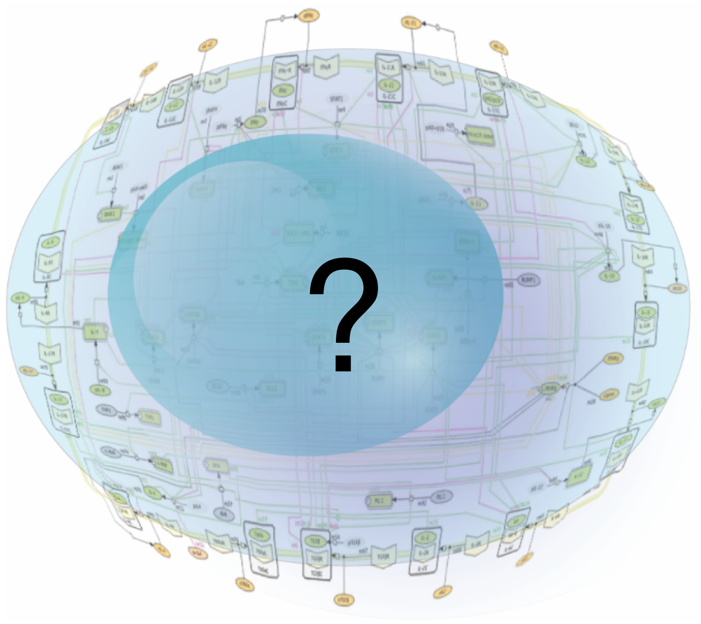
Evaluation of Parallel Tempering to Accelerate Bayesian Parameter Estimation in Systems Biology

Sanjana Gupta, Liam Hainsworth, Justin S. Hogg,
Robin E.C. Lee, James R. Faeder

Department of Computational and Systems Biology,
University of Pittsburgh

A key question in Systems Biology: How do cells signal?

Environmental cues →

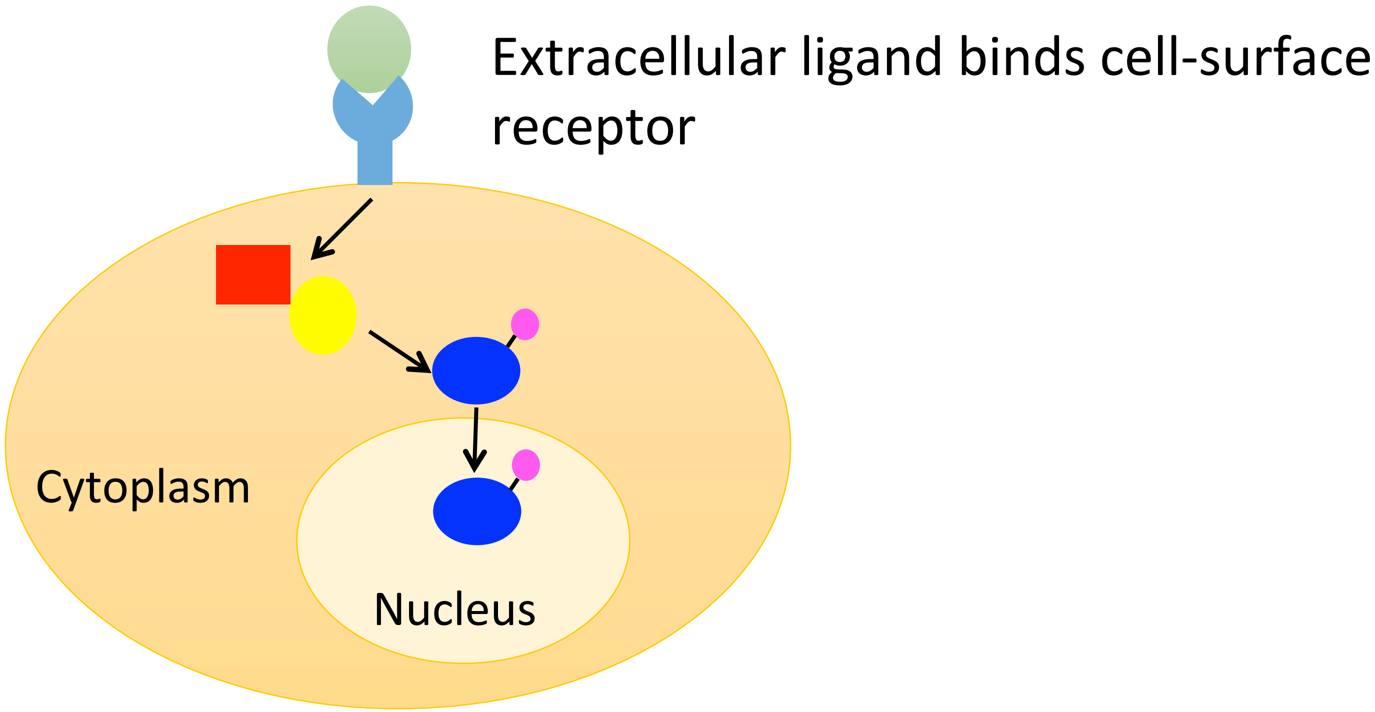


Cell

→ Cell death
→ Cell survival
→ Cell proliferation
→ Cell differentiation
→ Cell migration

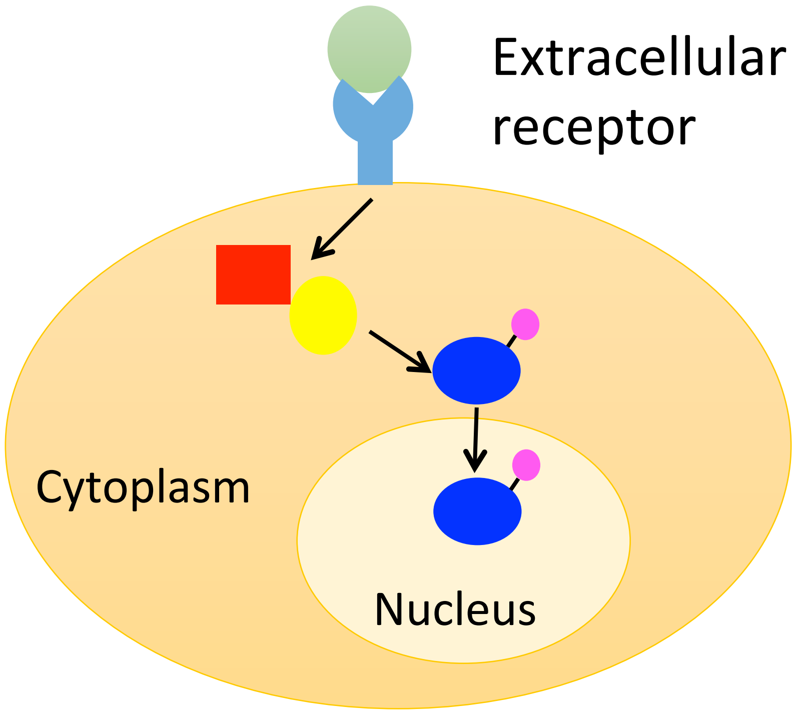
A key question in Systems Biology: How do cells signal?

Models to understand cell signaling systems

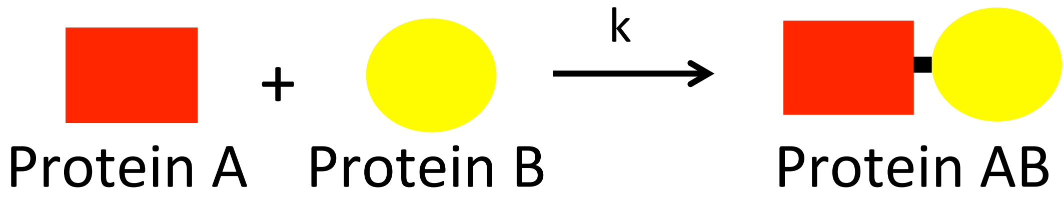


A key question in Systems Biology: How do cells signal?

Models to understand cell signaling systems

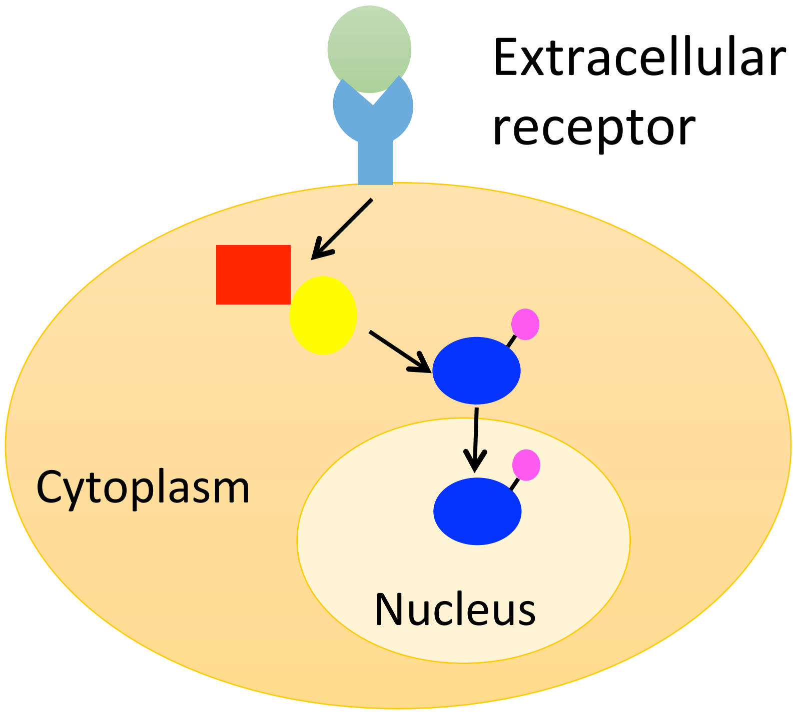


Extracellular ligand binds cell-surface receptor

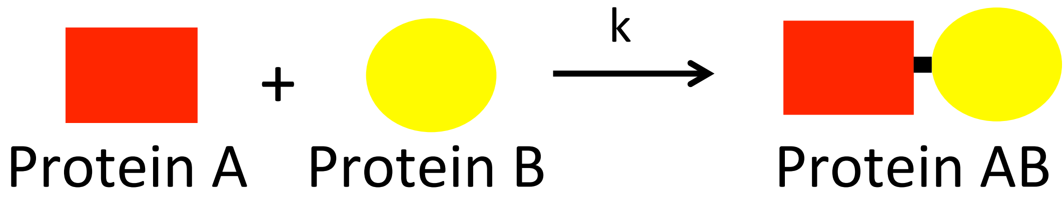


A key question in Systems Biology: How do cells signal?

Models to understand cell signaling systems



Extracellular ligand binds cell-surface receptor

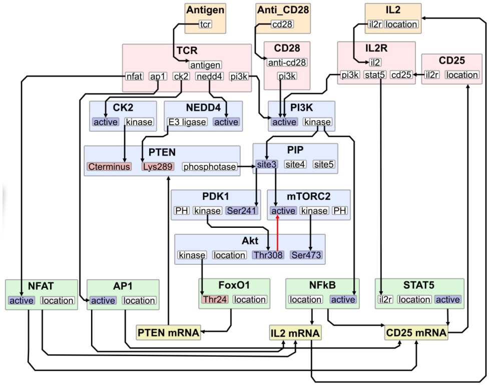


$$\frac{d[\text{protein AB}]}{dt} = k[\text{protein A}][\text{protein B}]$$

Model parameters: k, Total protein A, Total protein B

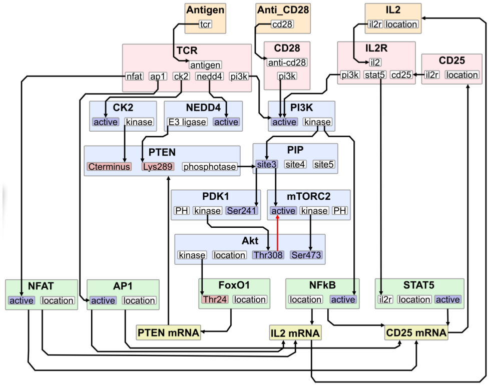
Closed loop Systems Biology

1. Construct model based on **known** and **hypothesized** interactions.

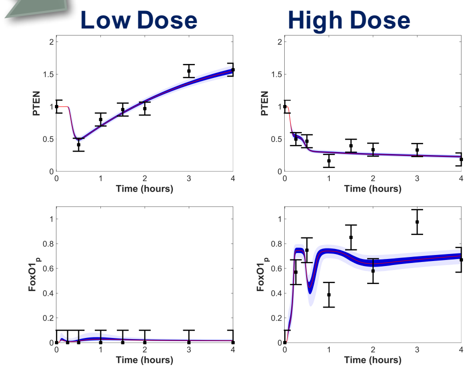


Closed loop Systems Biology

1. Construct model based on **known** and **hypothesized** interactions.

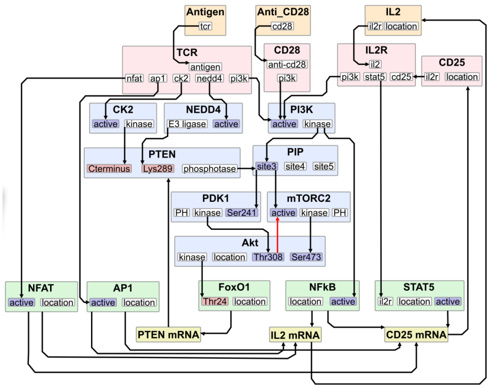


2. Calibrate model parameters to available data.

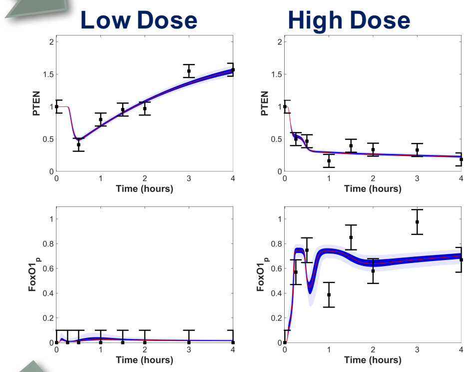


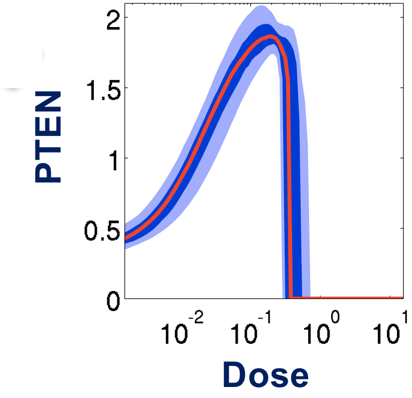
Closed loop Systems Biology

1. Construct model based on **known** and **hypothesized** interactions.



2. Calibrate model parameters to available data.

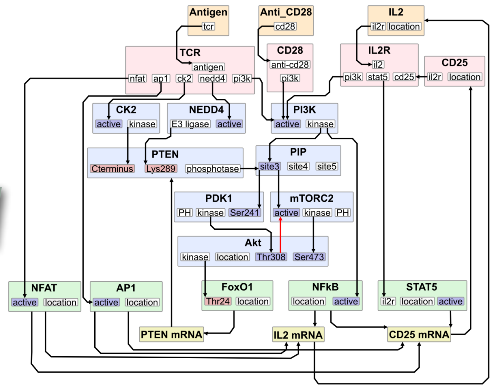




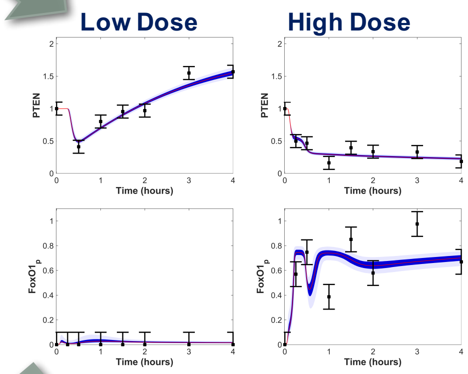
3. Analyze model to find novel behaviors.

Closed loop Systems Biology

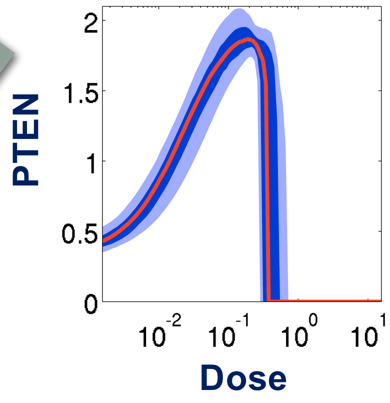
1. Construct model based on **known** and **hypothesized** interactions.



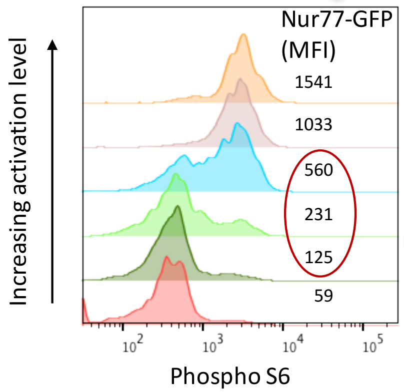
2. Calibrate model parameters to available data.



3. Analyze model to find novel behaviors.

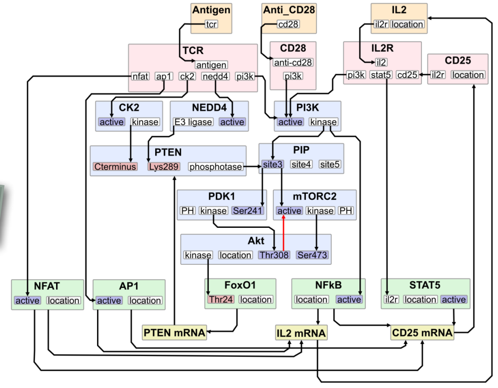


4. Perform experiments to test predictions.

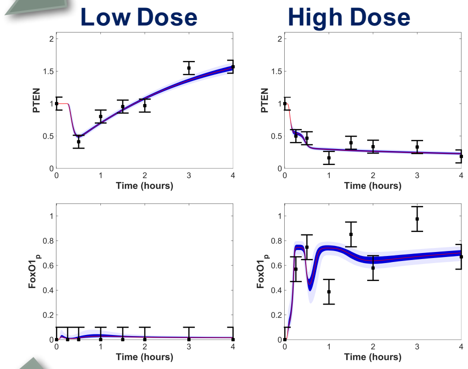


Closed loop Systems Biology

1. Construct model based on **known** and **hypothesized** interactions.

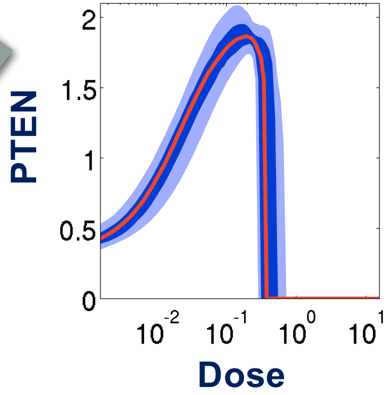


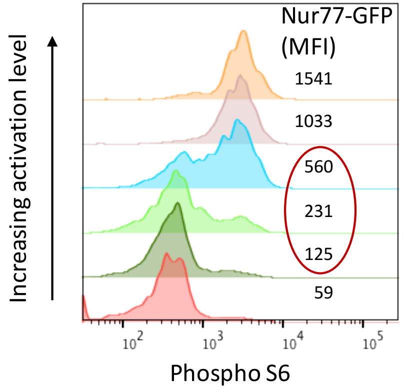
2. Calibrate model parameters to available data.



Repeat as needed

3. Analyze model to find novel behaviors.



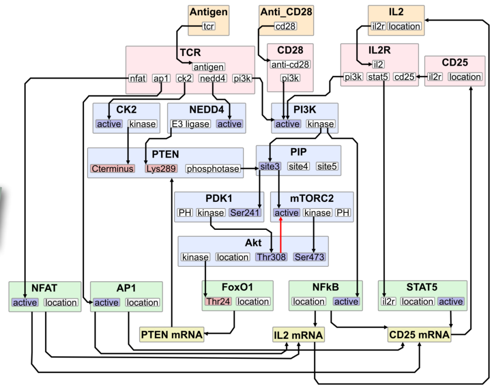


4. Perform experiments to test predictions.

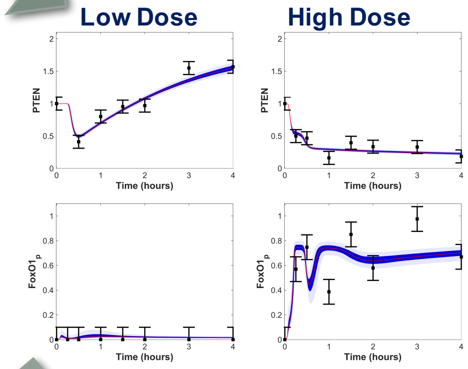
Closed loop Systems Biology

Computationally expensive

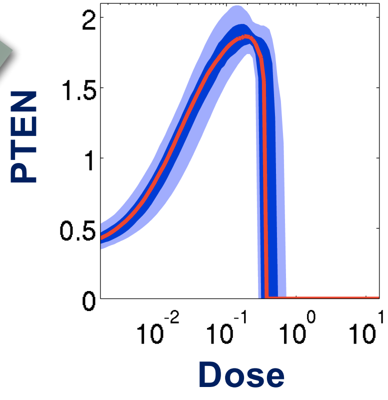
1. Construct model based on **known** and **hypothesized** interactions.



