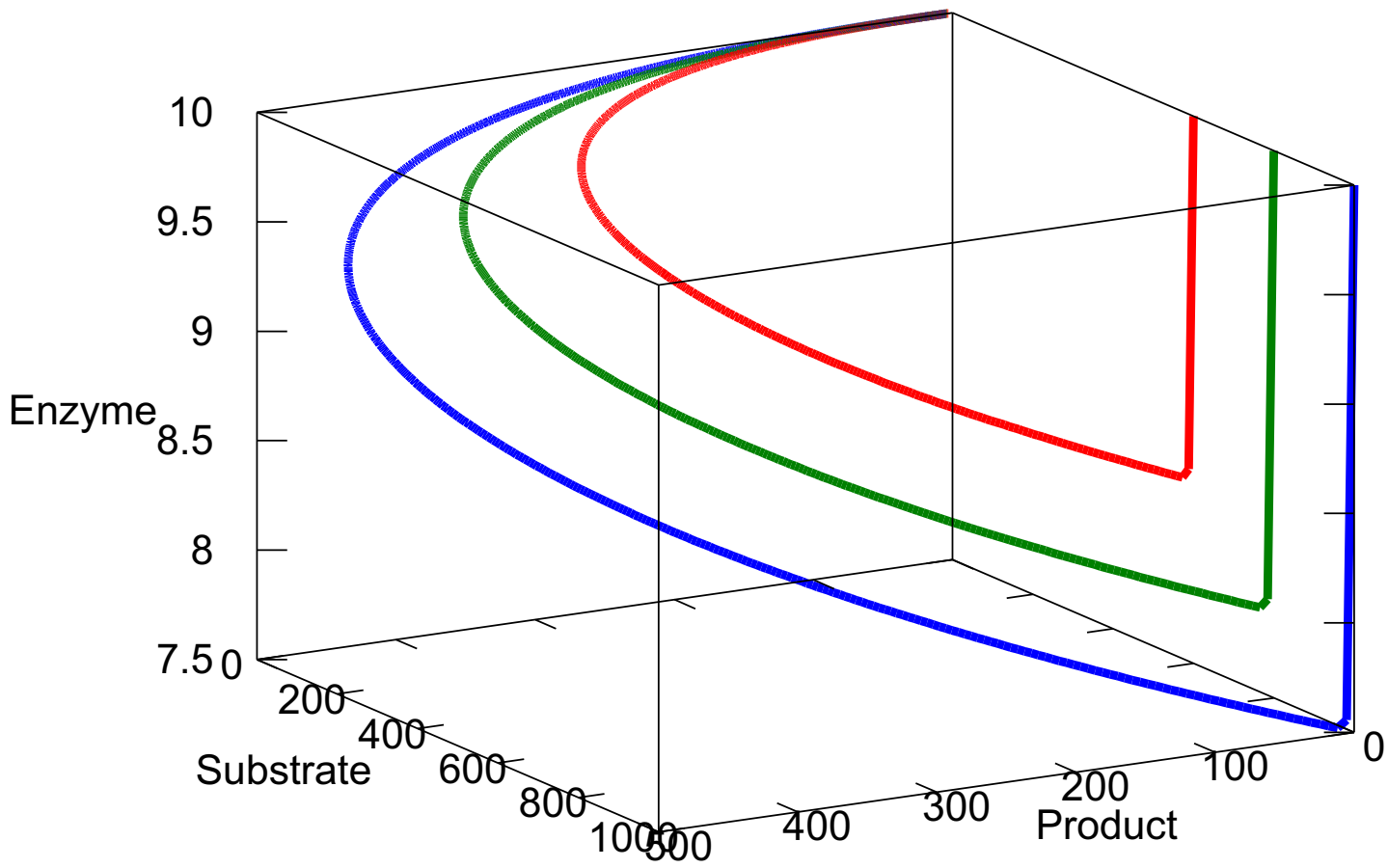
ODEs

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

$$\vdots$$



Process Space: Substrate & Product & Enzyme



Biomodelling in Continuous pi

For individual species, continuous pi uses a modelling idiom based on that of Regev and Shapiro:

- Reagent-centric rather than rule-based
- Individual species are represented by processes
- Complexes are modelled by name restriction $\nu x.(A \mid B)$
- Interaction is modelled by communication between names
- ...but with competition between multiple alternatives

From Species to Processes

Take a language for interaction between individual species and raise it into one for reactions in mixtures:

Species	A, B	$::= \Sigma\alpha.A \mid A \mid B \mid \nu M.A \mid \ldots$
Processes	P, Q	$::= 0 \mid c \cdot A \mid P \parallel Q$

Component $c \cdot A$ of species A at concentration $c \in \mathbb{R}_{\geqslant 0}$.

Mixture of processes $P \parallel Q$.

We can identify processes with elements of *process space* $\mathcal{P} = \mathbb{R}^{\mathcal{S}}$, where $\mathcal{S}$ is the set of species.

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

- Captures available reactivity
- Element of $\mathbb{R}^{\mathcal{N}\times\mathcal{S}\times\mathcal{C}}$

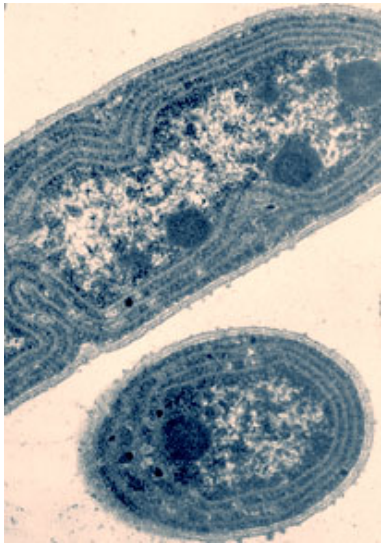
$$\partial(P \parallel Q) = \partial P + \partial Q$$

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

Both $\frac{dP}{dt}$ and ∂P are defined by induction on the structure of processes; and beneath that, from the transitions of component species $c \cdot A$.

With this, we are able to compose the phase portraits of our systems.