2. Calibrate model parameters to available data.

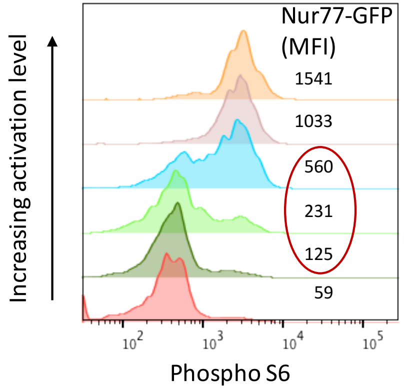


3. Analyze model to find novel behaviors.



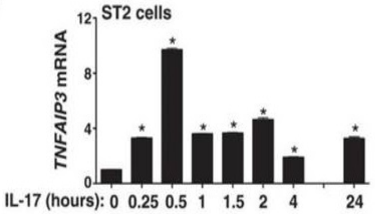
Repeat as needed

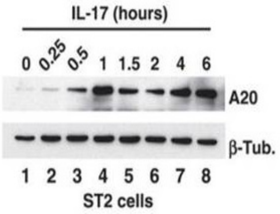
4. Perform experiments to test predictions.

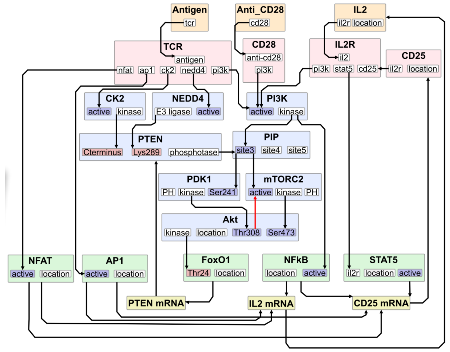


Parameter estimation as energy minimization

Experimental Data



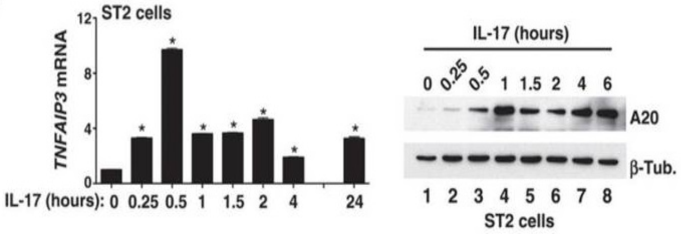


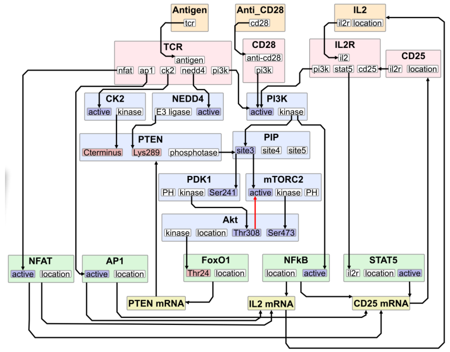


Model

Parameter estimation as energy minimization

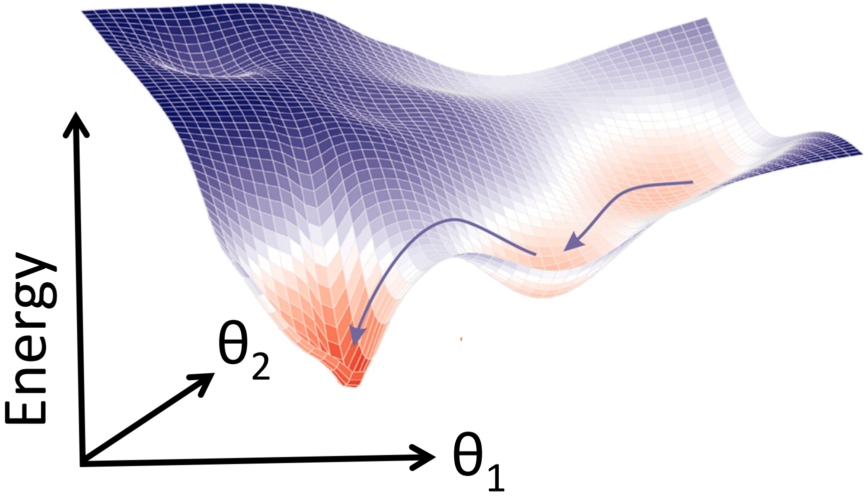
Experimental Data





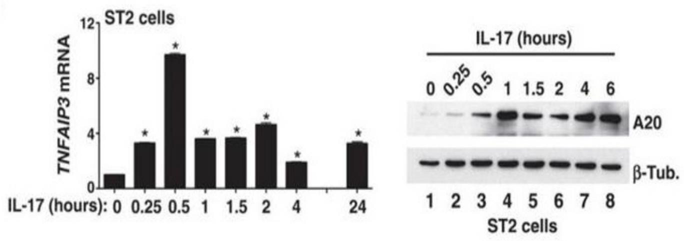
Model

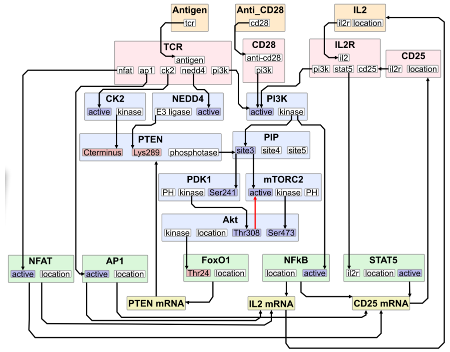
Energy landscape



Parameter estimation as energy minimization

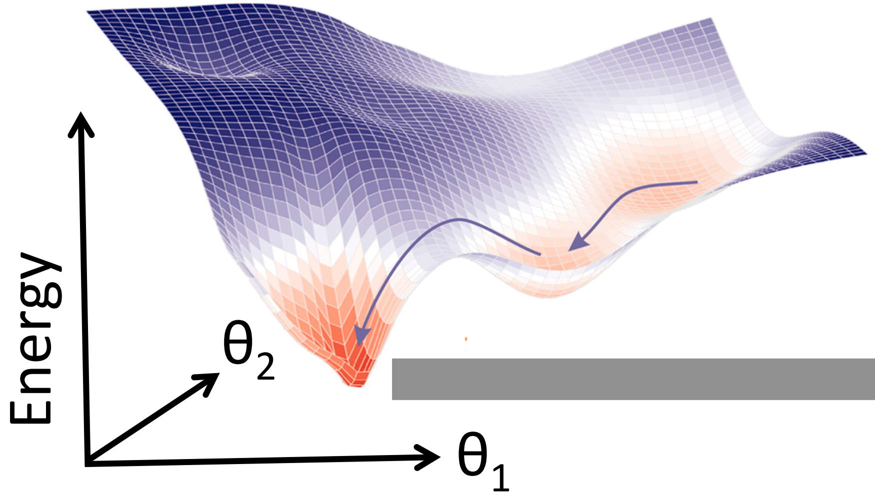
Experimental Data

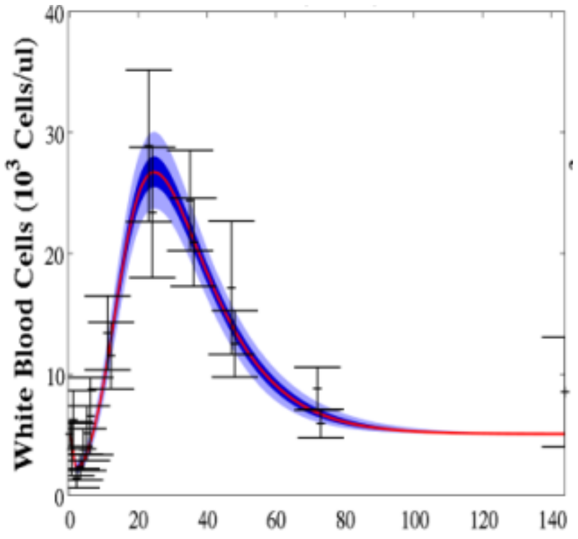




Model

Energy landscape

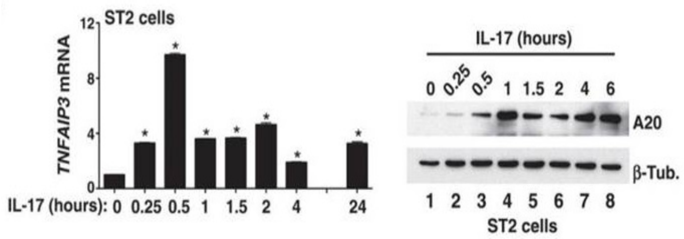




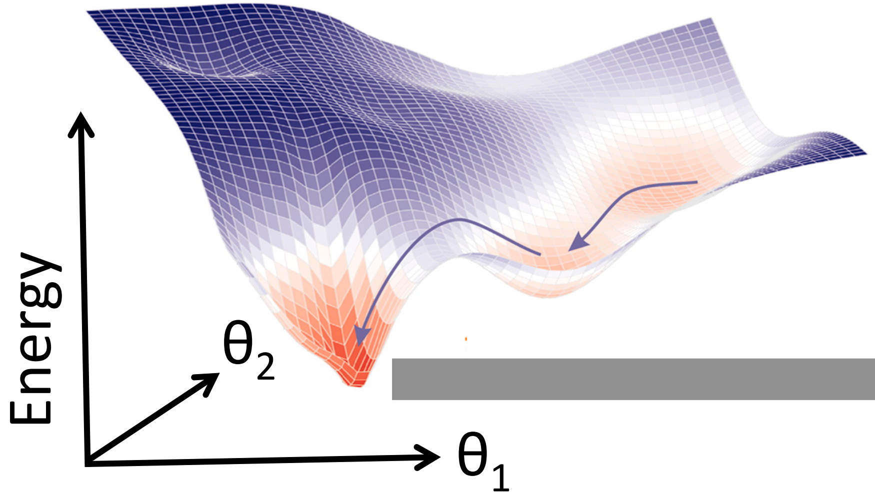
Energy minimum corresponds to best fit parameters

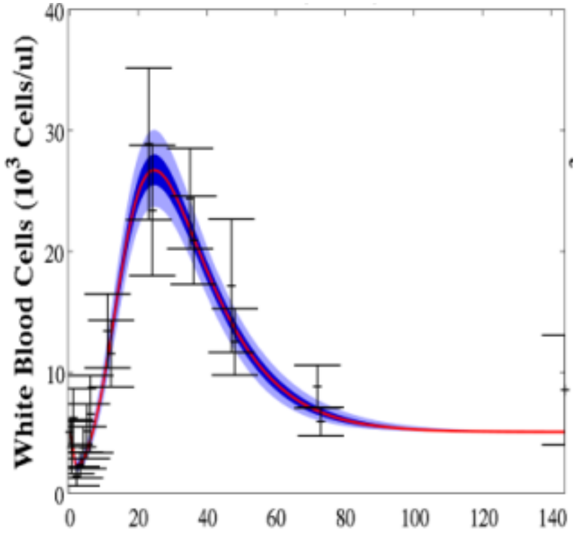
Parameter estimation as energy minimization

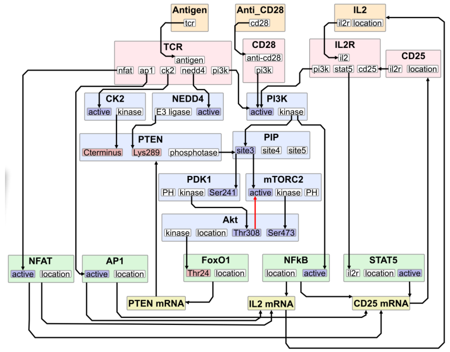
Experimental Data



Energy landscape





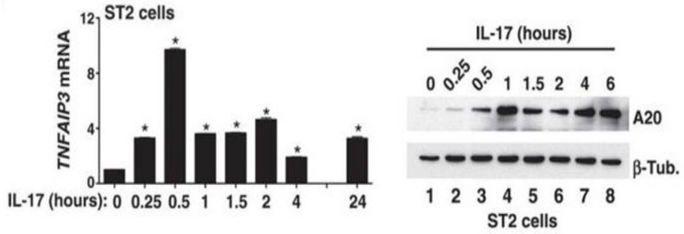


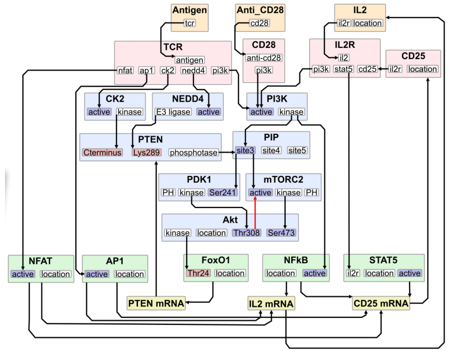
Model

Goal: Efficiently characterize the energy landscape

Parameter estimation as energy minimization

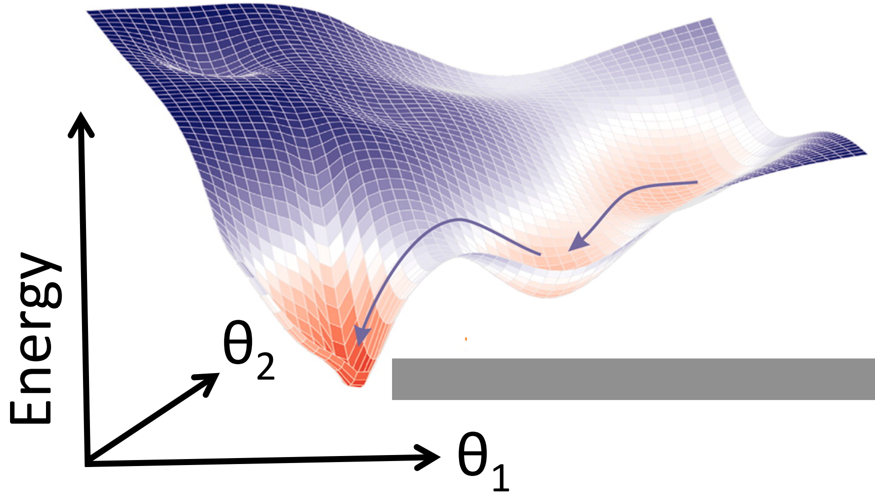
Experimental Data

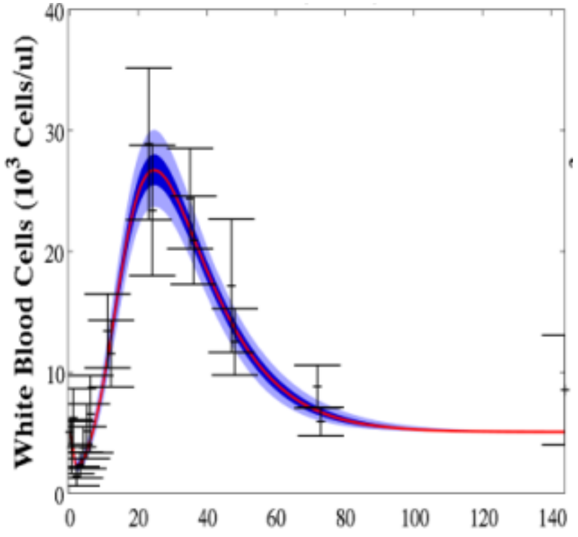




Model

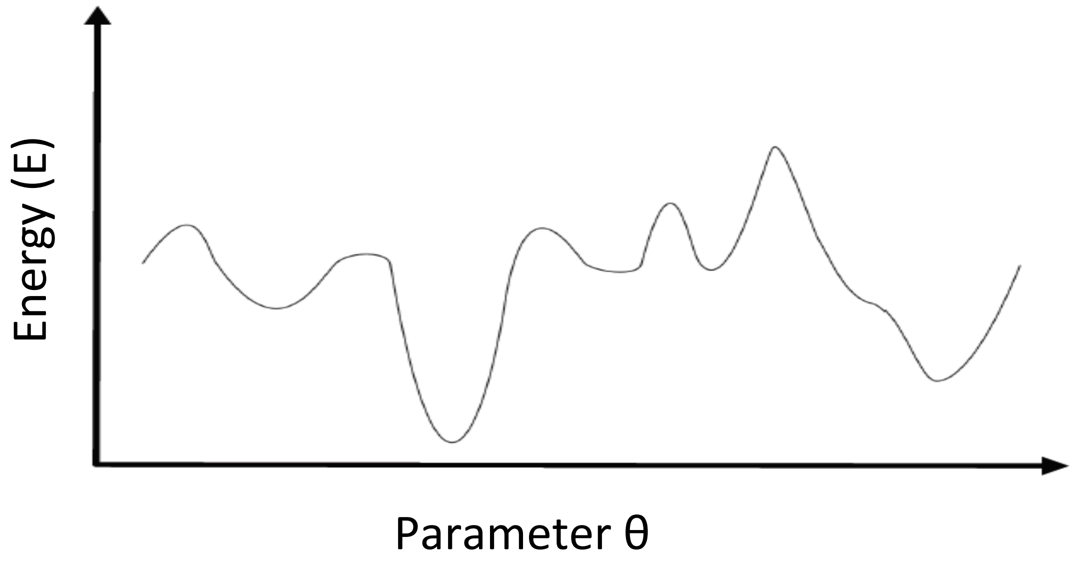
Energy landscape



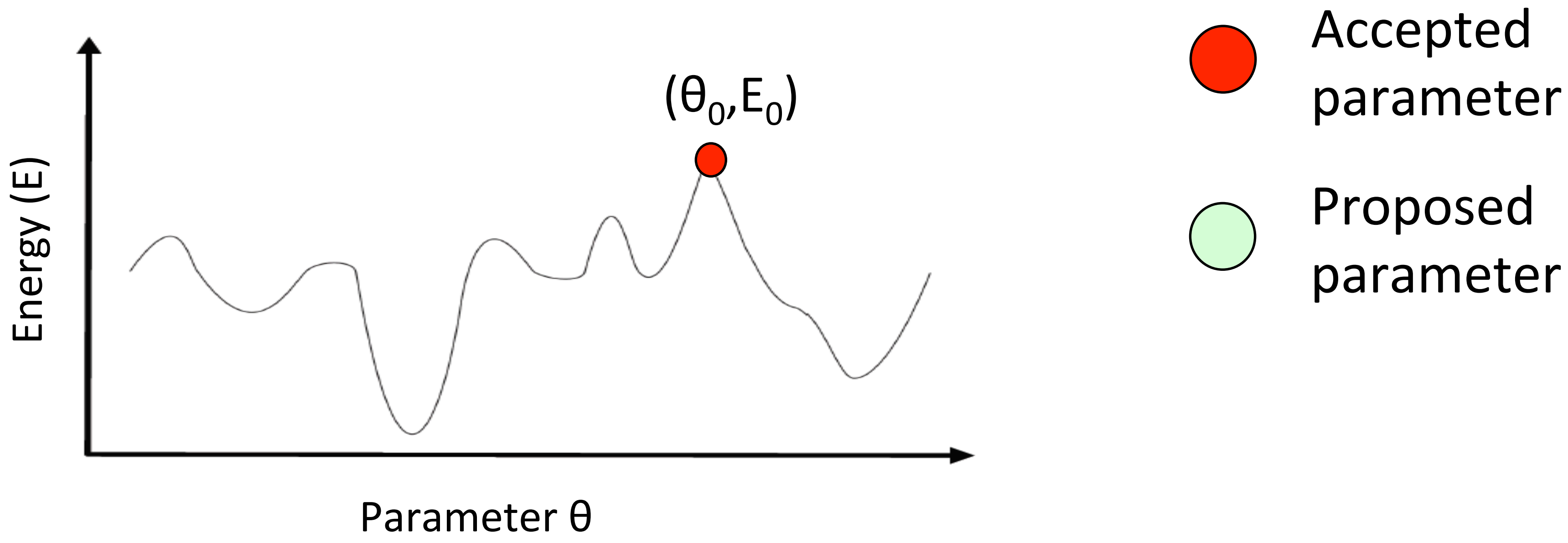


Common approach: Bayesian parameter estimation with Markov Chain Monte Carlo

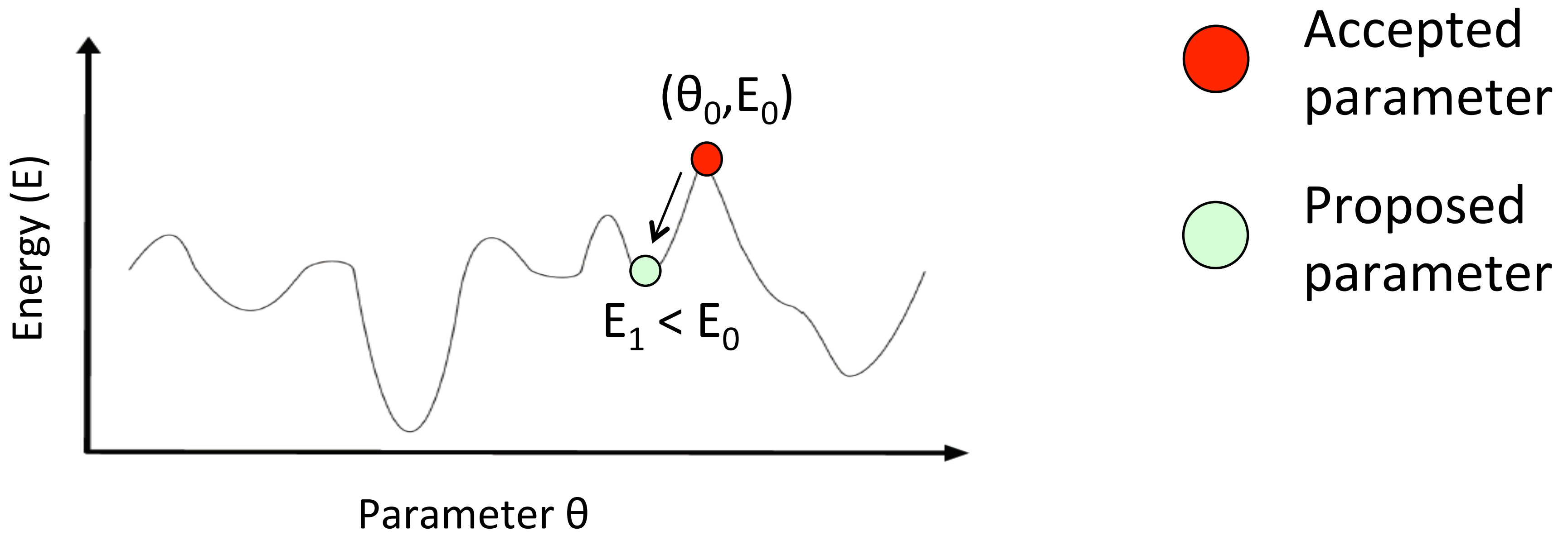
Conventional MCMC: Metropolis-Hastings (MH)



Conventional MCMC: Metropolis-Hastings (MH)

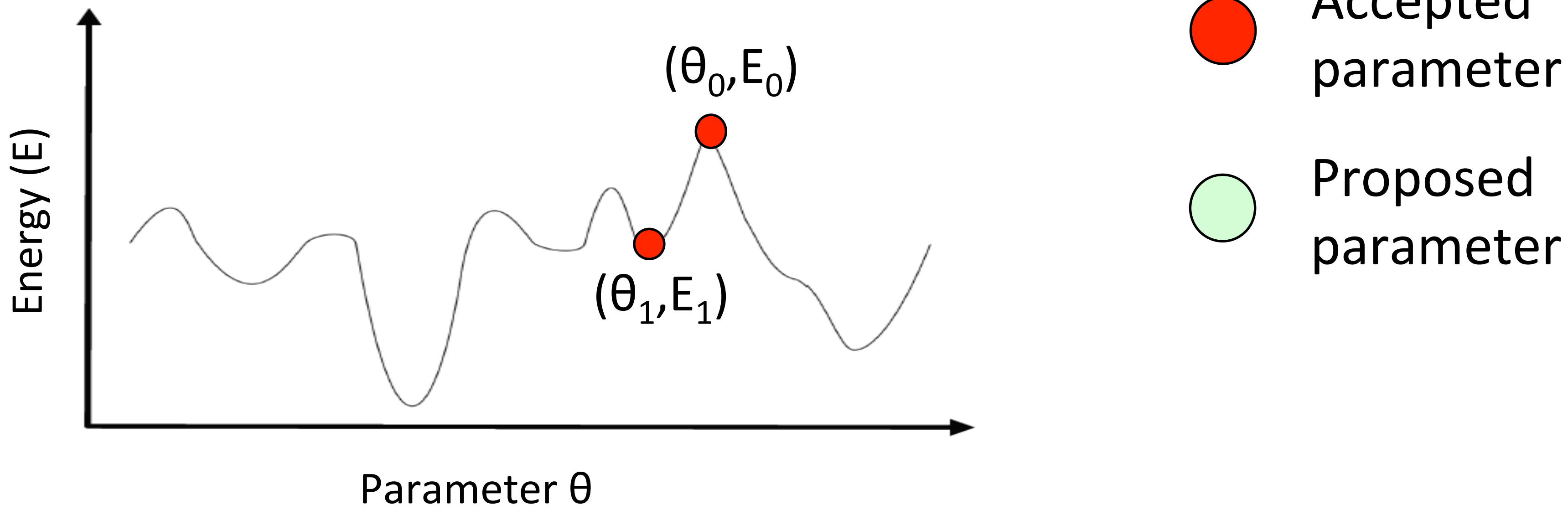


Conventional MCMC: Metropolis-Hastings (MH)



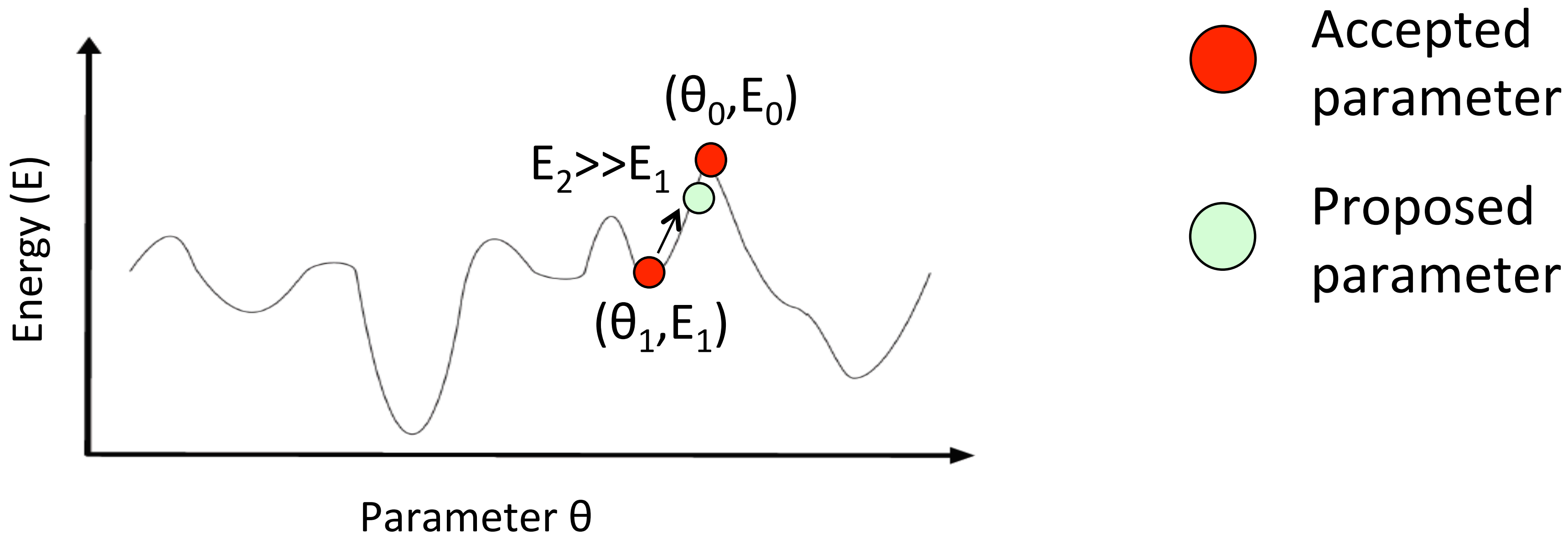
- Accept favorable moves with probability 1

Conventional MCMC: Metropolis-Hastings (MH)



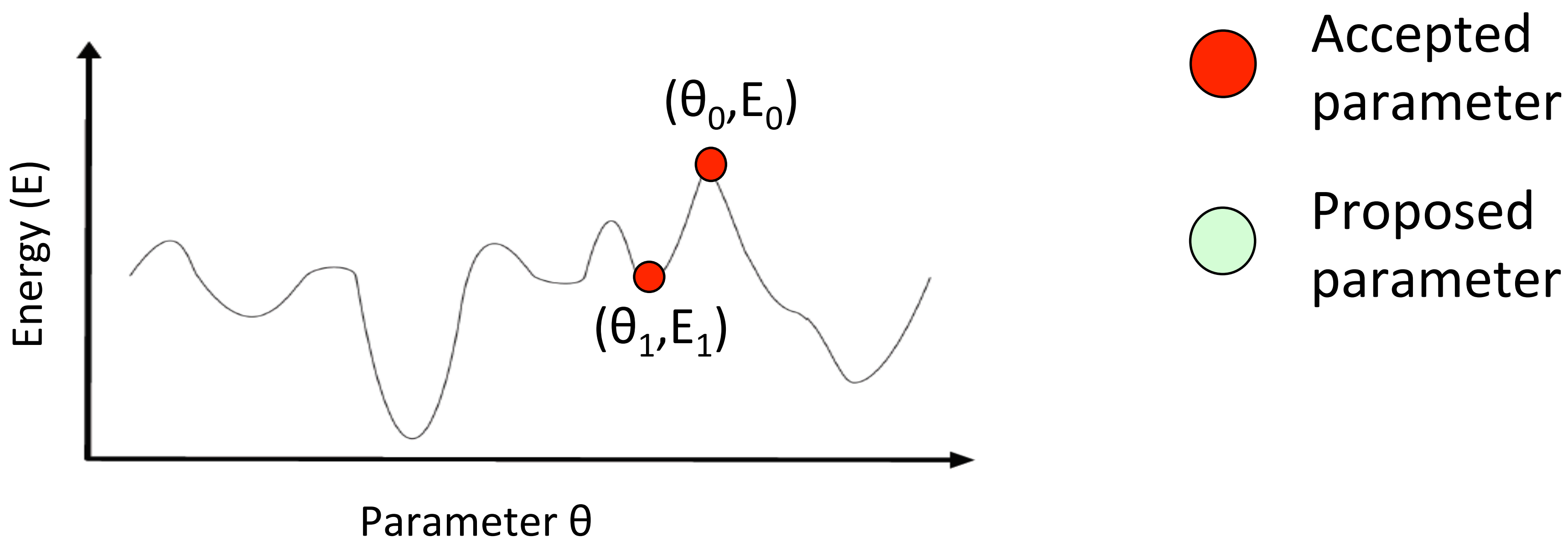
- Accept favorable moves with probability 1

Conventional MCMC: Metropolis-Hastings (MH)



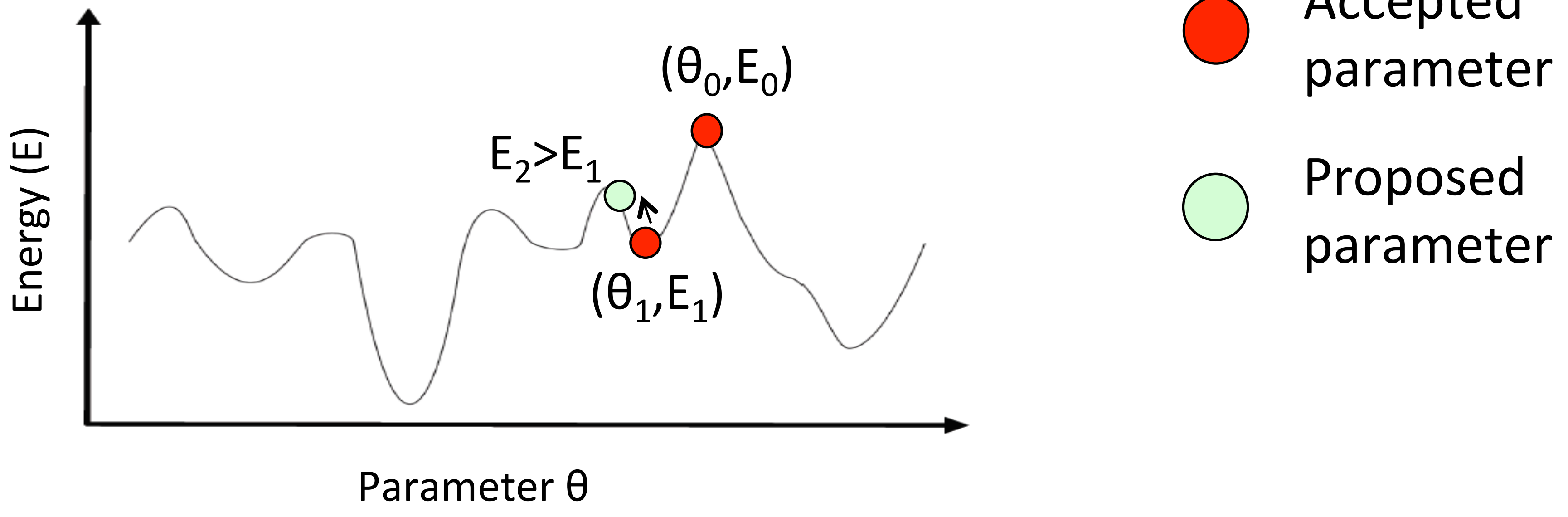
- Accept favorable moves with probability 1
- Accept unfavorable moves with probability: $e^{-\beta\Delta E}$, β=1

Conventional MCMC: Metropolis-Hastings (MH)



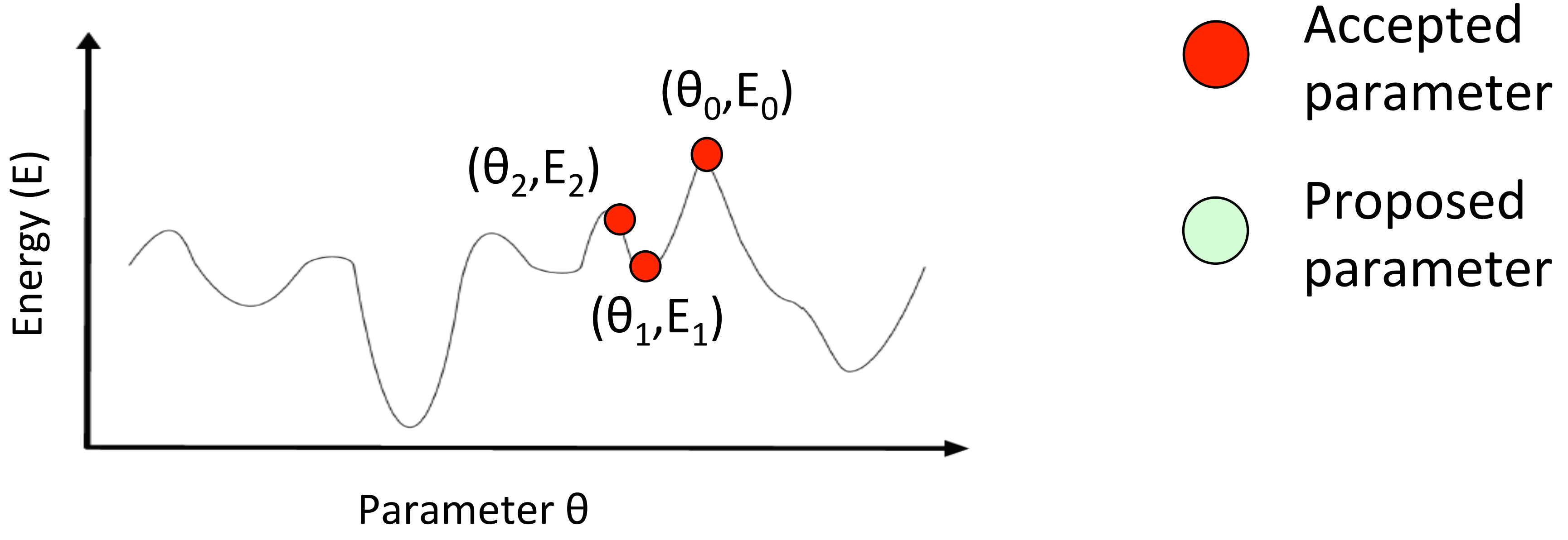
- Accept favorable moves with probability 1
- Accept unfavorable moves with probability: $e^{-\beta \Delta E}$, β=1

Conventional MCMC: Metropolis-Hastings (MH)



- Accept favorable moves with probability 1
- Accept unfavorable moves with probability: $e^{-\beta\Delta E}$, β=1

Conventional MCMC: Metropolis-Hastings (MH)



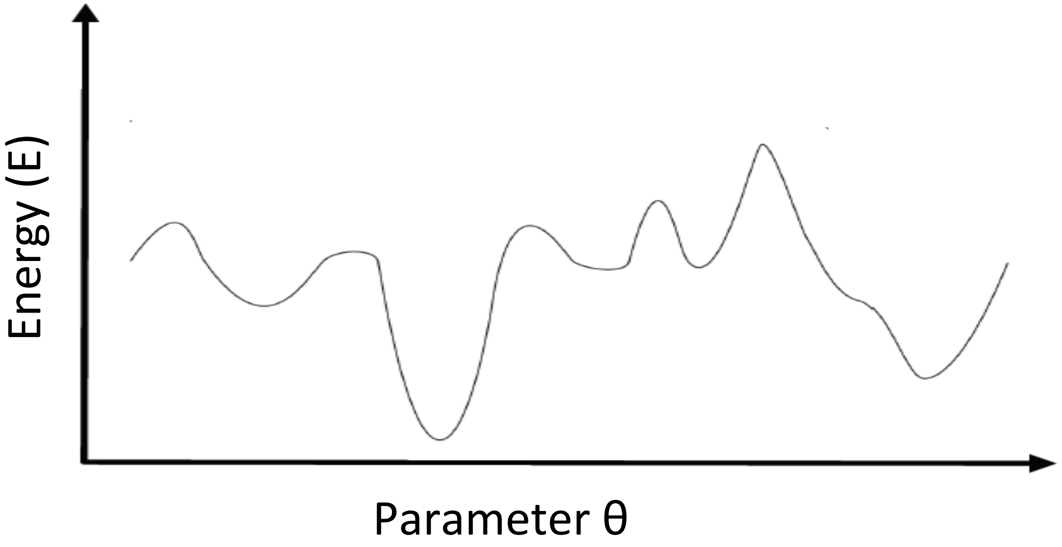
- Accept favorable moves with probability 1
- Accept unfavorable moves with probability: $e^{-\beta \Delta E}$, β=1

Parallel Tempering: an accelerated sampling method

- Commonly used in molecular dynamics simulation
- Sparsely used in systems biology

Parallel Tempering: an accelerated sampling method

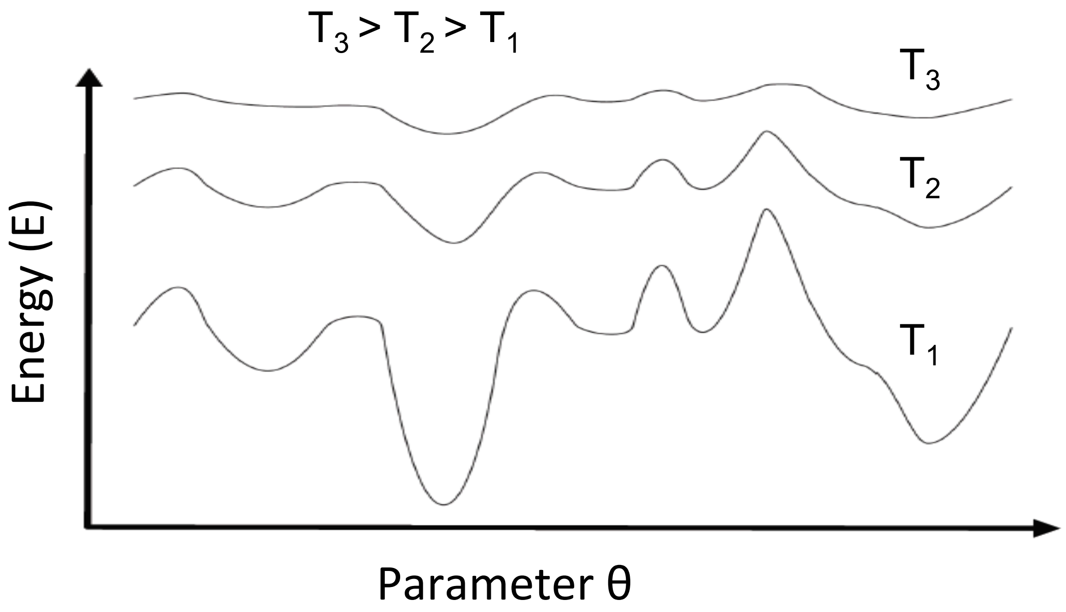
- Commonly used in molecular dynamics simulation
- Sparsely used in systems biology