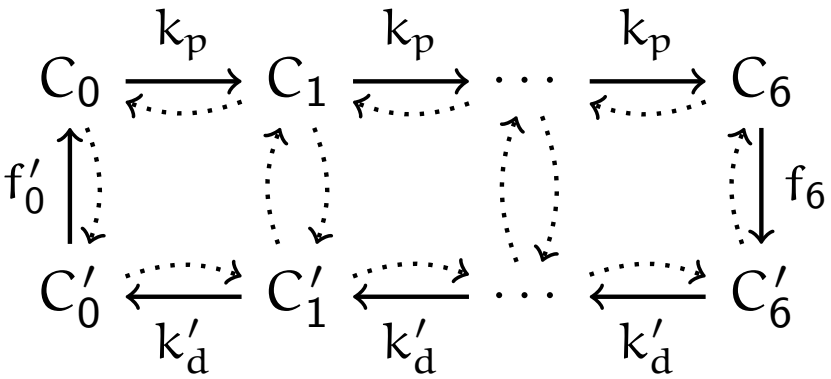
Example: Synechococcus Elongatus

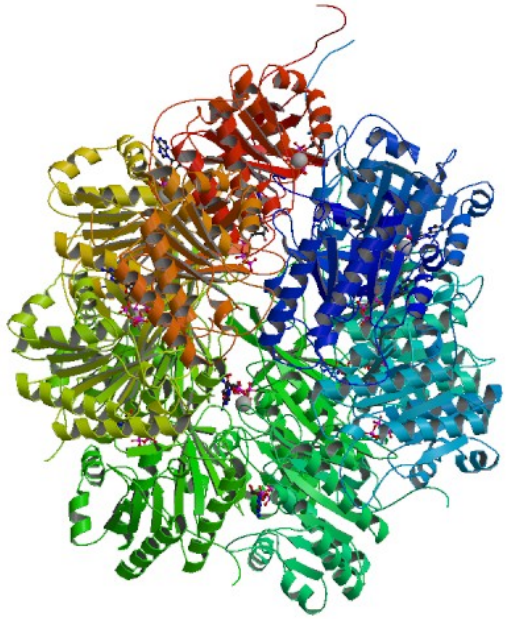


Sherman&Sherman, Purdue

S. Elongatus circadian clock proteins, effective in vitro: KaiA, KaiB and KaiC. (Tomita et al. 2005)

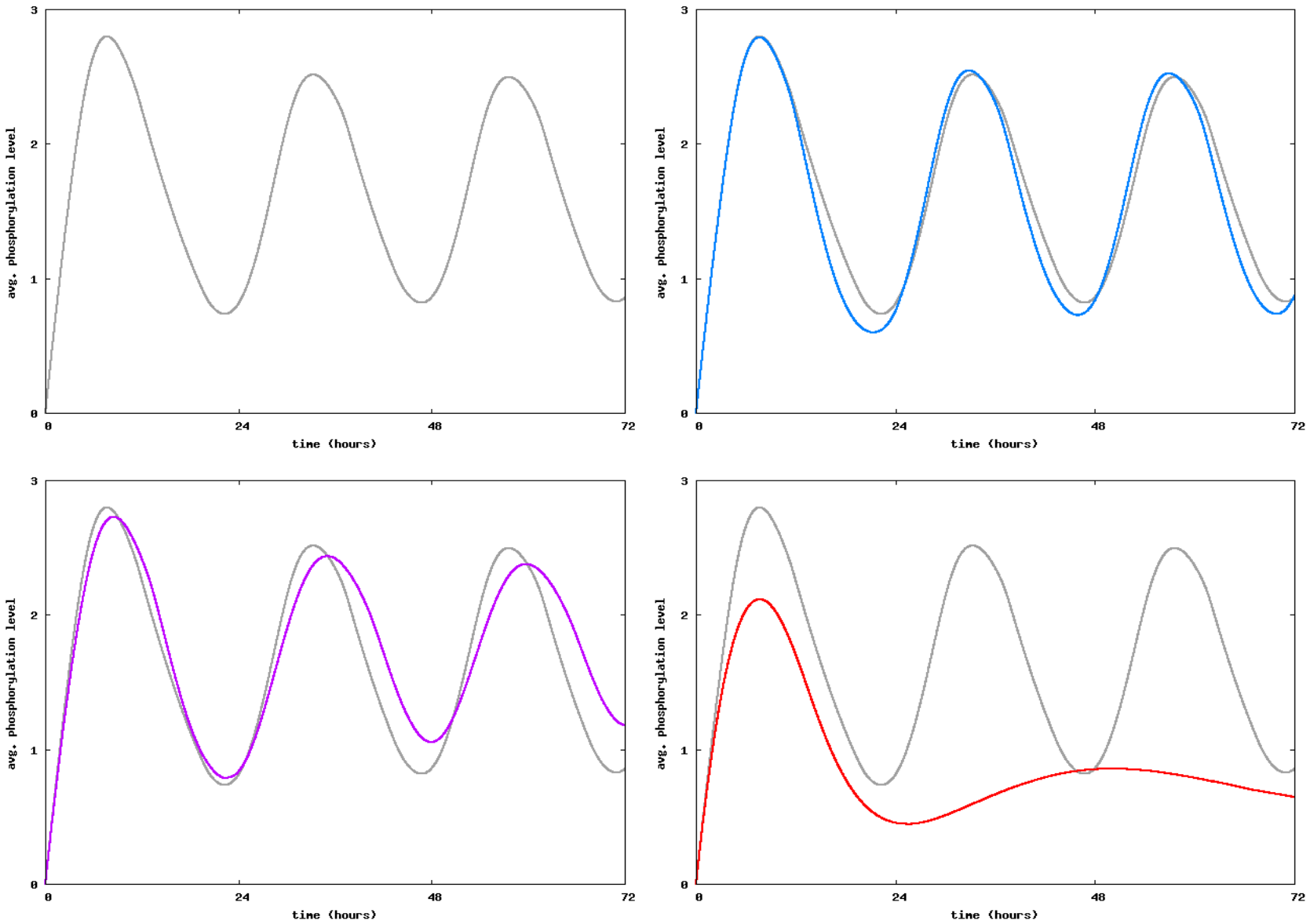
Several mechanisms have been proposed: one is the cyclic six-fold phosphorylation of KaiC hexamers in two alternative conformations, stabilised by KaiA and KaiB. (van Zon et al. 2007)





RCSB Protein Data Bank

Execution and Modification



Process Algebras for Molecular Evolution

One way to model molecular evolution is by specific modifications of concrete mathematical models.

Process algebras, and similar intermediate languages, offer a framework to generalise this model for variation and selection.

$$\text{Process} \sim \text{Genotype}$$
$$\text{Execution} \sim \text{Development}$$
$$\text{Behaviour} \sim \text{Phenotype}$$

Relevant features of models like continuous pi include:

- Reagent-centric models to match genetic variation
- Free formation of new terms, particularly novel complexes
- Computable behaviour of created components

Variation Operators

Variation operators are transformations of c-pi models which correspond to evolutionary events.

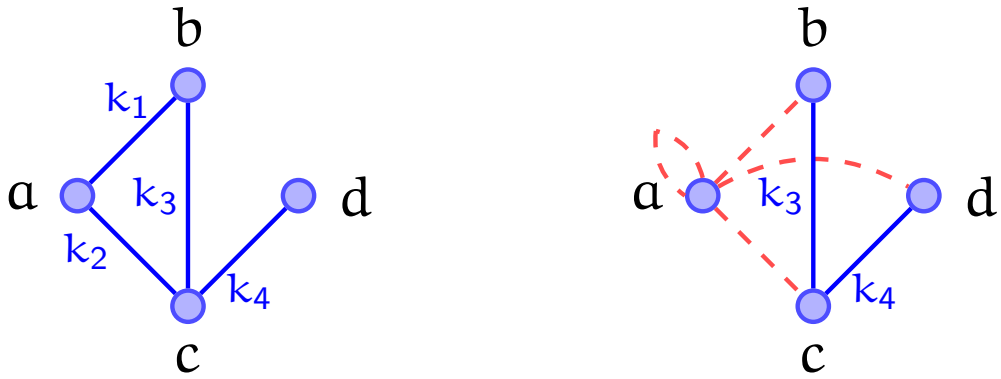
Ideally, a suite of such operations should:

- Maintain the biological idiom
- Be biologically meaningful
- Be expressive enough to build new reaction networks from scratch

Variation Operators

Variation operators are transformations of c-pi models which correspond to evolutionary events.