Parallel Tempering: an accelerated sampling method

- Commonly used in molecular dynamics simulation
- Sparsely used in systems biology

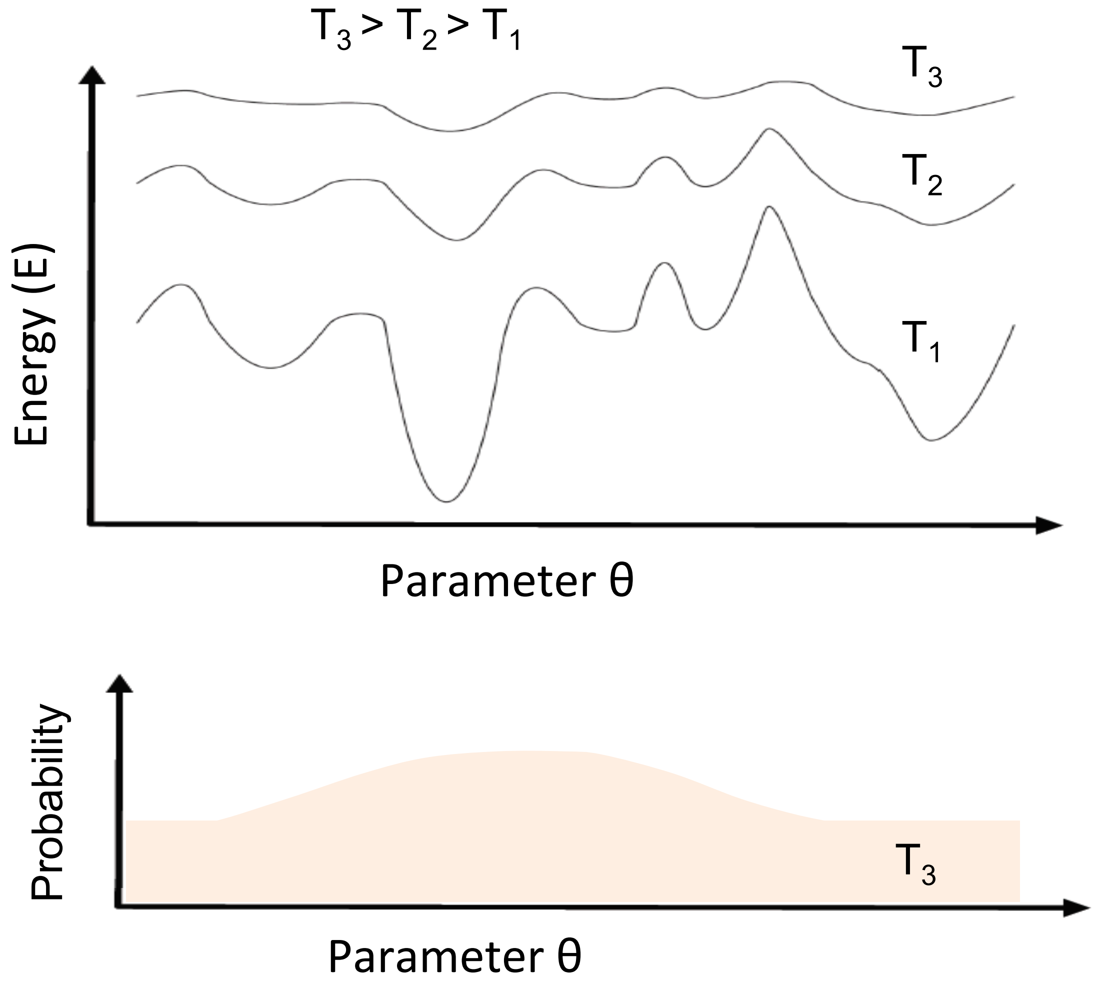
Chains with different temperatures run in parallel



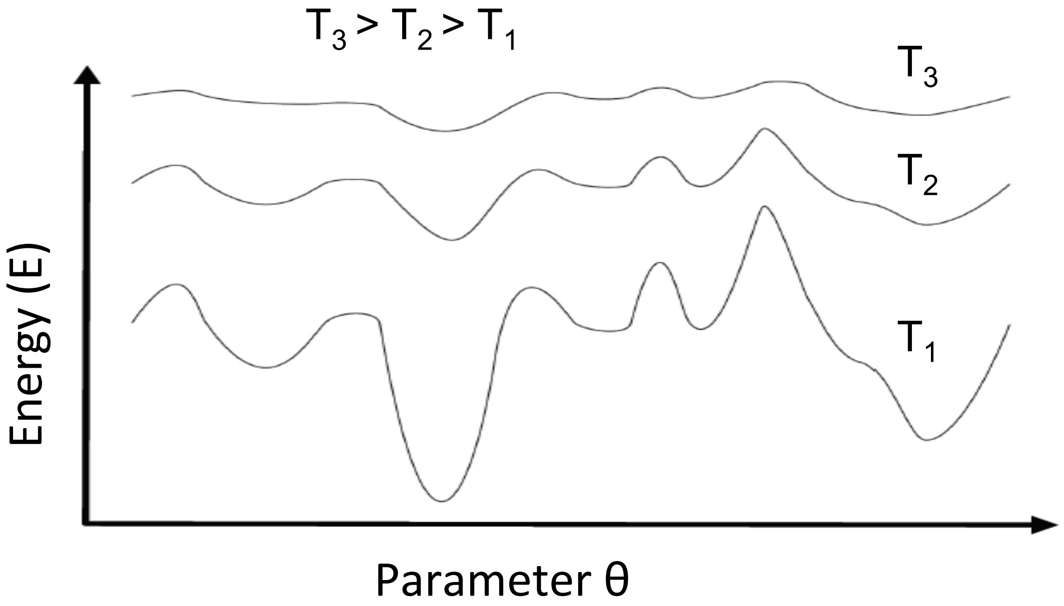
Acceptance probability:

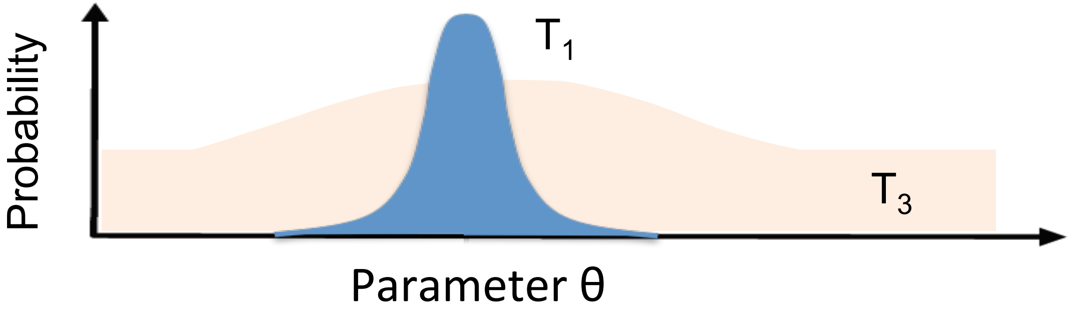
$e^{-\beta\Delta E}$, $\beta \propto 1/T$

Parallel Tempering: an accelerated sampling method



Parallel Tempering: an accelerated sampling method





Performance evaluation of Parallel Tempering (PT)

Performance evaluation of Parallel Tempering (PT)

- 6 increasingly complex biological models

Performance evaluation of Parallel Tempering (PT)

- 6 increasingly complex biological models
- Comparison with Metropolis-Hastings (MH)

Performance evaluation of Parallel Tempering (PT)

- 6 increasingly complex biological models
- Comparison with Metropolis-Hastings (MH)
- Comparison with Approximate Bayesian Computation (ABC)

Performance evaluation of Parallel Tempering (PT)

- 6 increasingly complex biological models
- Comparison with Metropolis-Hastings (MH)
- Comparison with Approximate Bayesian Computation (ABC)
- Model reduction using parallel tempering

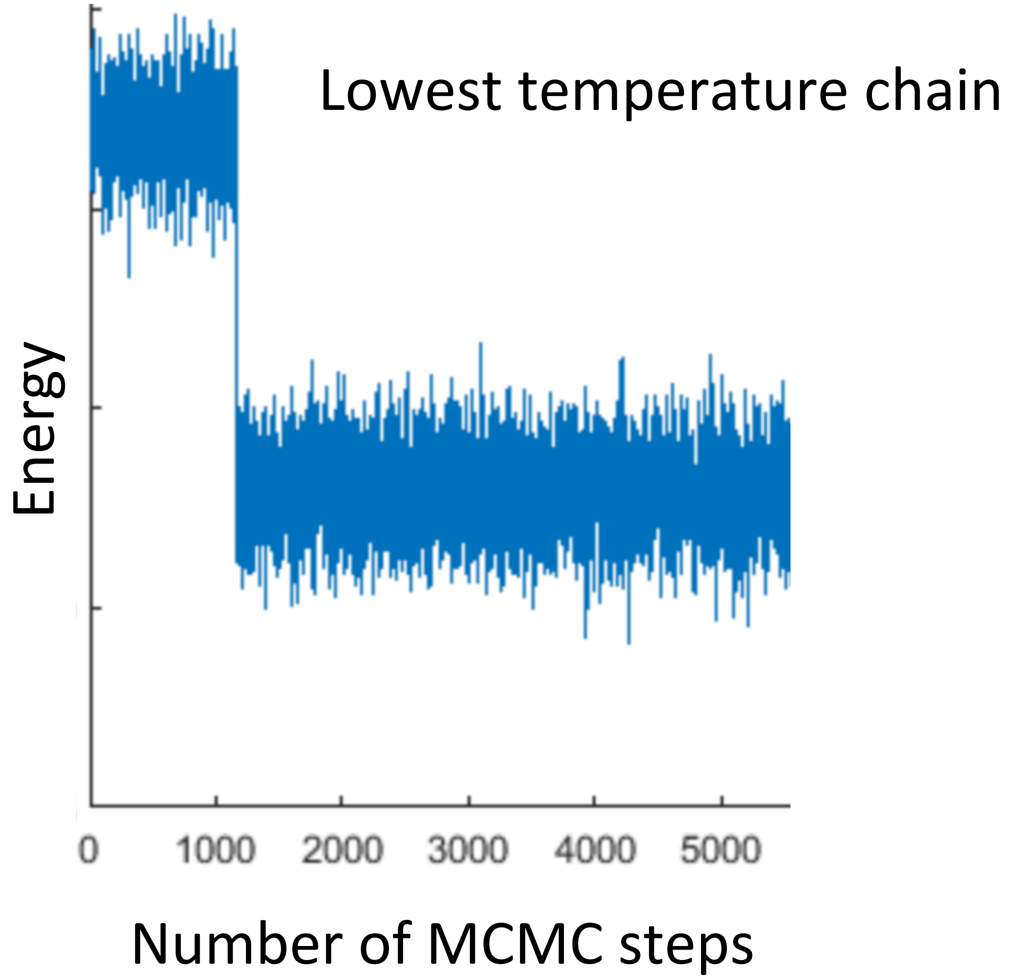
Performance metrics used to compare algorithms

1. Convergence time

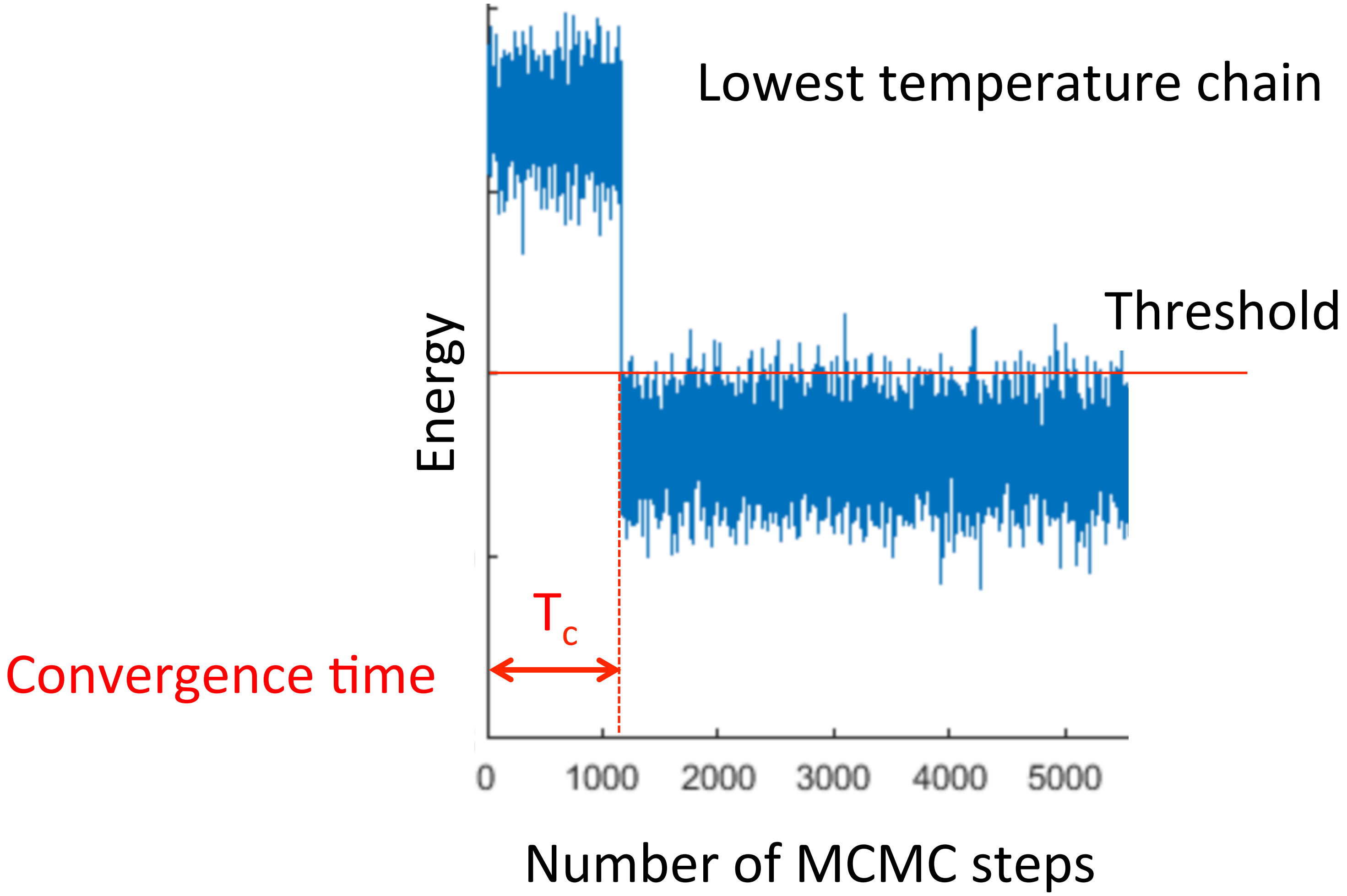
2. Sampling efficiency

3. Quality of fit under constrained computational resource

Performance metric 1: Convergence time



Performance metric 1: Convergence time



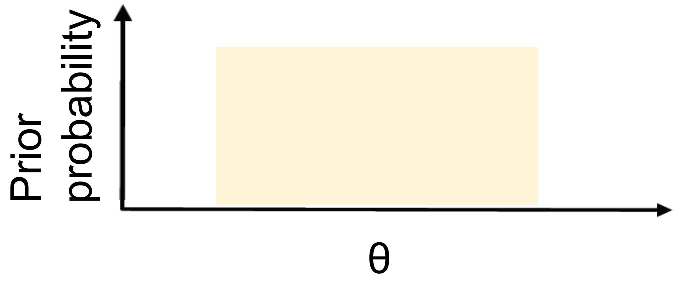
Performance metric 2: Sampling efficiency

Performance metric 2: Sampling efficiency

- Sampling efficiency
  - θ is a dummy parameter that does not contribute to the model output
  - The distribution of θ w.r.t the model output is uniform

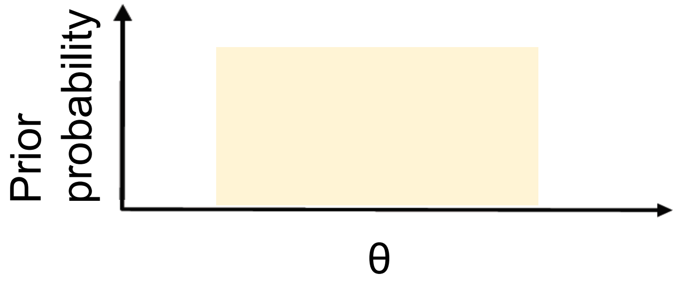
Performance metric 2: Sampling efficiency

- Sampling efficiency
  - θ is a dummy parameter that does not contribute to the model output
  - The distribution of θ w.r.t the model output is uniform



Performance metric 2: Sampling efficiency

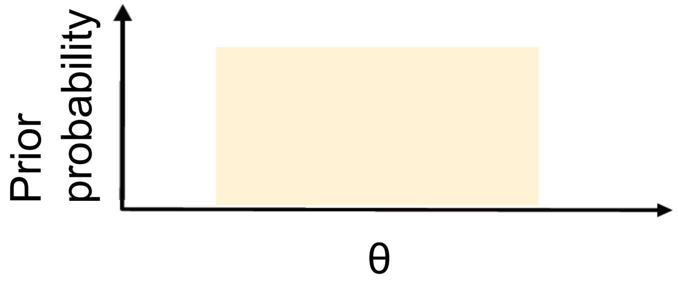
- Sampling efficiency
  - θ is a dummy parameter that does not contribute to the model output
  - The distribution of θ w.r.t the model output is uniform



Ideal sampling

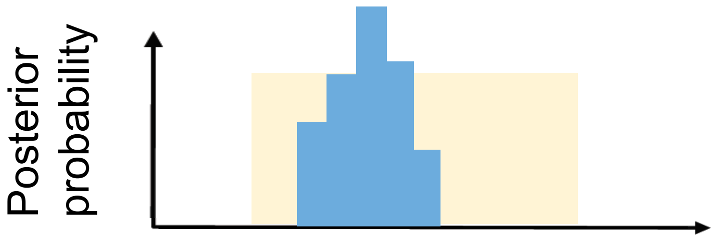
Performance metric 2: Sampling efficiency

- Sampling efficiency
    - θ is a dummy parameter that does not contribute to the model output
    - The distribution of θ w.r.t the model output is uniform



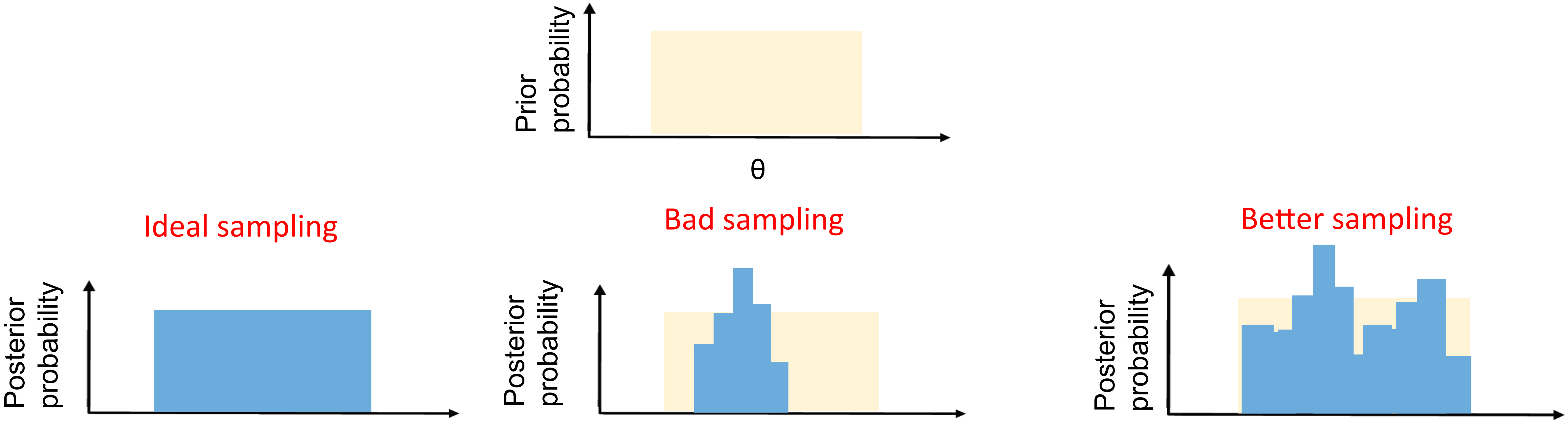
Ideal sampling

Bad sampling



Performance metric 2: Sampling efficiency

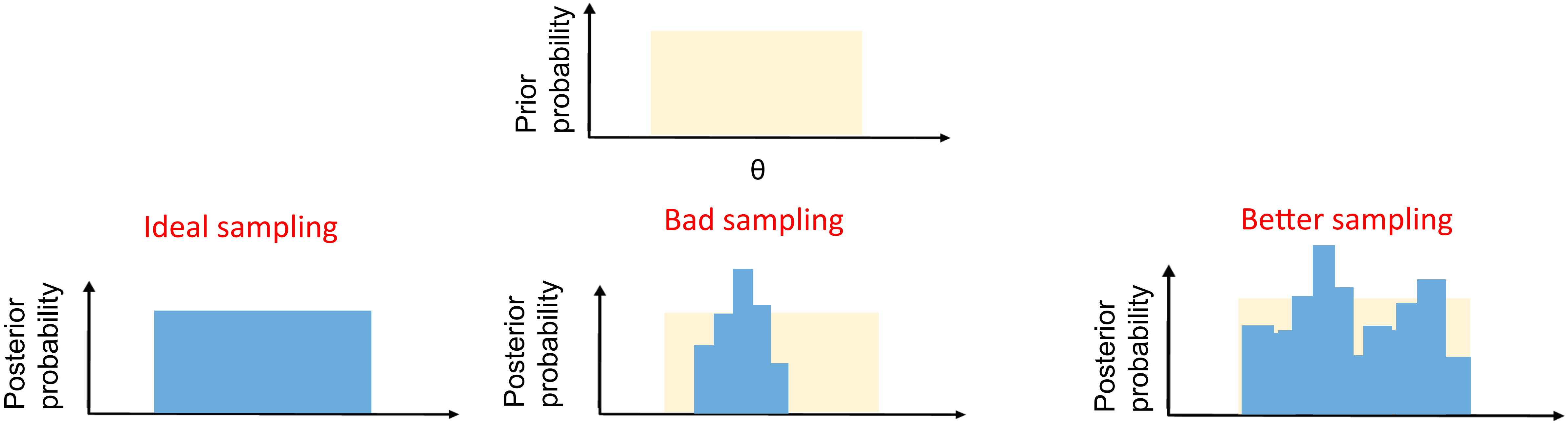
- Sampling efficiency
    - θ is a dummy parameter that does not contribute to the model output
    - The distribution of θ w.r.t the model output is uniform



Performance metric 2: Sampling efficiency

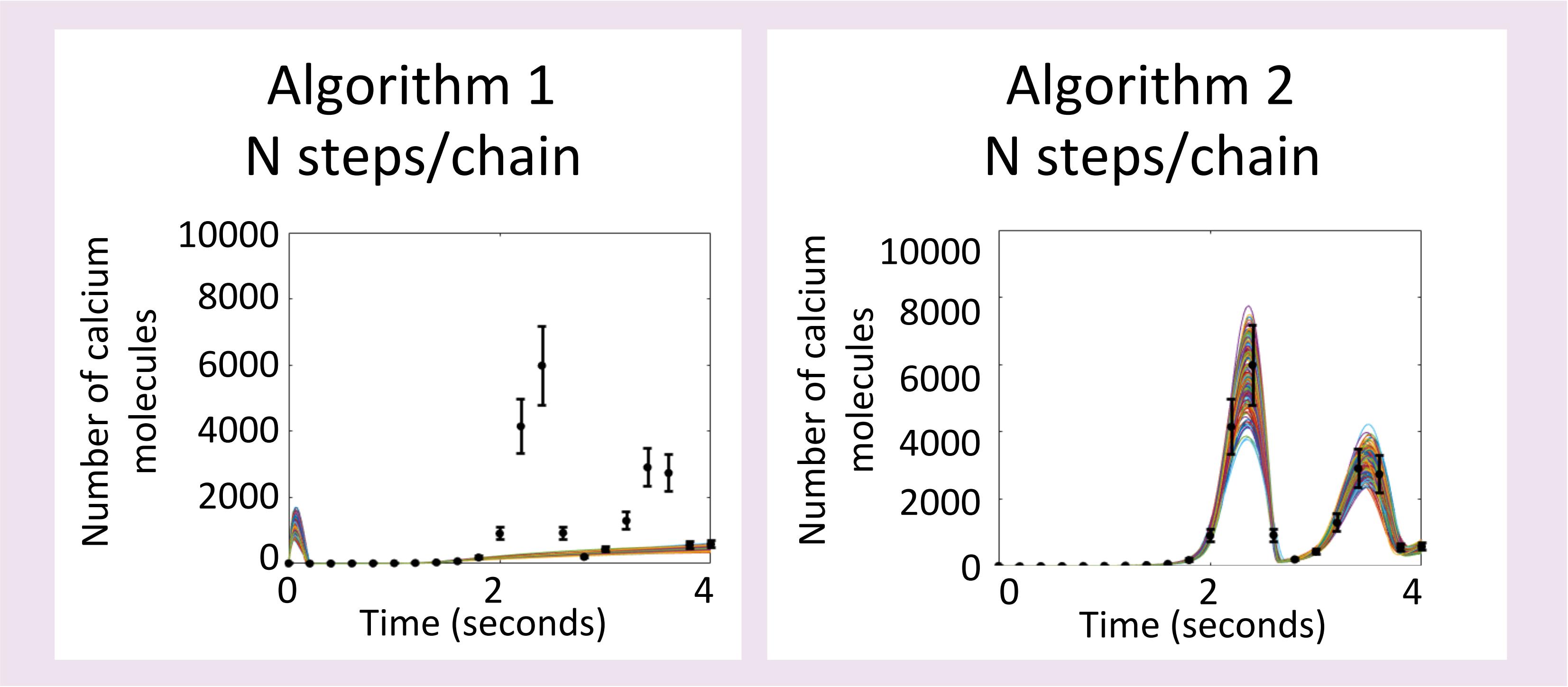
- Sampling efficiency

Sampling efficiency = Range of posterior/ Range of prior, for a dummy parameter θ



Performance metric 3: Quality of fit with fixed budget

How far can you get with fixed computational resources?



Case 1: A 3-parameter model with constrained parameter relationships

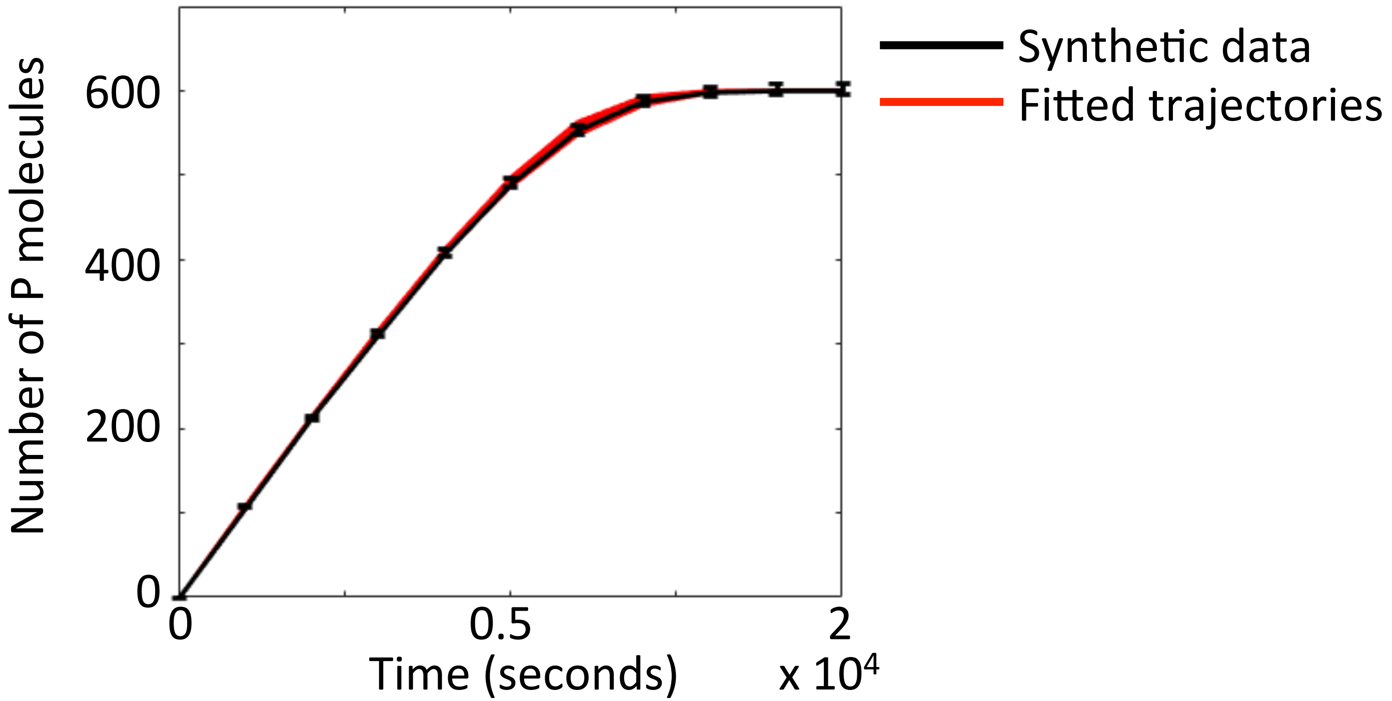
The Michaelis-Menten model

$$E + S \underset{k_r}{\overset{k_f}{\rightleftharpoons}} ES \overset{k_{cat}}{\rightarrow} E + P$$

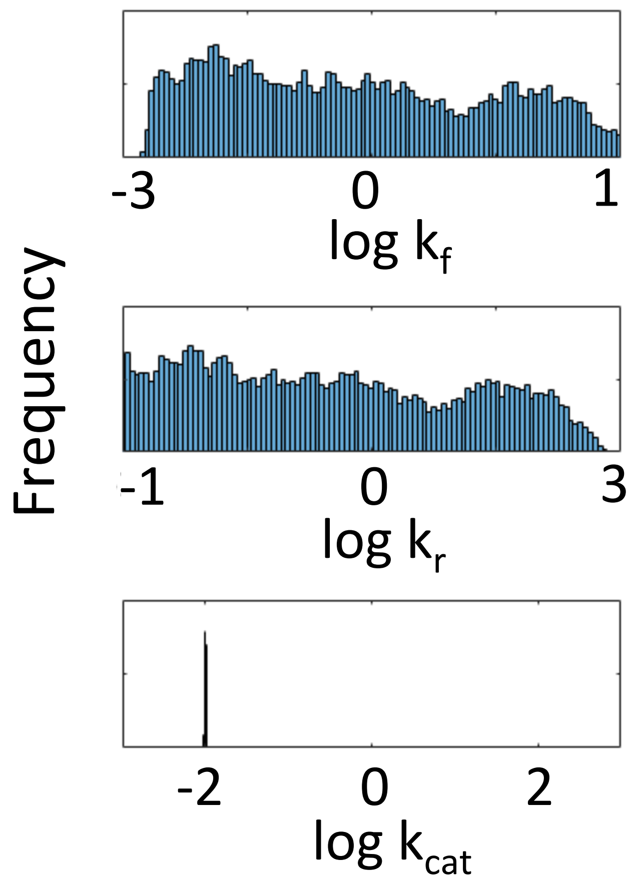
Case 1: A 3-parameter model with constrained parameter relationships

The Michaelis-Menten model

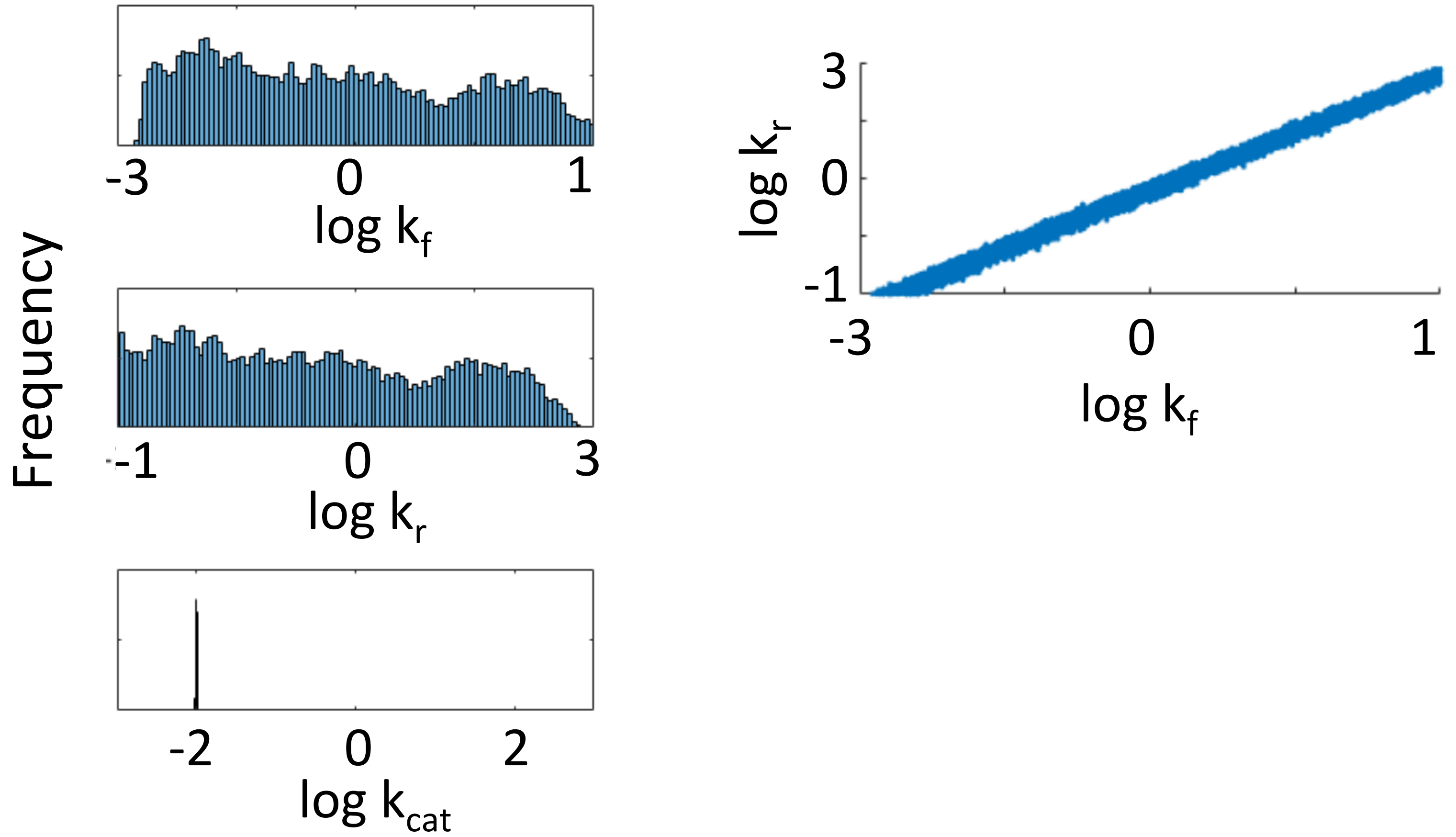
$$E + S \underset{k_r}{\overset{k_f}{\rightleftharpoons}} ES \overset{k_{cat}}{\rightarrow} E + P$$



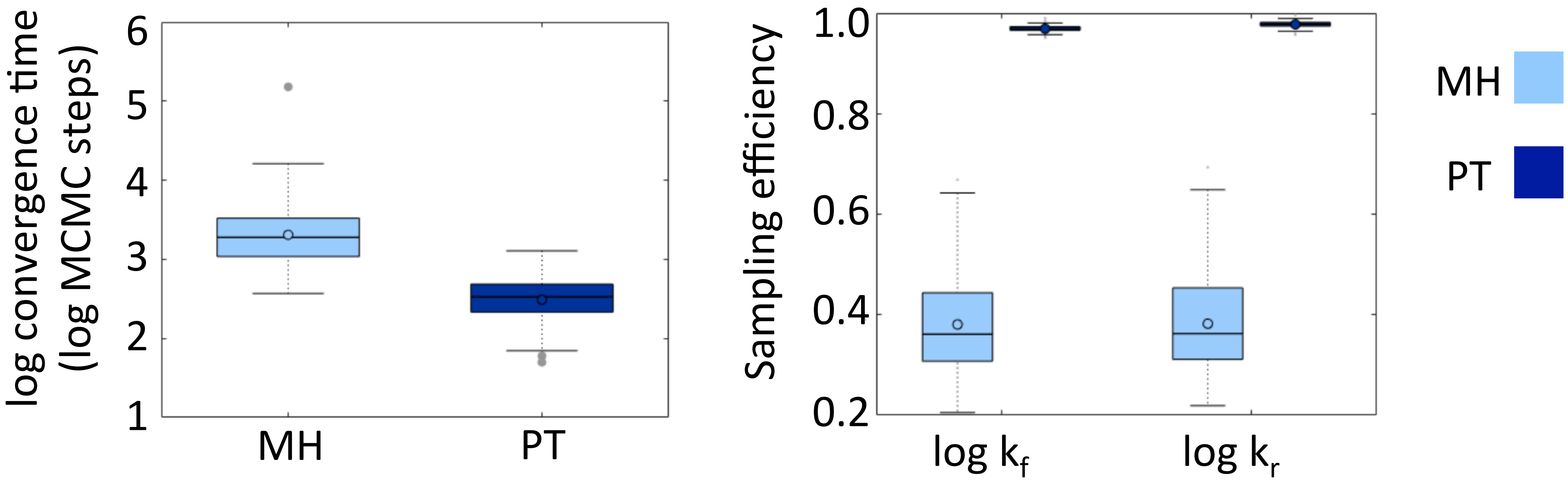
Identifying constrained parameter relationships in the Michaelis-Menten model



Identifying constrained parameter relationships in the Michaelis-Menten model



Performance comparison for 3-parameter Michaelis-Menten model



Comparing MCMC methods with ABC

What is Approximate Bayesian Computation?

Comparing MCMC methods with ABC

What is Approximate Bayesian Computation?

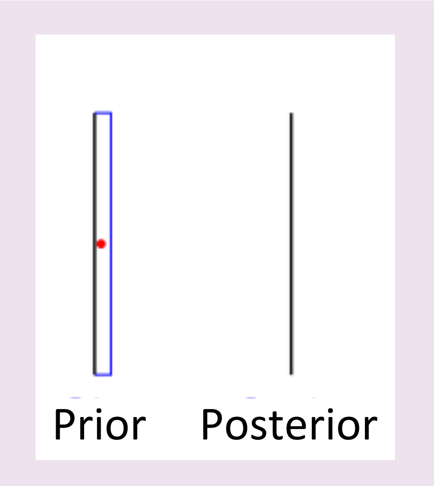


Figure adapted from Toni and Stumpf, “Tutorial on ABC rejection and ABC SMC for parameter estimation and model selection”, arxiv, 2010

Comparing MCMC methods with ABC

What is Approximate Bayesian Computation?

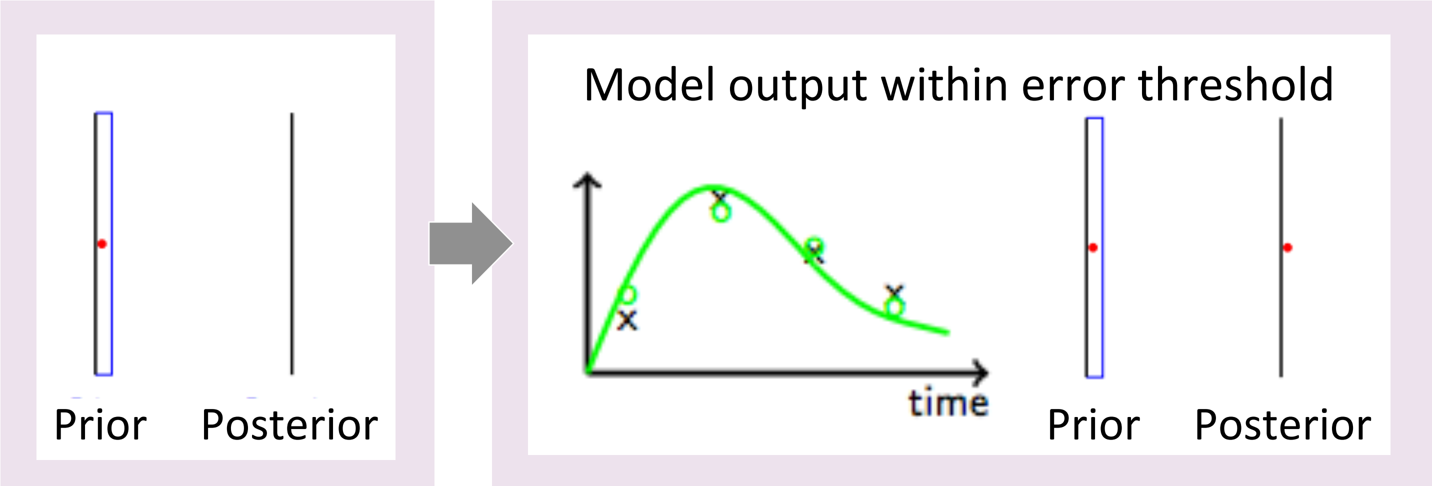


Figure adapted from Toni and Stumpf, "Tutorial on ABC rejection and ABC SMC for parameter estimation and model selection", arxiv, 2010

Comparing MCMC methods with ABC

What is Approximate Bayesian Computation?