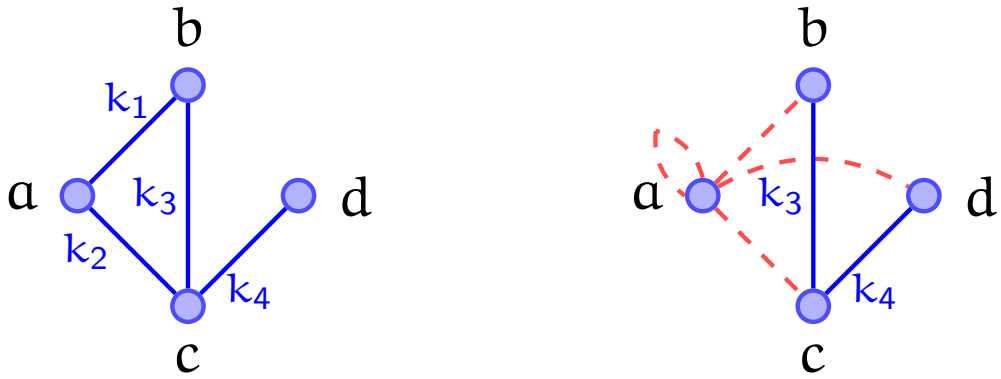
For example: site reconfiguration



Variation Operators

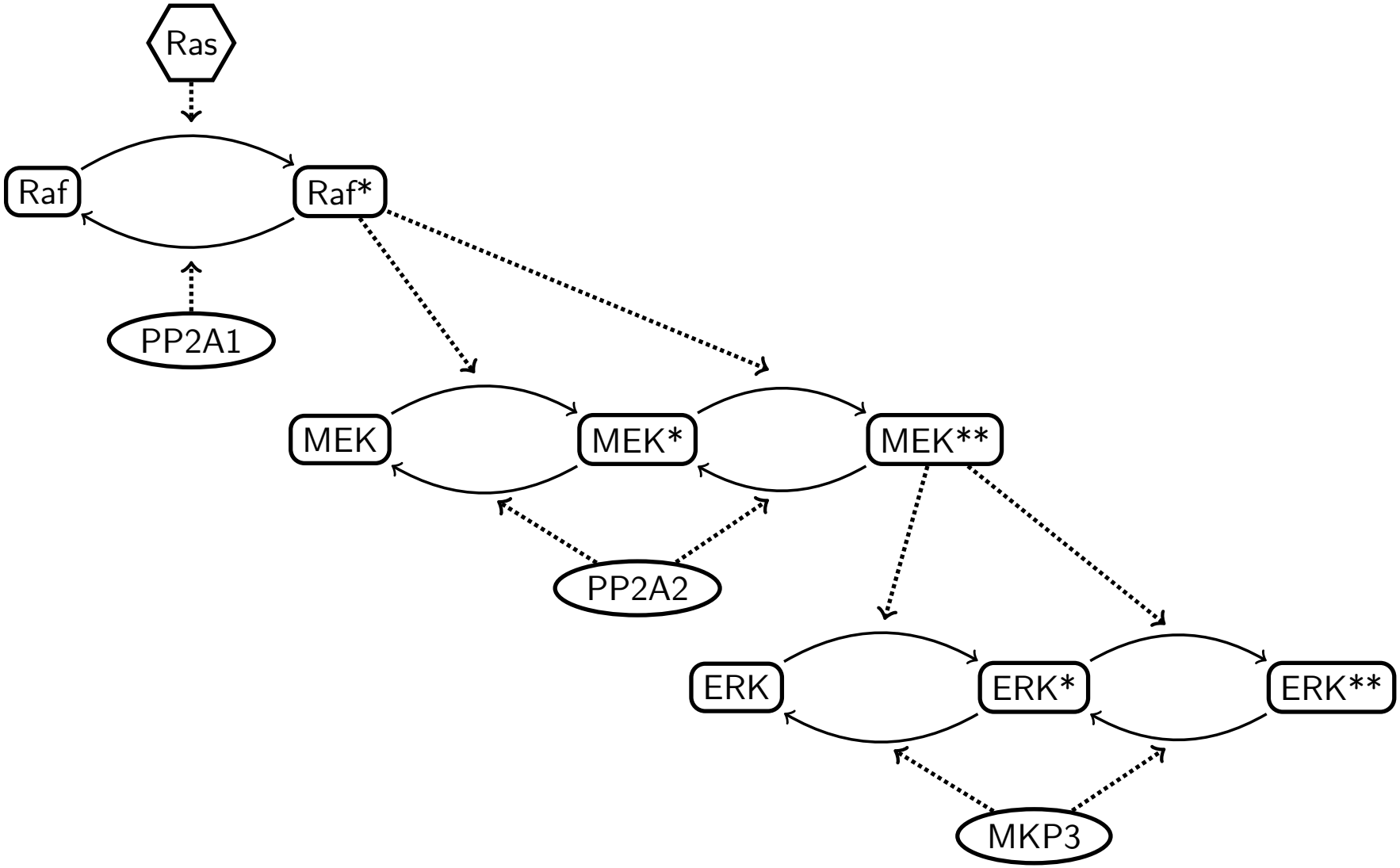
Variation operators are transformations of c-pi models which correspond to evolutionary events.

For example: site reconfiguration



We have defined a dozen such operators modelling gene duplications, gene knockouts, changes in activity rates within complexes, and more.

Simplified MAPK Cascade



MAPK in Continuous pi

$$Ras = (\nu x \frown \overline{x})\, ras(x;y).(\overline{x}.Ras + y.Ras)$$

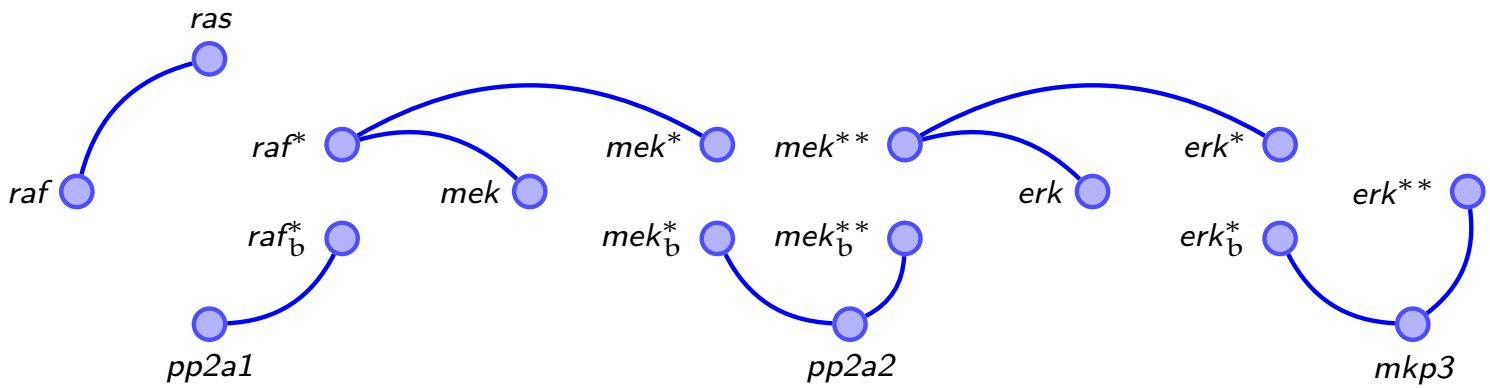
$$Raf = (\nu x \frown \overline{x})\, raf(x;y).(\overline{x}.Raf + y.Raf^{*})$$

$$\ldots$$

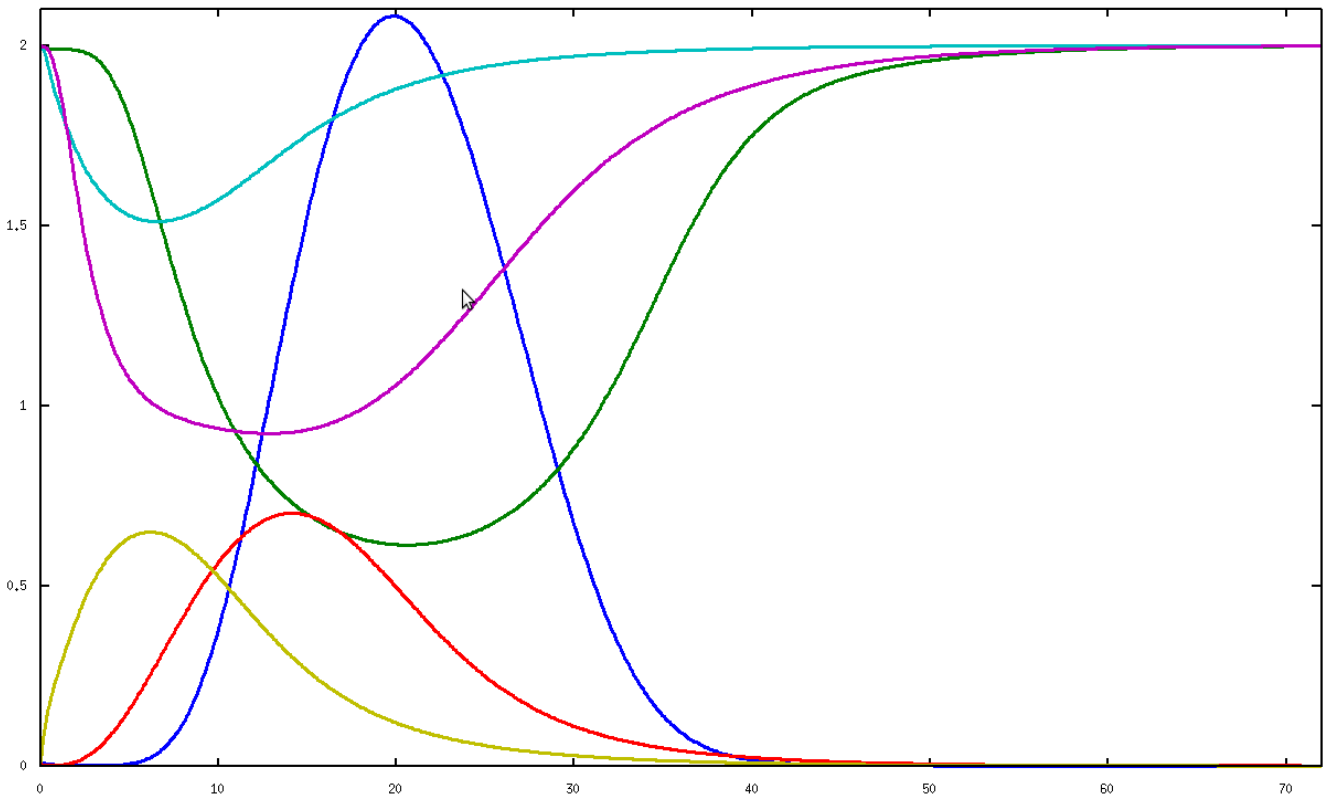
$$ERK^{**} = (\nu x \frown \overline{x})\, erk^{**}_{b}(x;y).(\overline{x}.ERK^{**} + y.ERK^{*})$$

$$MKP3 = (\nu x \frown \overline{x})\, mkp3(x;y).(\overline{x}.MKP3 + y.MKP3)$$

$$\Pi = \mathbf{c_1} \cdot \boldsymbol{Raf} \parallel \mathbf{c_2} \cdot \boldsymbol{Ras} \parallel \ldots \parallel \mathbf{c_4} \cdot \boldsymbol{ERK} \parallel \mathbf{c_7} \cdot \boldsymbol{MKP3}$$

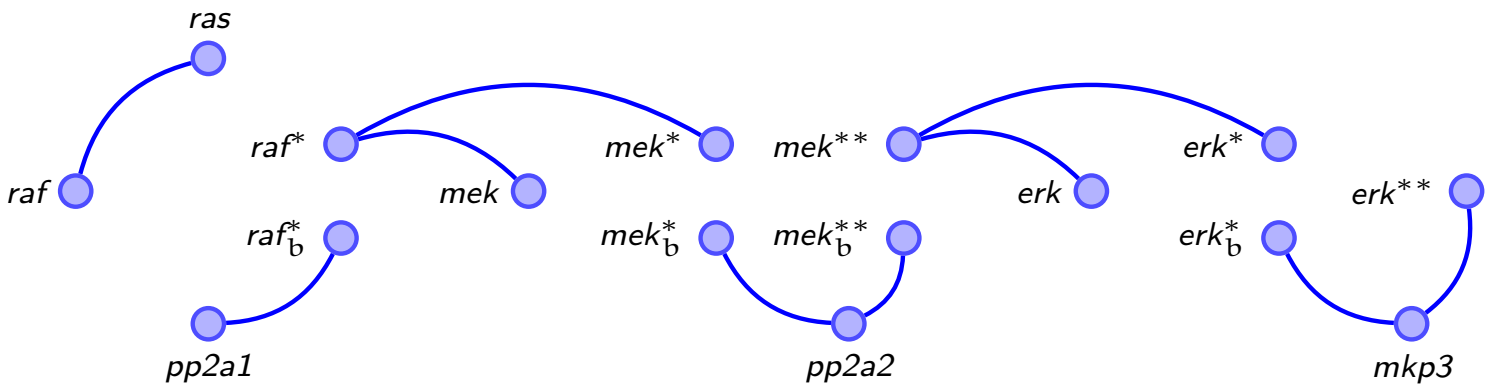


MAPK Behaviour



This MAPK model compiles into 23 differential equations, which are then solved with Octave. The signalling cascade correctly transmits initial presence of **Ras** into a peak of **ERK**** via **Raf*** and **MEK****.

Evolutionary Analysis of MAPK



- Reconfigure every site in every way possible ($16 \times 2^{16} \approx 10^6$).
- Generate ODEs and hence behaviour traces for every variant.