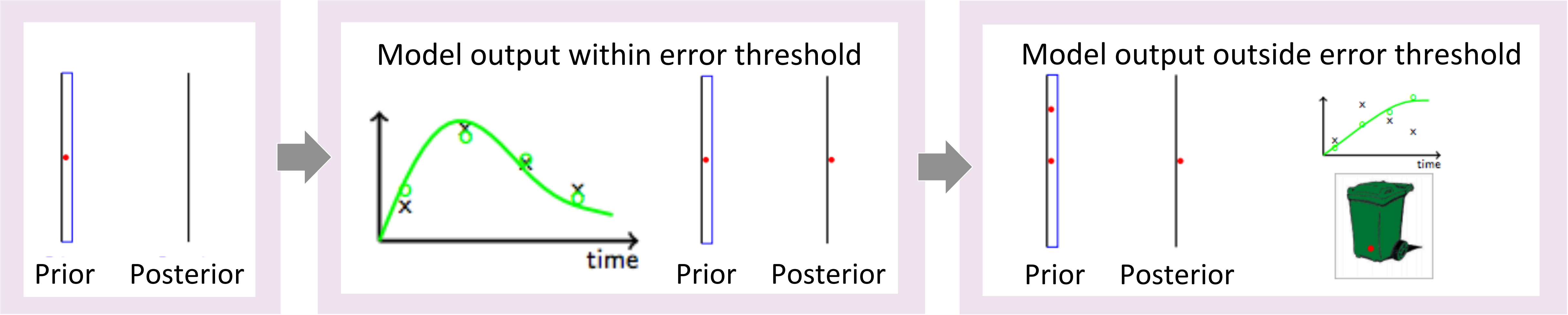


Figure adapted from Toni and Stumpf, “Tutorial on ABC rejection and ABC SMC for parameter estimation and model selection”, arxiv, 2010

Comparing MCMC methods with ABC

What is Approximate Bayesian Computation?

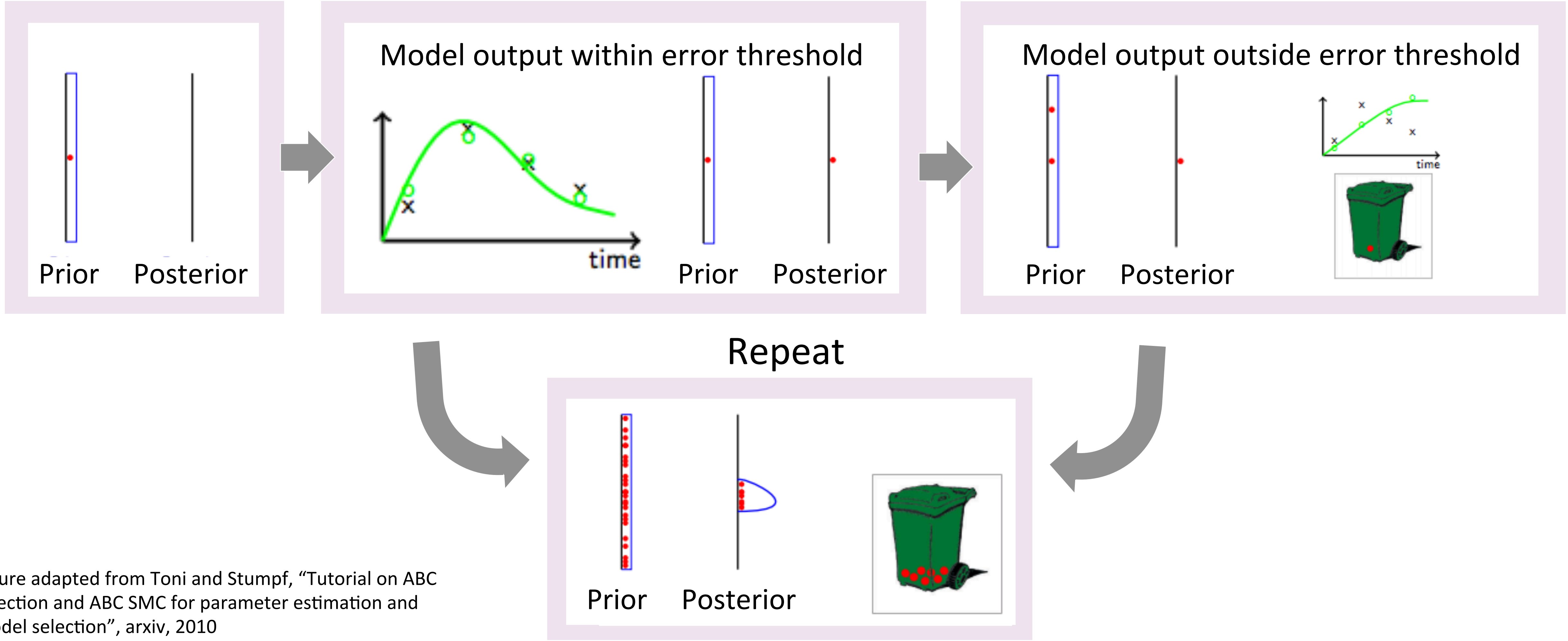


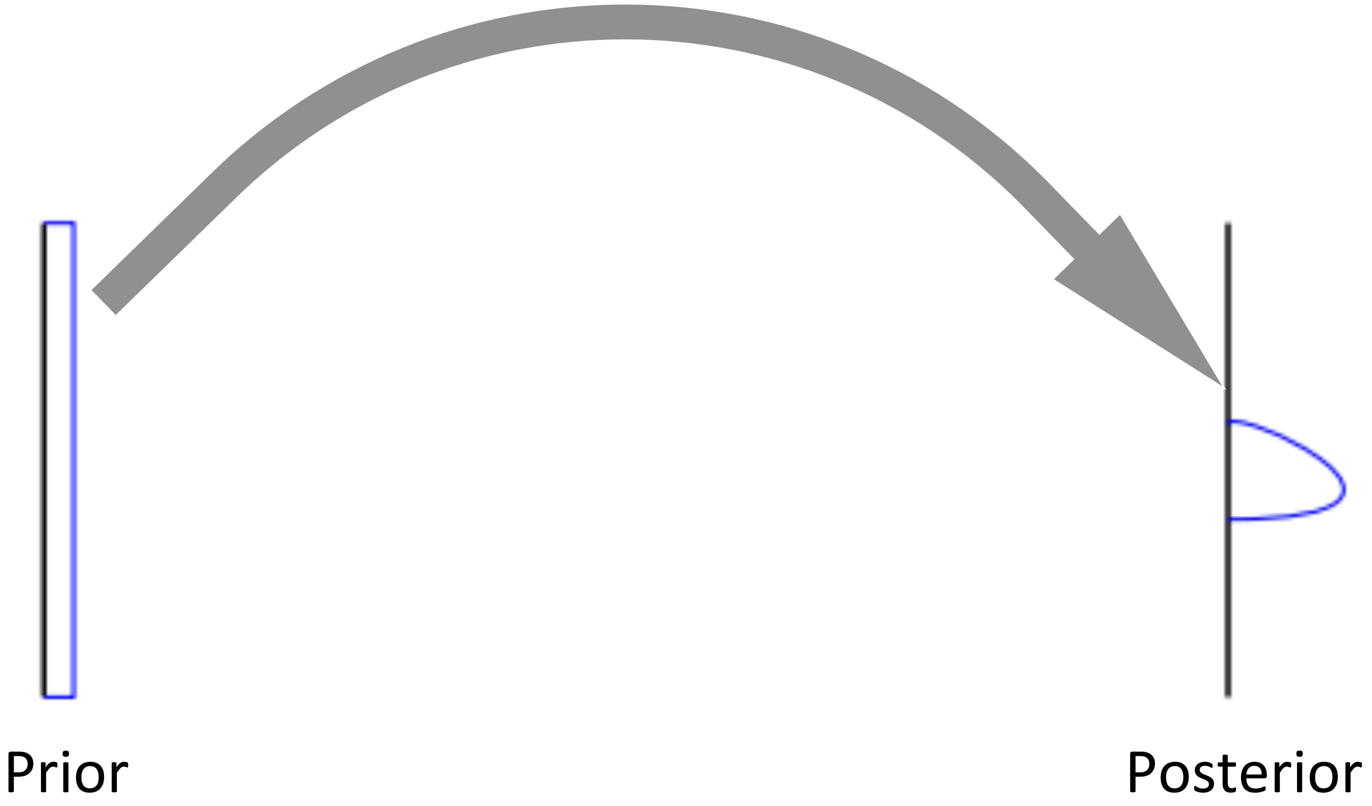
Figure adapted from Toni and Stumpf, "Tutorial on ABC rejection and ABC SMC for parameter estimation and model selection", arxiv, 2010

Comparing MCMC methods with ABC

Approximate Bayesian Computation –Sequential Monte Carlo (ABC-SMC):
Sequentially constructs the posterior via intermediate distributions with decreasing error

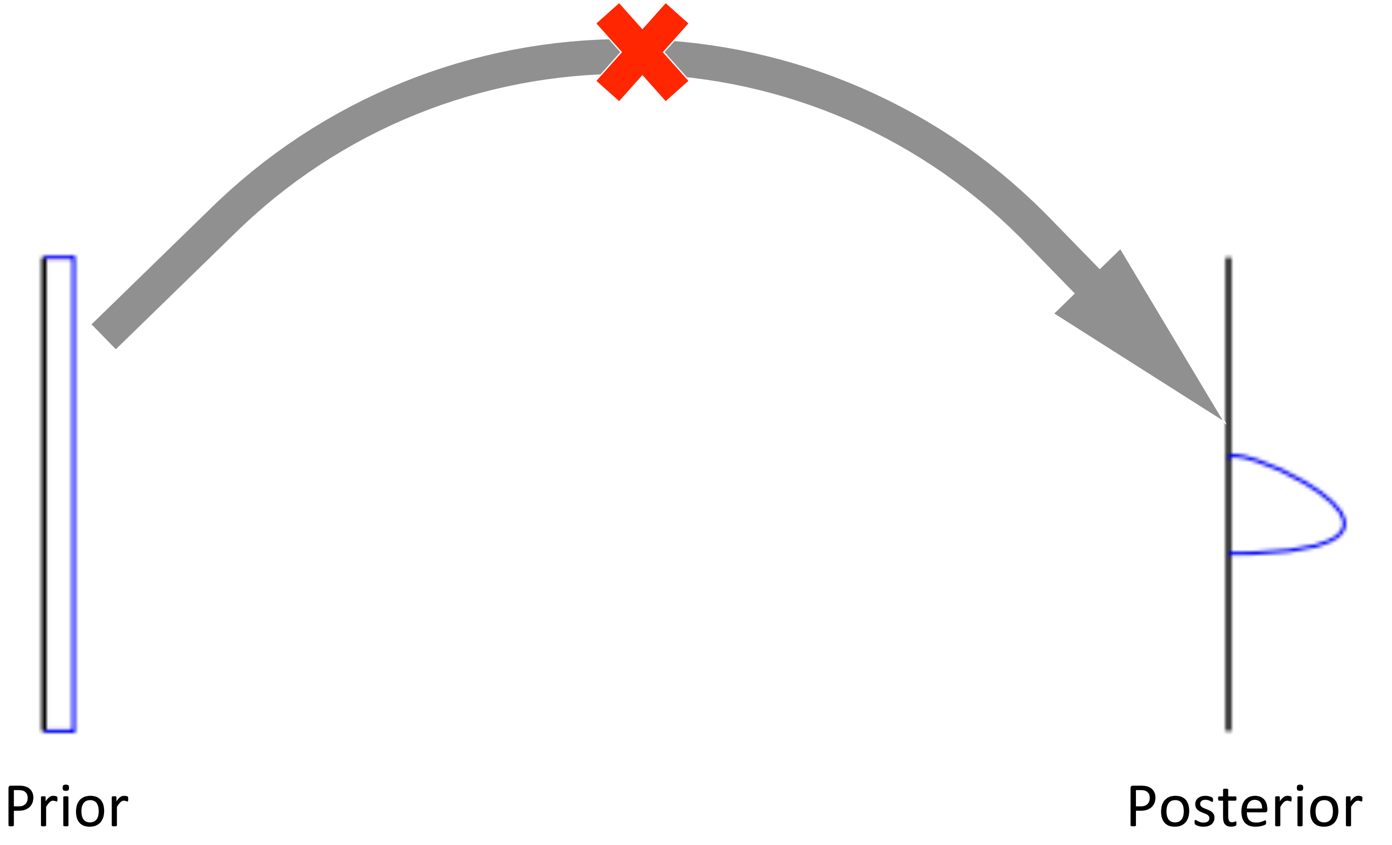
Comparing MCMC methods with ABC

Approximate Bayesian Computation –Sequential Monte Carlo (ABC-SMC):
Sequentially constructs the posterior via intermediate distributions with decreasing error



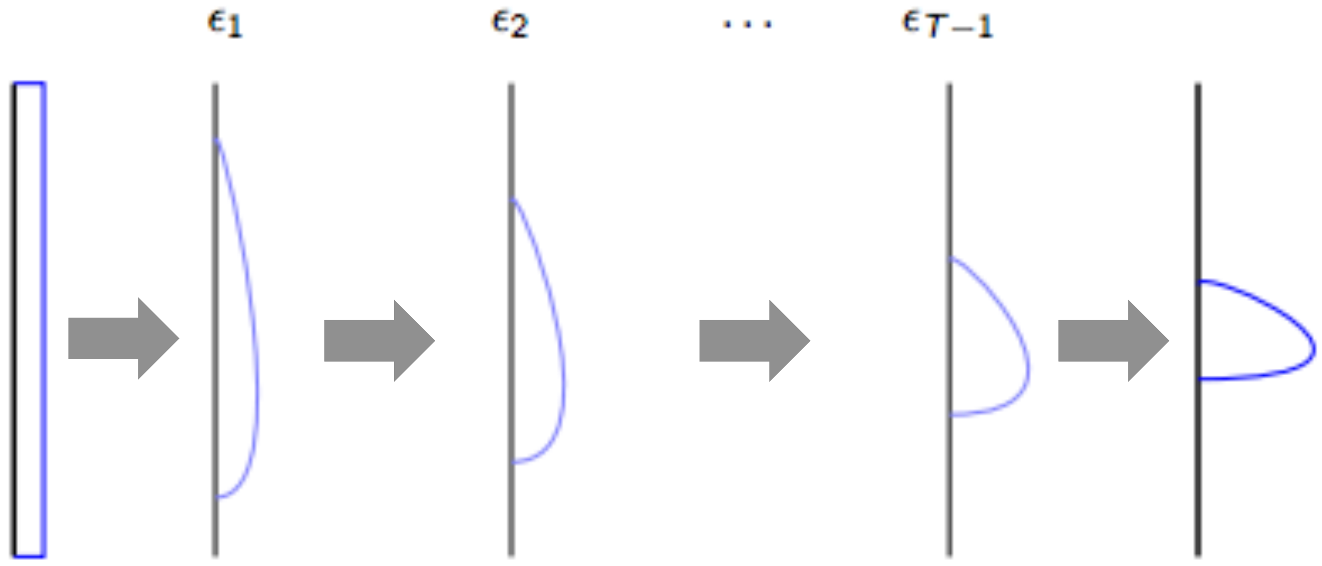
Comparing MCMC methods with ABC

Approximate Bayesian Computation –Sequential Monte Carlo (ABC-SMC):
Sequentially constructs the posterior via intermediate distributions with decreasing error



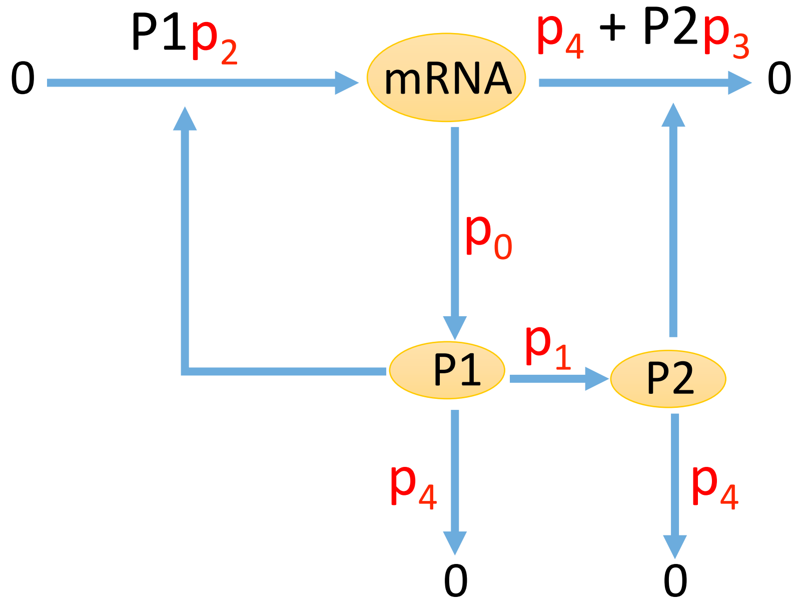
Comparing MCMC methods with ABC

Approximate Bayesian Computation –Sequential Monte Carlo (ABC-SMC):
Sequentially constructs the posterior via intermediate distributions with decreasing error

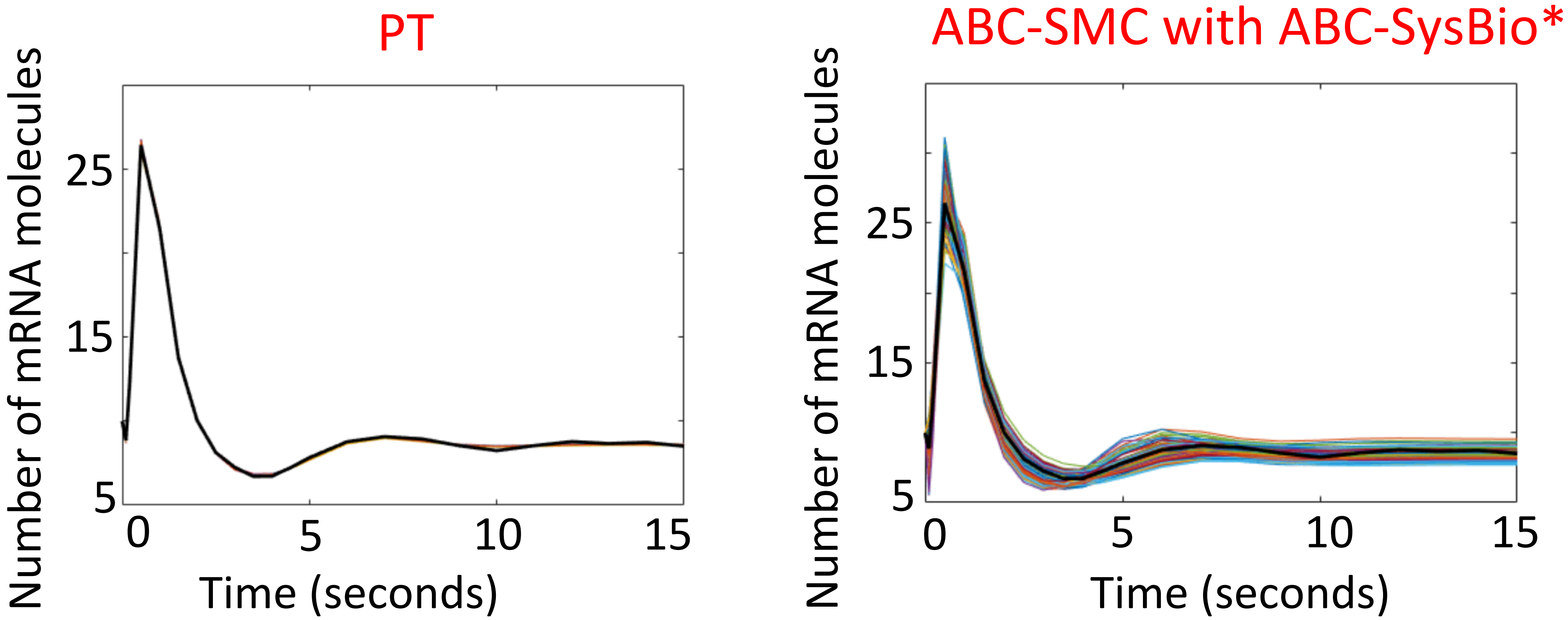


Prior Intermediate distributions Posterior

Case 2: Comparing MCMC methods with ABC: 5-parameter model of mRNA self-regulation