Qualitative analysis

- Classify phenotypes with LTL model-checking
- Find evolutionarily fragile and robust sites

Quantitative analysis

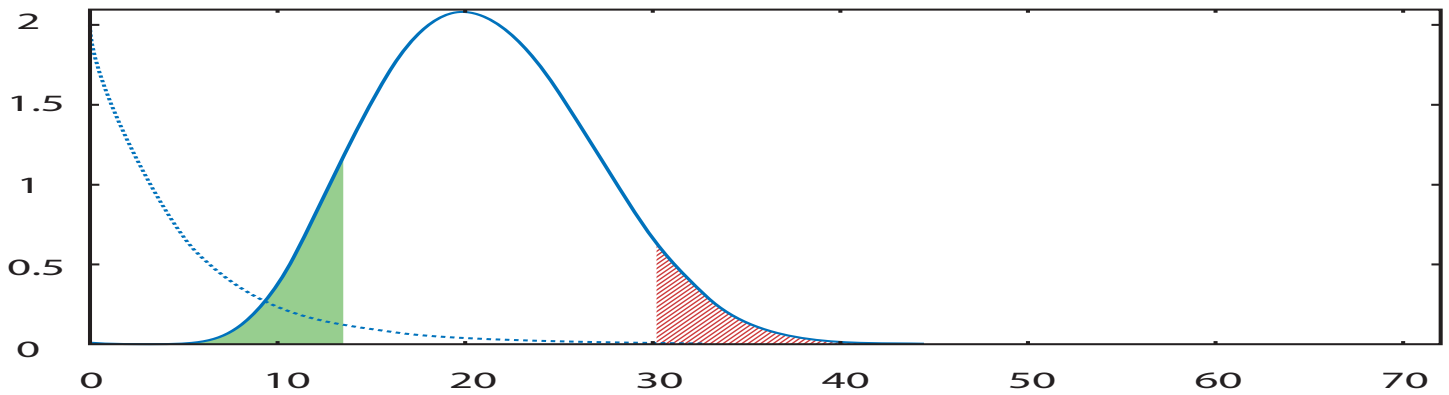
- Compute the fitness of every variant using signal integration
- Find the distribution of mutation effects on fitness

Phenotype Classes and Fitness

Phenotype classes

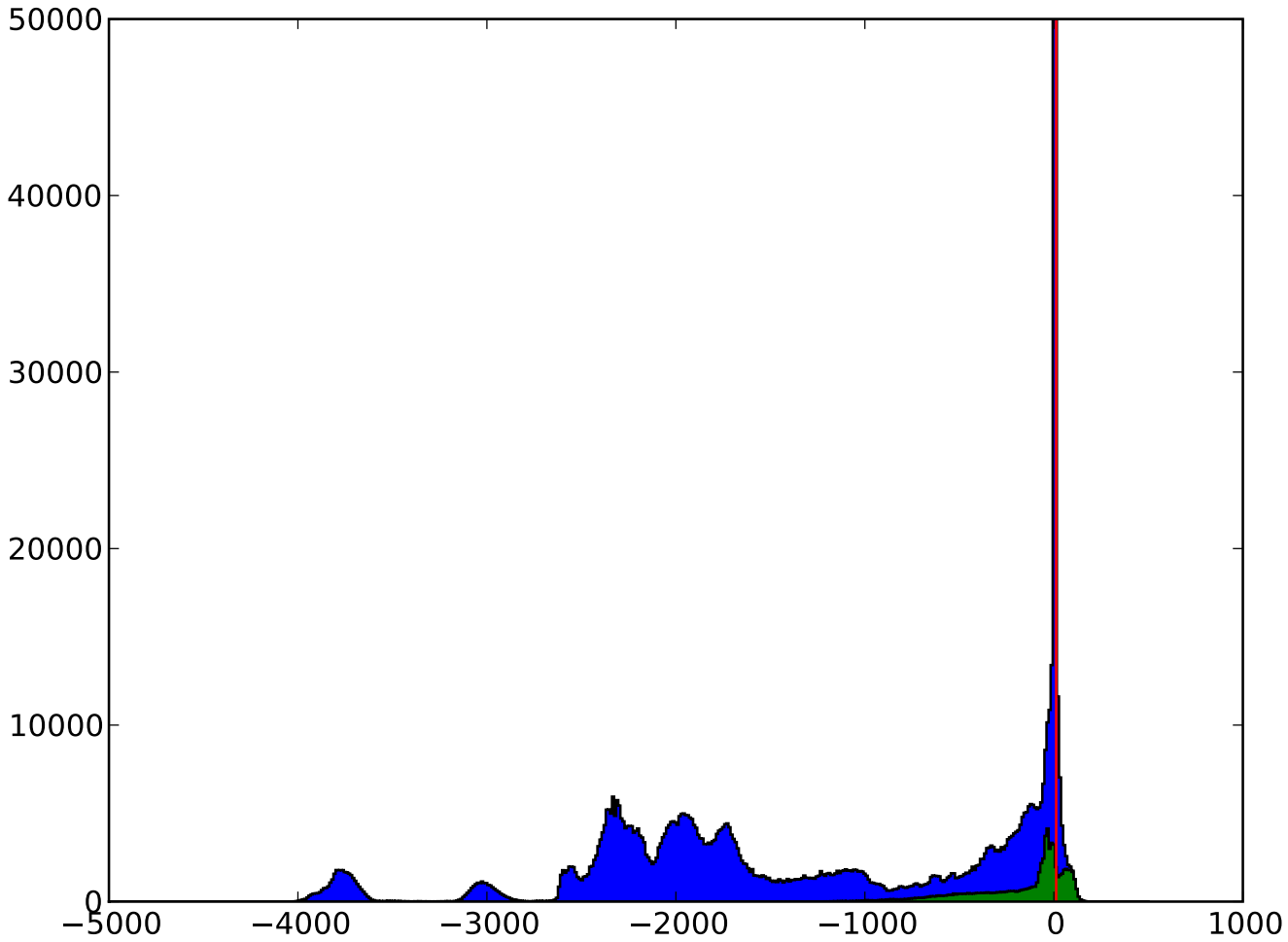
- Four categories: peak, switch, oscillatory, noise.
- Automatically identified using LTL checking.
- Results: peak 7.0%; switch 45.2%; oscillatory 0.0; noise 47.8%.

Fitness



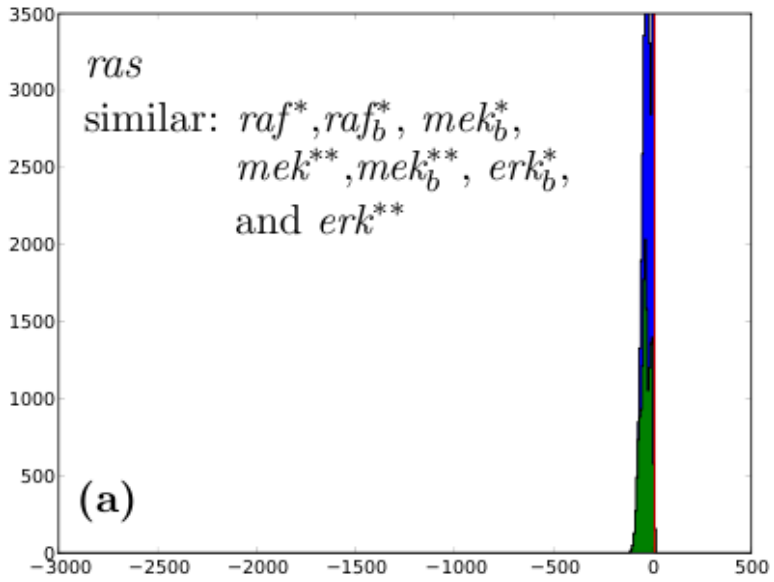
Fitness is the area marked green minus the area marked red.