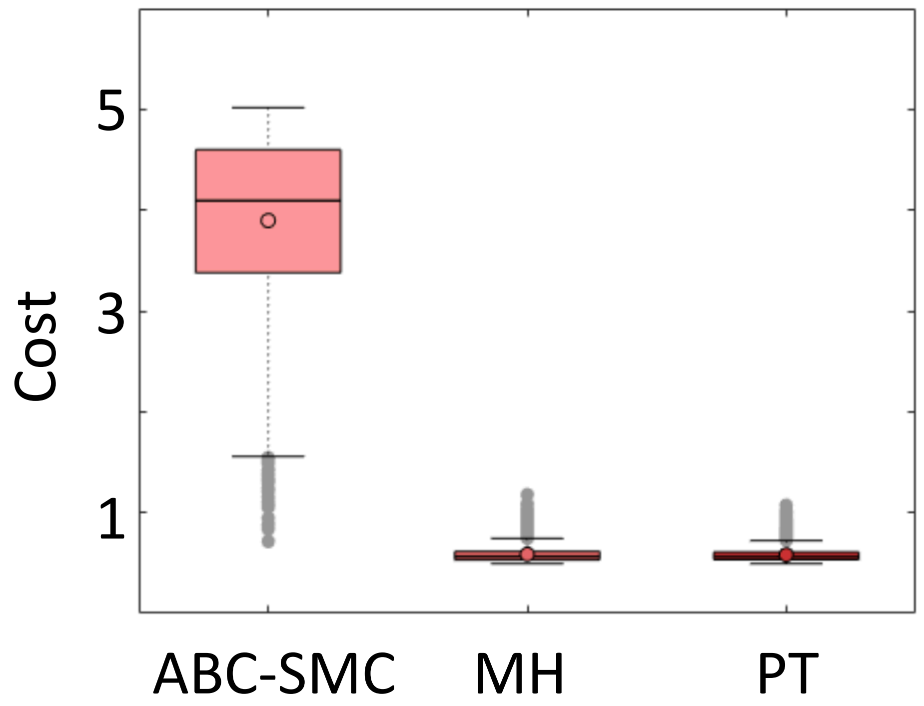


Comparing quality of fit with a fixed number of model integrations



*Liepe et al. A framework for parameter estimation and model selection from experimental data in systems biology using approximate Bayesian computation. Nature Protocols, 2014

Comparing quality of fit with a fixed number of model integrations



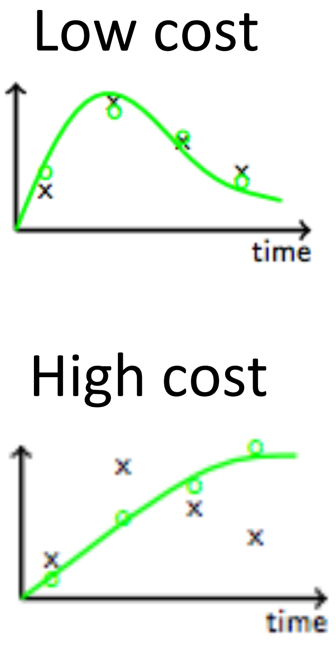
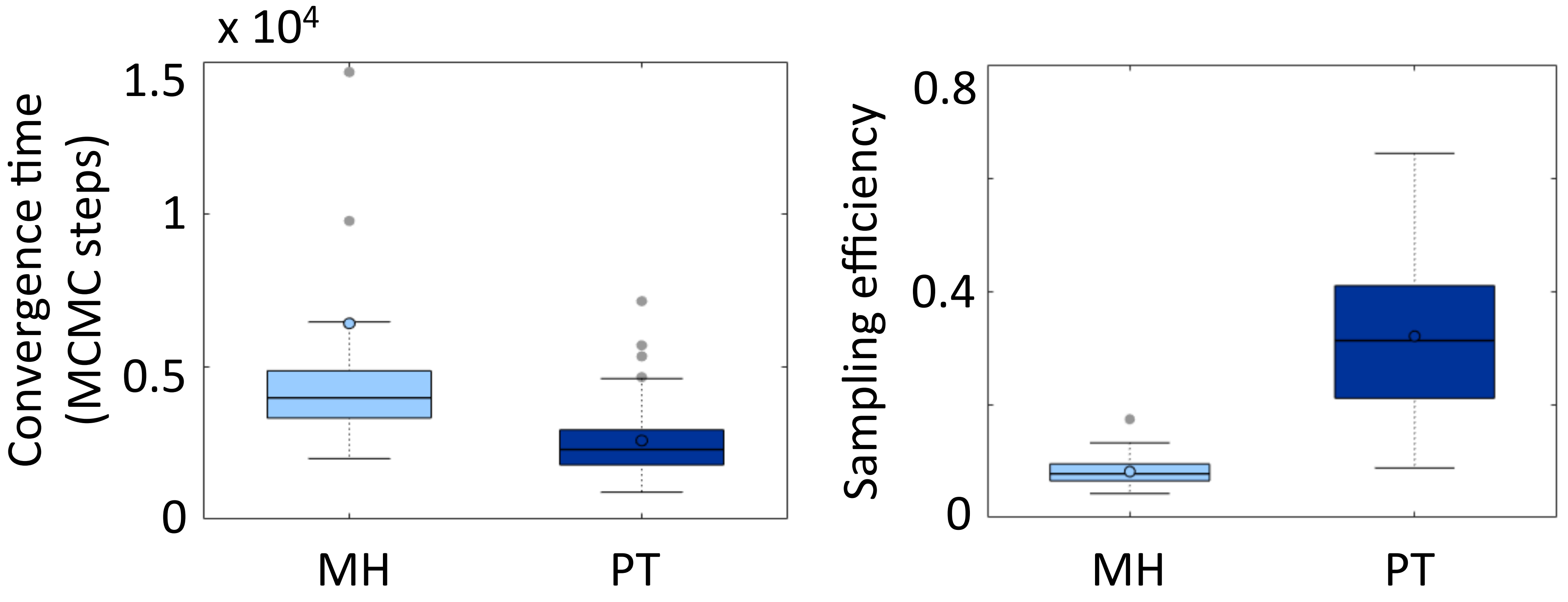
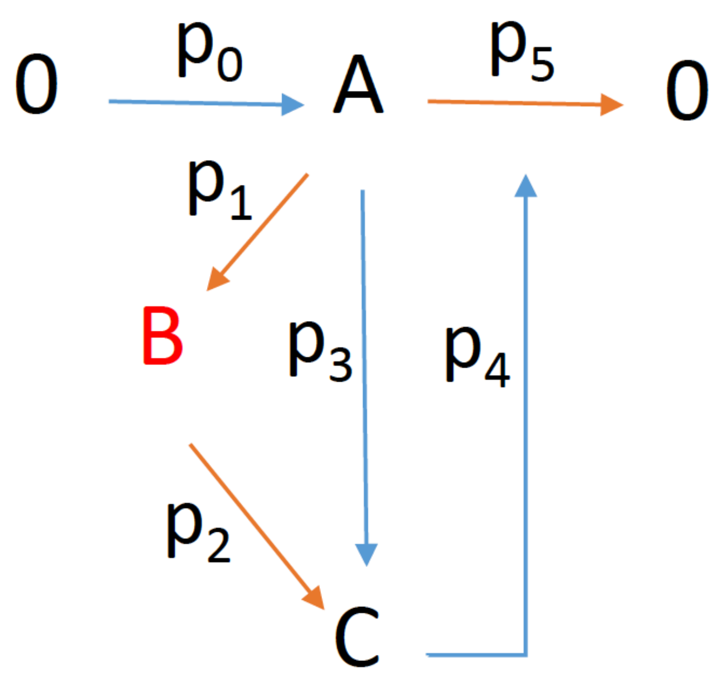


Figure adapted from Toni and Stumpf, "Tutorial on ABC rejection and ABC SMC for parameter estimation and model selection", arxiv, 2010

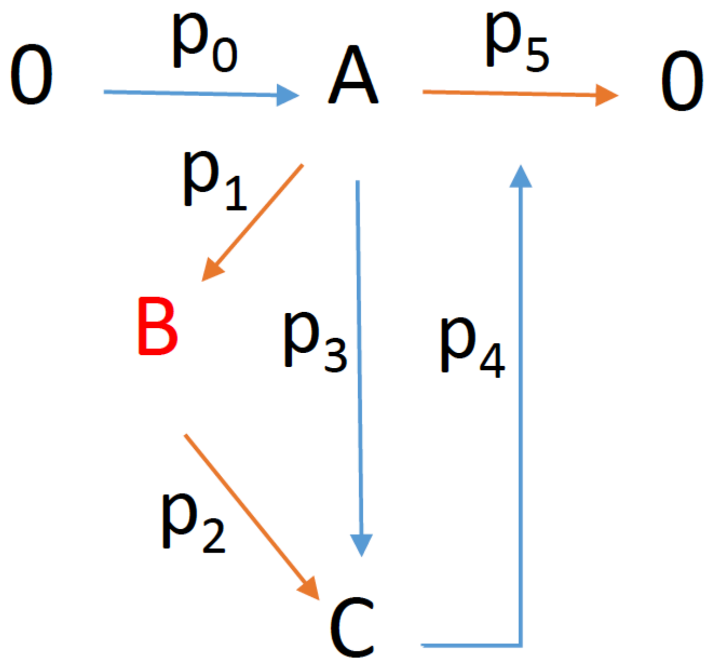
Performance comparison for 5-parameter mRNA self-regulation model



Case 3: Model reduction with the lasso penalty in a simple negative feedback model

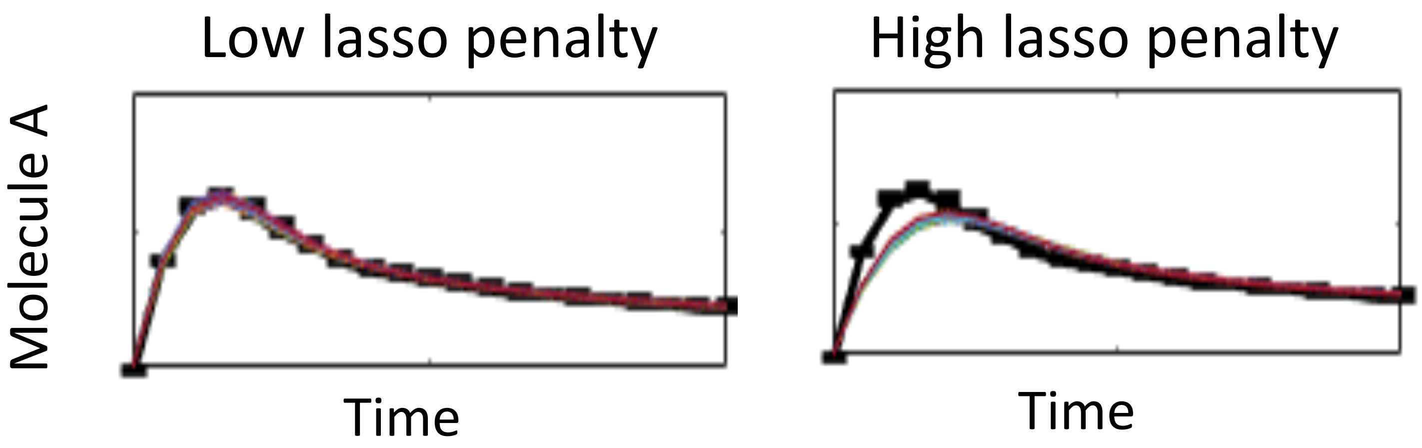


Case 3: Model reduction with the lasso penalty in a simple negative feedback model



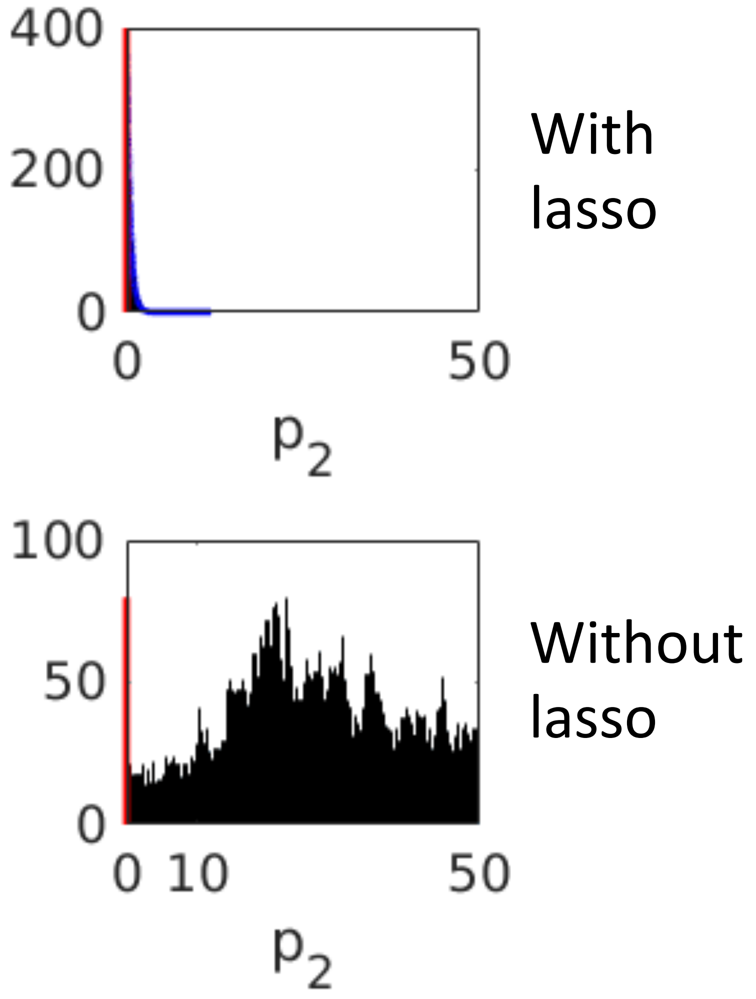
Penalty for non-zero parameters

Quality of fit with increasing strength of lasso penalty

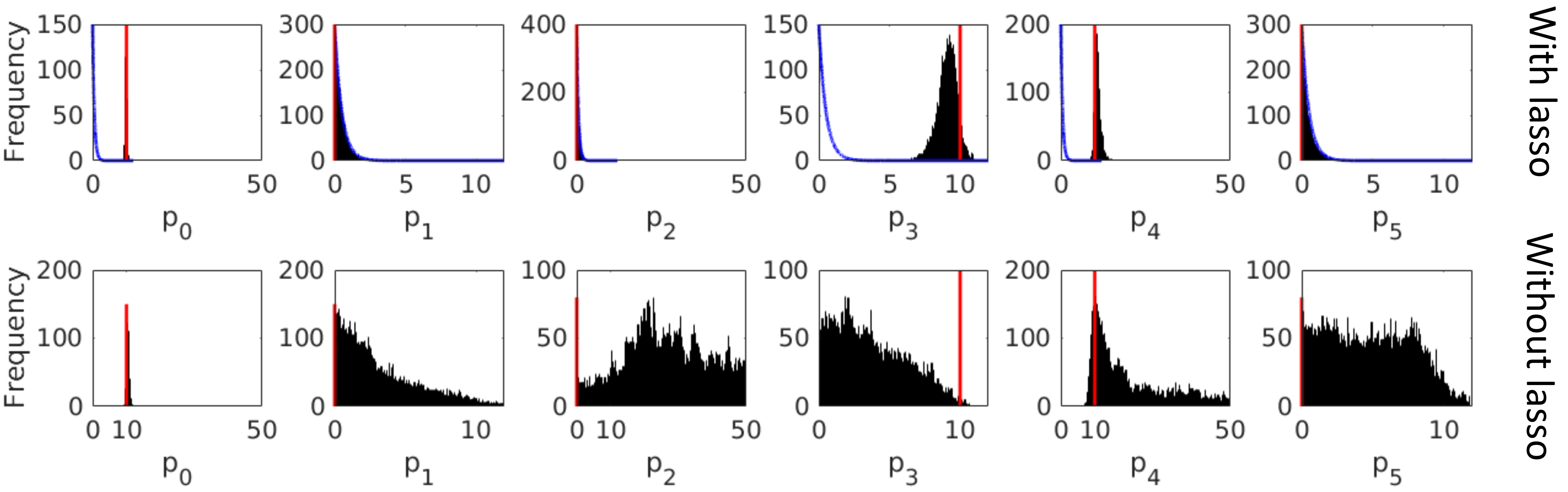


Choose the highest penalty that does not affect the quality of the fit

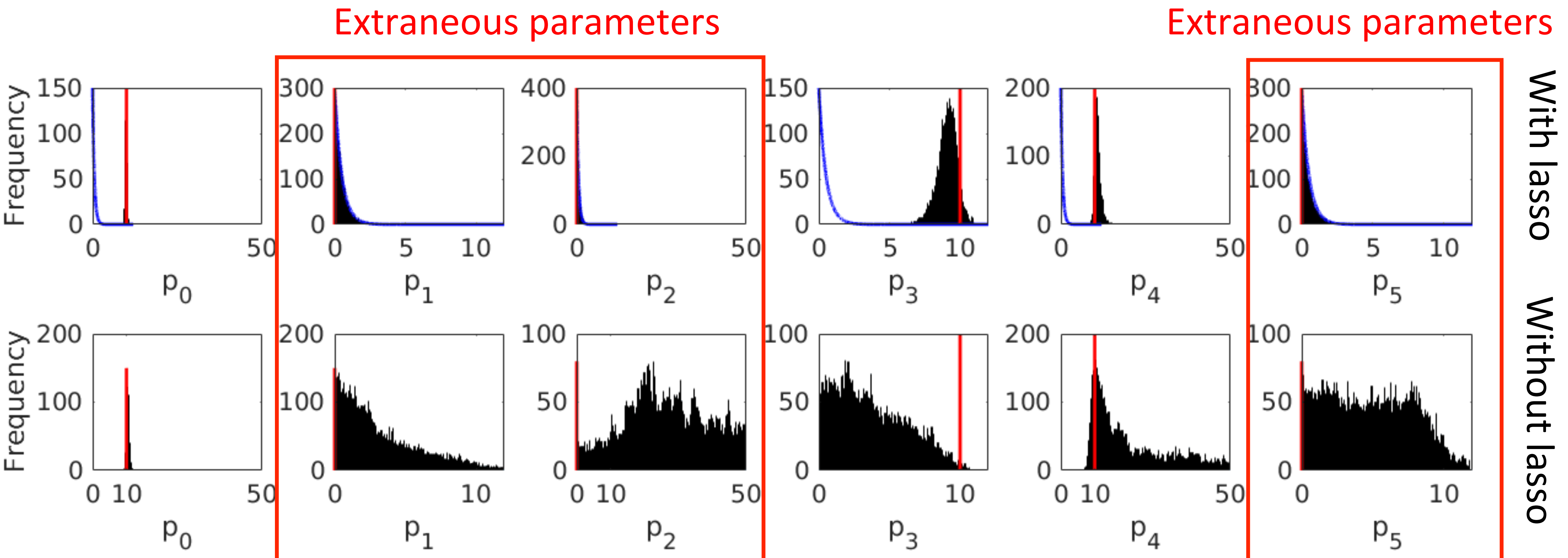
Parameter distributions inferred with and without the lasso penalty



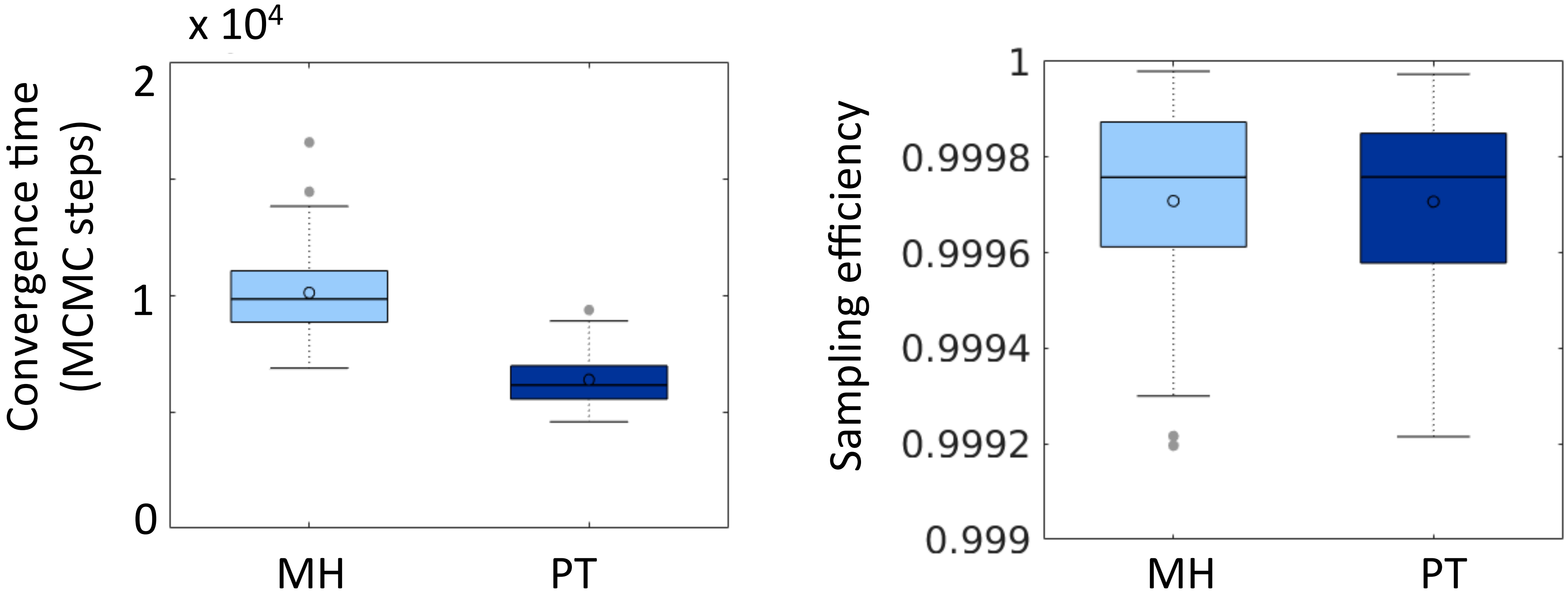
Parameter distributions inferred with and without the lasso penalty