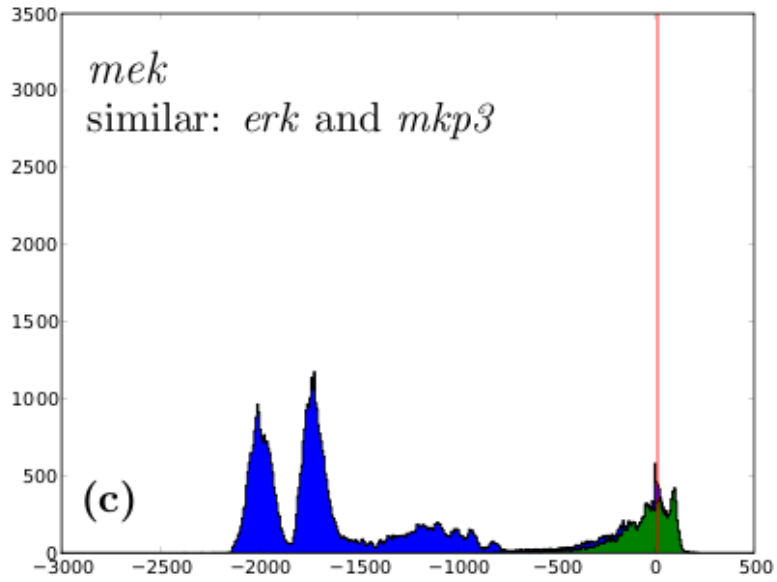
Fitness Distribution

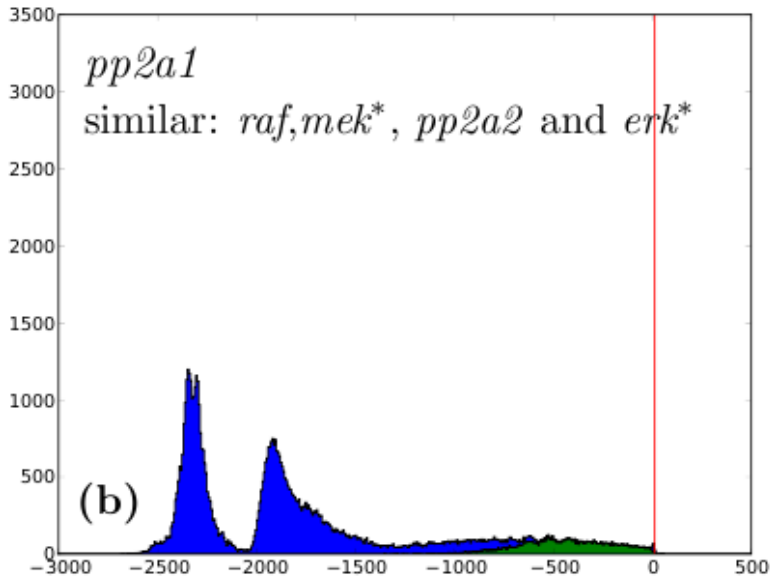


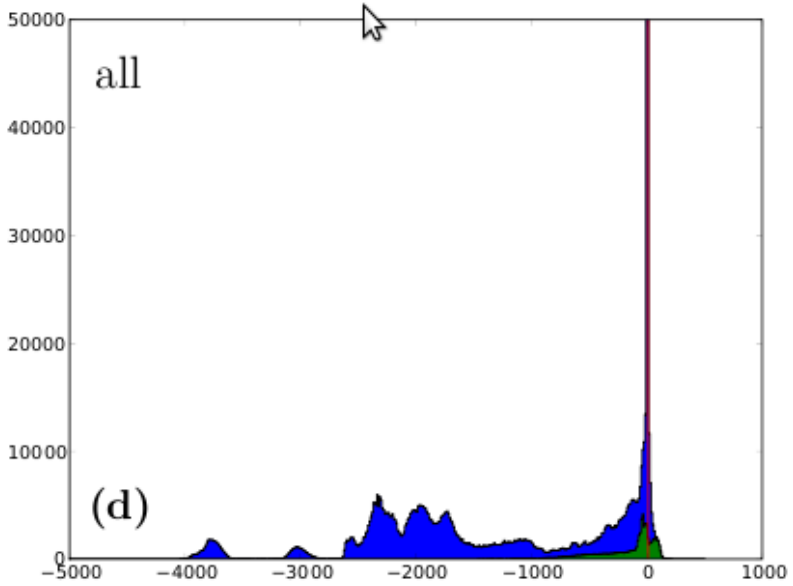
Histogram with 500 evenly-sized bins; green sections are *peak* variants; red vertical line shows initial model.

Fitness Distributions by Site Modified

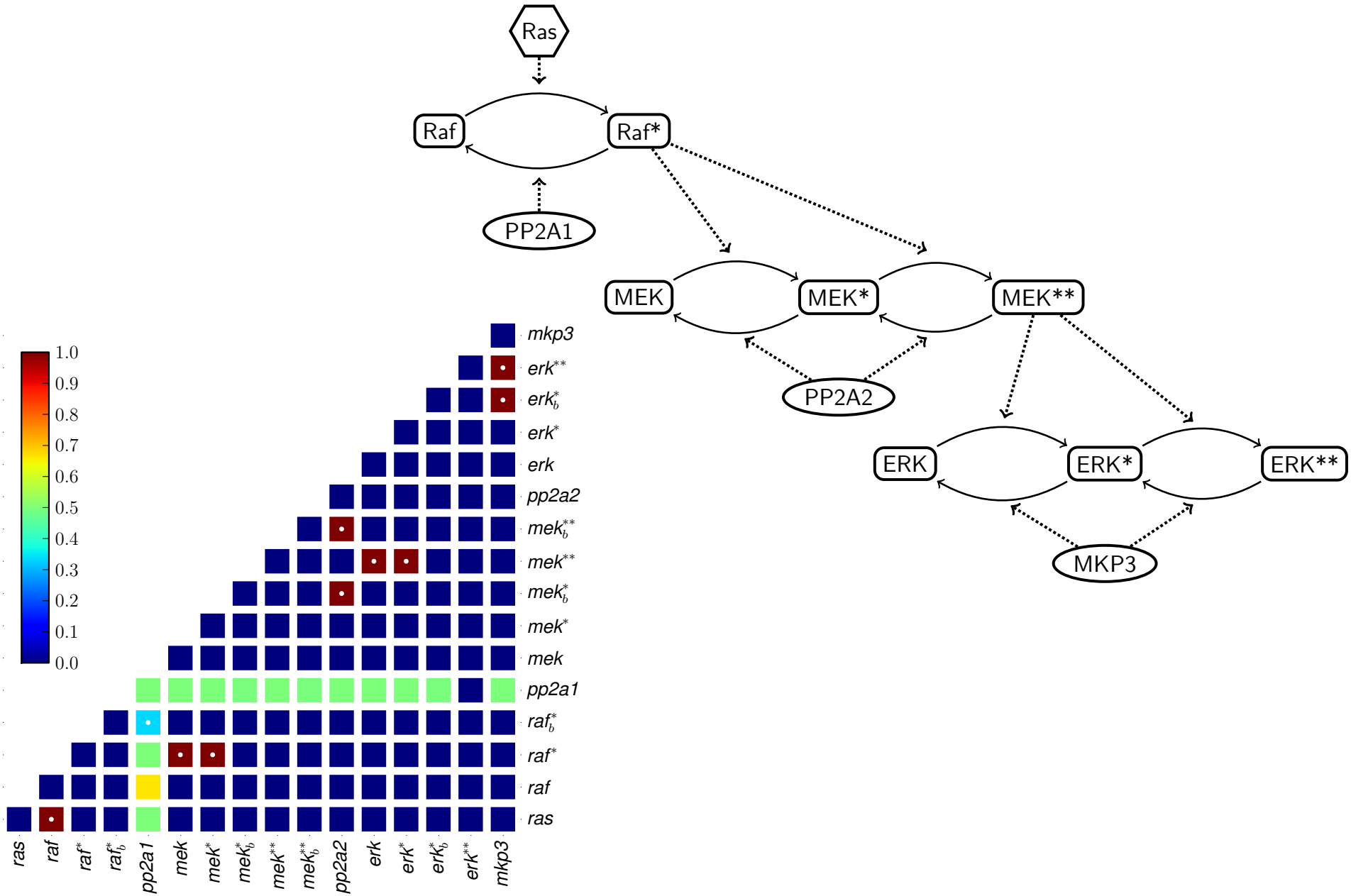




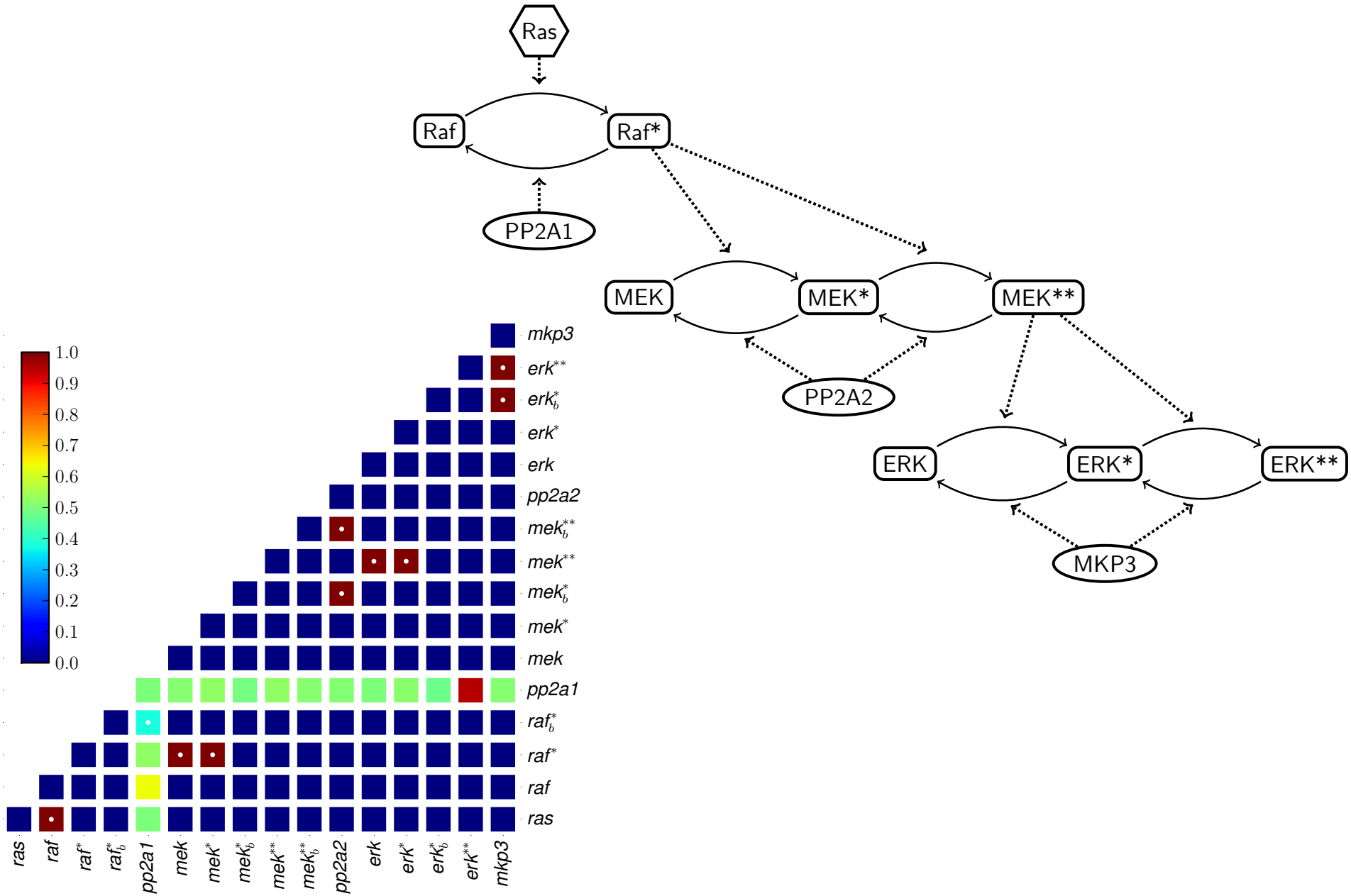




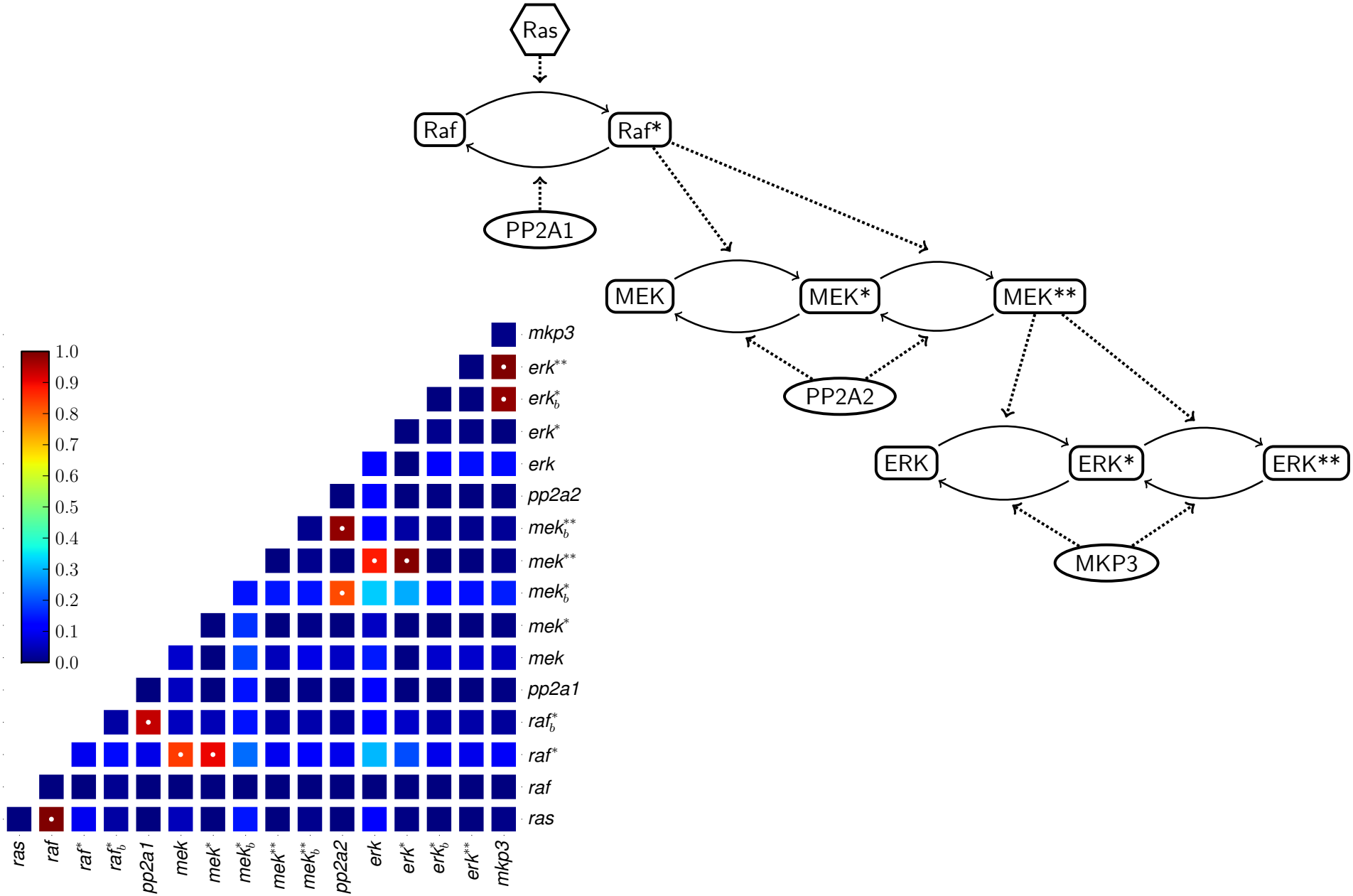
Less Fit Peaks (Left)



Less Fit Peaks (Right)



Advantageous Mutations



Observations

We have been able to explore the complete one-step evolutionary neighbourhood of a MAPK cascade under modifications of site activity.

For this model, we observe:

- Signal transmission has some robustness.
- Switch behaviour is readily accessible.
- Almost all mutations reduce fitness, although many only slightly so.
- A few give improvement against the chosen fitness measure.

Logic for Behaviour in Context

To complement systematic operations for variation in processes, we give a language for classifying the resulting behaviours:

$$P \models b$$

Process P *exhibits behaviour* b

Our language is a temporal logic with real-valued constraints and behaviour in context:

Basic observations	Concentration $[A] > c$, rate of change $[B]' < k$
Logical operators	$b \wedge b'$, $\neg b$, ...
Behaviour over time	$F(b)$, $G(b)$, $b_1 \, U \, b_2$, ...
Time-limited behaviour	$F_t(b)$, ...
Behaviour in context	$(Q \triangleright b)$

$$P \models (Q \triangleright b) \iff (P \parallel Q) \models b$$

Sample Behavioural Classifiers

$$\mathsf{F}([\mathrm{Mek}*] > \mathrm{c})$$

$$\mathsf{G}(([\mathrm{Raf}] > 200) \vee ([\mathrm{Raf}*] > 200))$$

$$\mathsf{G}(\mathsf{F}([\mathrm{KaiC_6}]') > 0.44)$$

$$\mathsf{G}([\mathrm{Prod}] < 5) \wedge (\mathrm{En} \triangleright \mathsf{F}_{\mathrm{t}}([\mathrm{Prod}] > 20))$$

$$(\mathrm{En} \triangleright \mathsf{F}_{\mathrm{t}}([\mathrm{Prod}] > 20)) \wedge (\mathrm{Inhib} \triangleright \mathsf{G}(\neg(\mathrm{En} \triangleright \mathsf{F}([\mathrm{Prod}] > 20))))$$

If En is added then within t seconds the concentration of Prod will rise above 20mM, but if instead Inhib is introduced then from that point on, addition of En will never lead to production of Prod.

Model-Checking Continuous pi

We can check whether process P exhibits behaviour $\mathfrak{b}$ by:

- Compiling P to a collection of ODEs
- Solving numerically to give a trace of species concentrations over time
- Checking whether that trace satisfies $\mathfrak{b}$

However, this approach has limitations:

Precision Indefinite temporal operators like $\mathsf{G}(-)$ and $\mathsf{F}(-)$ cannot always be checked with finite traces. Even for finite time operators $\mathsf{F}_{\mathrm{t}}(-)$, traces are only intermittent.

Cost Checking temporal operators is linear in trace length. But combining with contextual operators $\mathsf{G}(\mathrm{Q} \triangleright -)$ requires computation and traversal of many traces.

Summary

The *continuous pi-calculus* (c-pi) is a process algebra for modelling behaviour and variation in biomolecular systems: *e.g.* enzyme activation and inhibition; circadian clocks; signalling pathways.

Expressions in c-pi represent mixtures of chemical reagents, and can be compiled to conventional ODE models for fast numerical simulation.

With a language of potential changes in c-pi processes we systematically explore evolutionary neighbourhoods of a specific signalling pathway, and observe instances of robustness, neutrality and evolvability.

A complementary temporal logic for behaviour in context gives a language to classify these variations in behaviour.

Conclusion

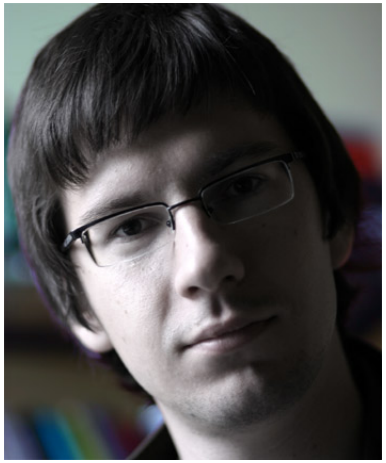
Limitations

- Over-expressiveness of c-pi: stay within the biological
- Artificiality of behaviour modelling within complexes
- Low-count species (DNA) and discrete state transitions

Further Directions

- Explore computational cost of model-checking
- Lazier model-checking algorithms
- Other non-transcriptional clocks; bistable systems
- Hybrid models for discrete states

The Continuous pi-makers



Marek Kwiatkowski
Institute of Integrative Biology
Department of Environmental Sciences
EAWAG & ETH Zürich
http://mareklab.org

Seeking a job in evolutionary aspects of theoretical/ computational/systems biology. Hire him, he's excellent.

Chris Banks
Laboratory for Foundations of Computer Science
School of Informatics
The University of Edinburgh
http://banks.ac

PhD student 2010–

References


Kwiatkowski and Stark.
On Executable Models of Molecular Evolution. In *Proc. 8th International Workshop on Computational Systems Biology WCSB 2011*, pp. 105–108.


Kwiatkowski and 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


Kwiatkowski.
A Formal Computational Framework for the Study of Molecular Evolution
PhD Dissertation, University of Edinburgh, December 2010.


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