Extraneous parameters are distributed near zero with the lasso penalty

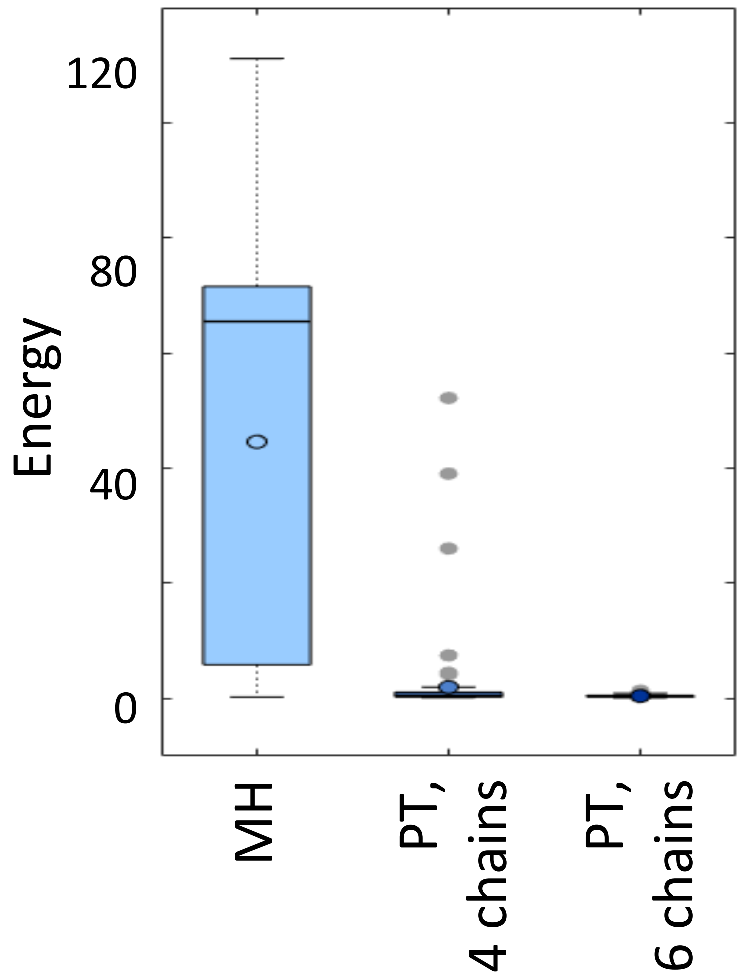


Performance comparison for 6-parameter negative feedback model

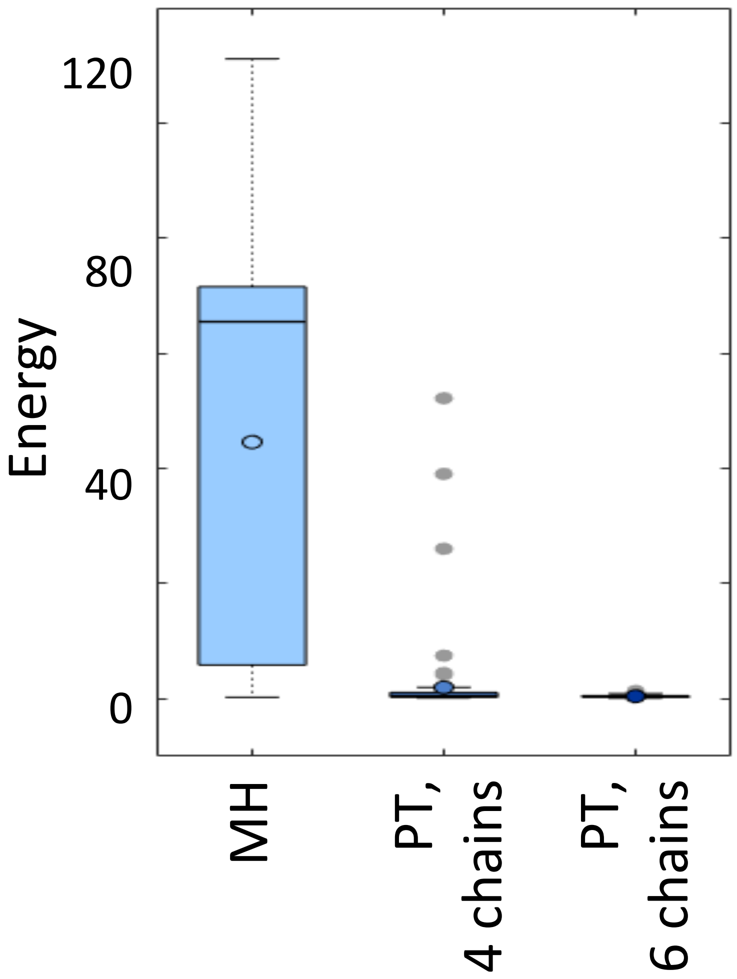


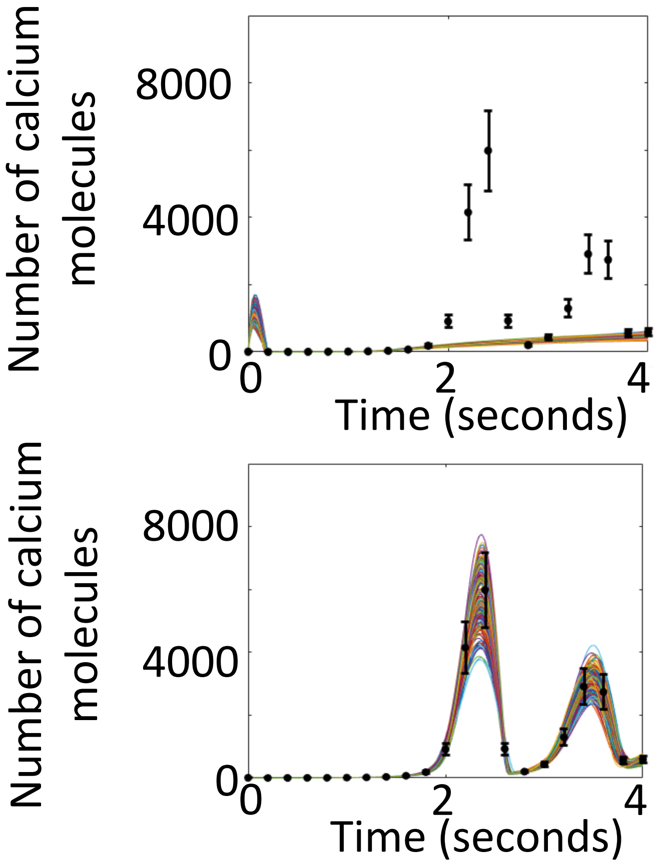
Case 4: Fitting to oscillations in a 12-parameter calcium signaling model

Case 4: Fitting to oscillations in a 12-parameter calcium signaling model



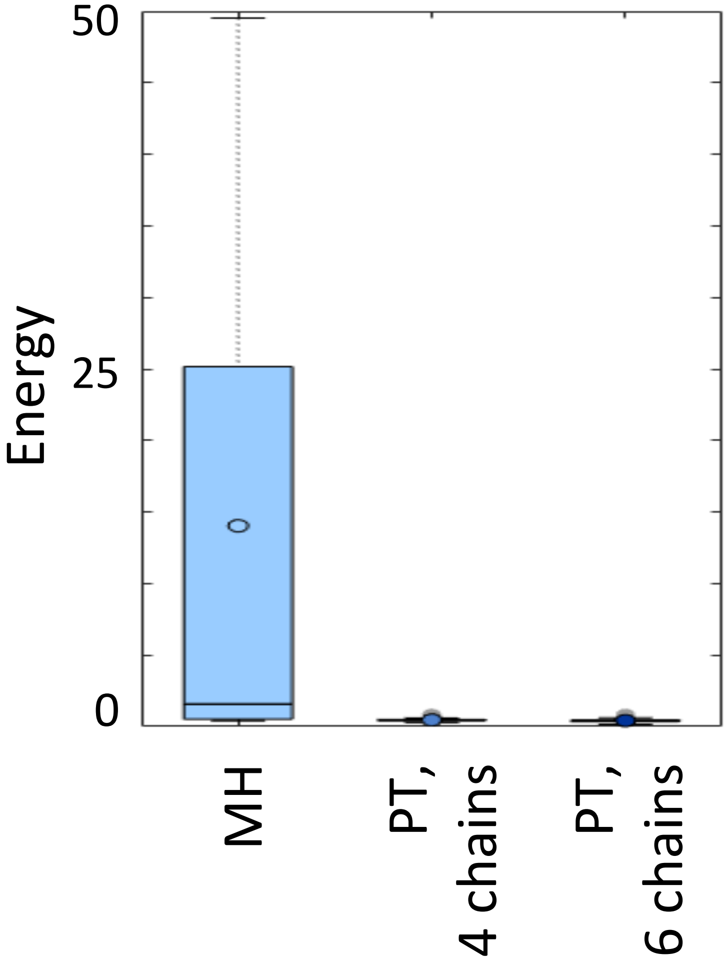
Case 4: Fitting to oscillations in a 12-parameter calcium signaling model

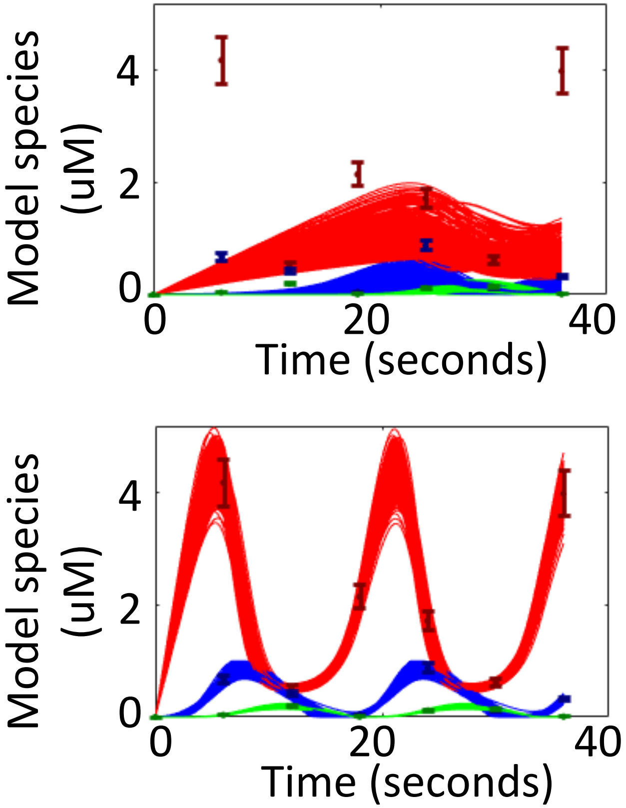




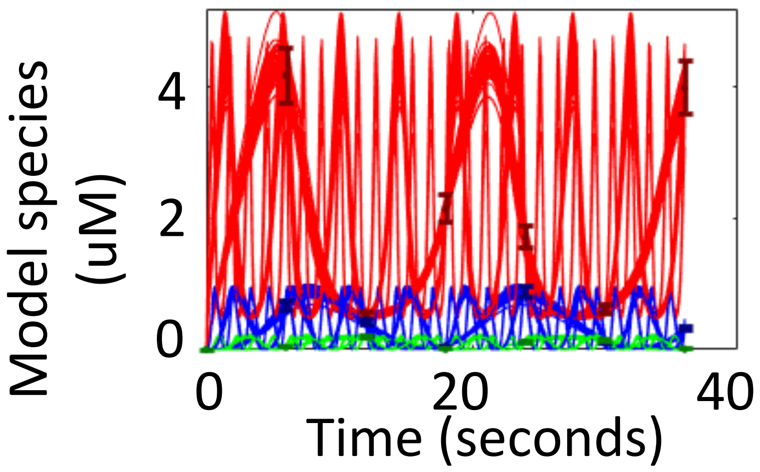
Case 5: Fitting to oscillations in 3 species: 13-parameter Negative Feedback Oscillator

Case 5: Fitting to oscillations in 3 species: 13-parameter Negative Feedback Oscillator

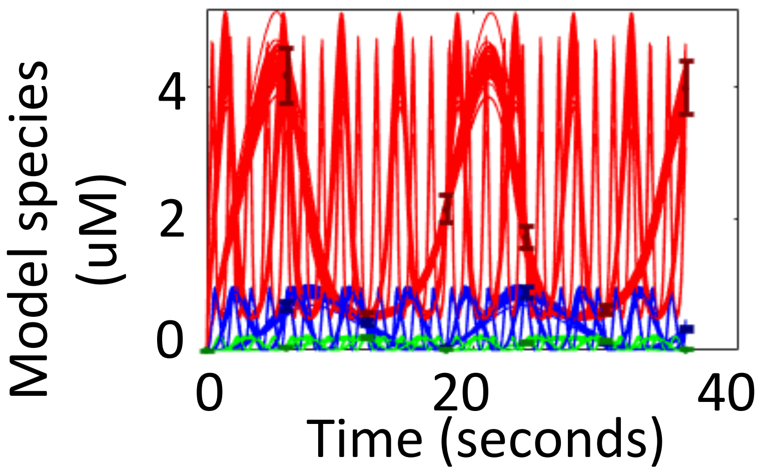


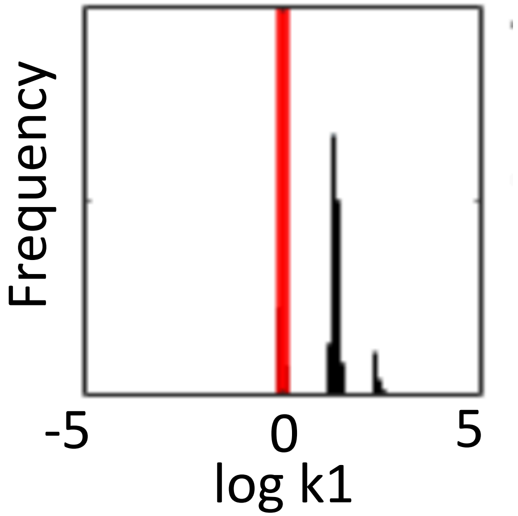


PT identifies multiple global minima

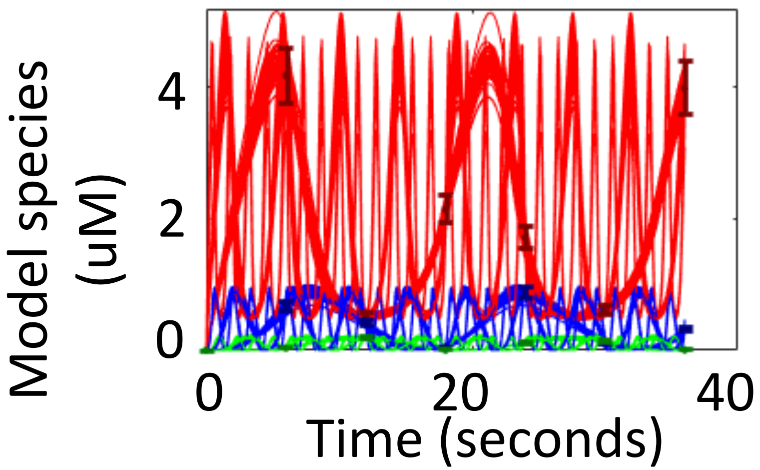


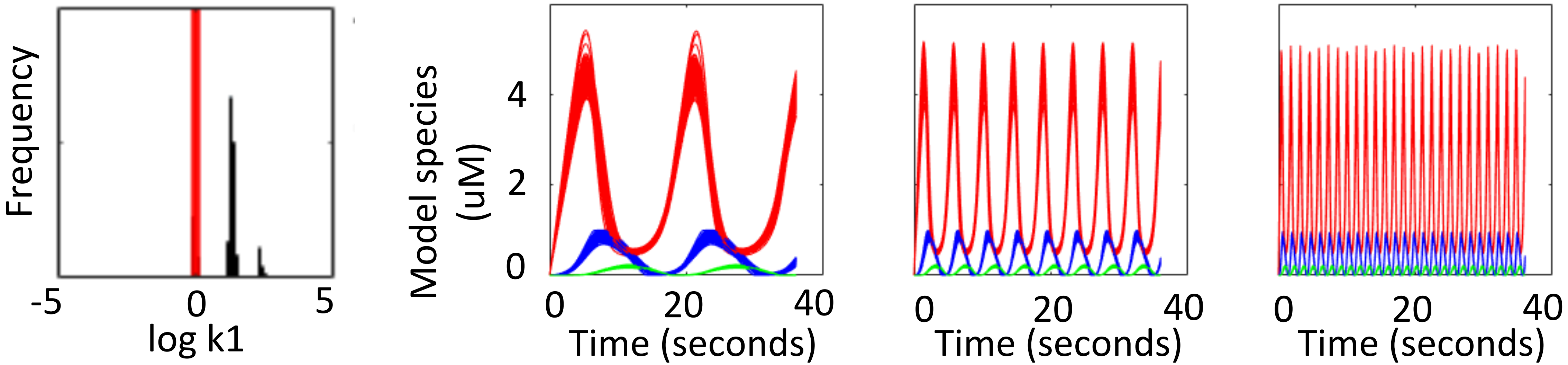
PT identifies multiple global minima



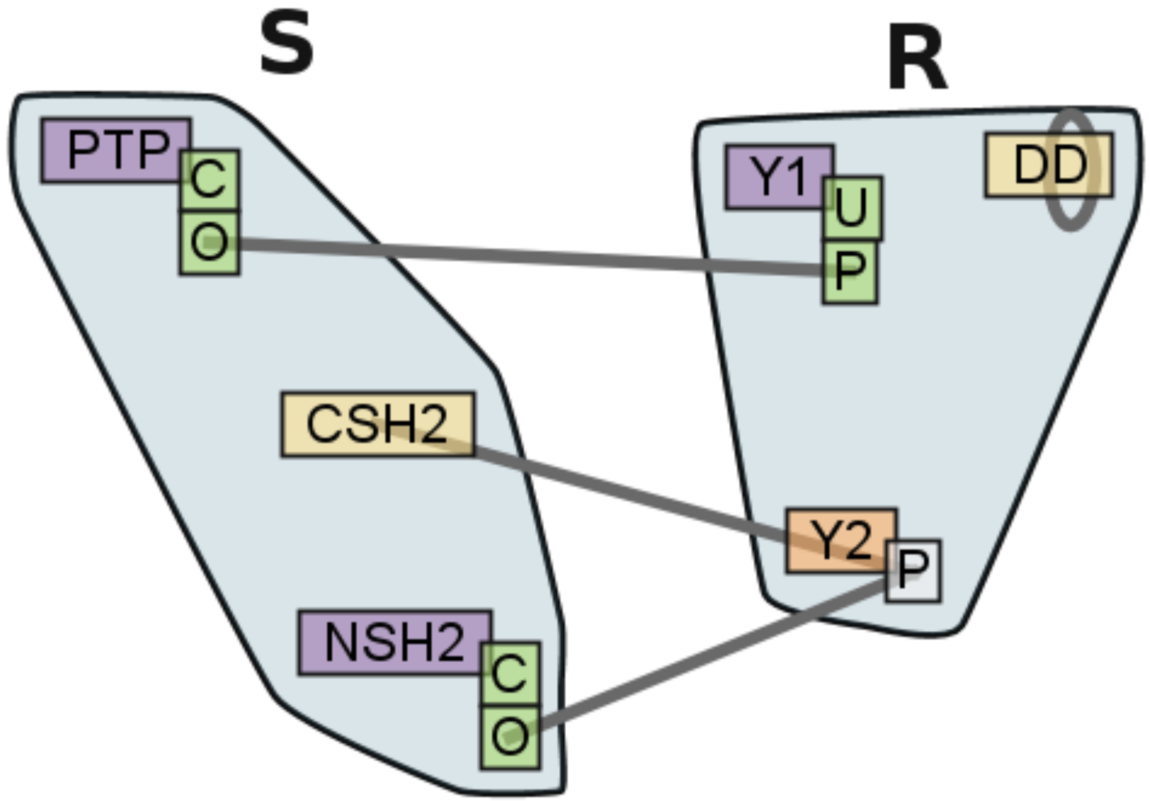


PT identifies multiple global minima



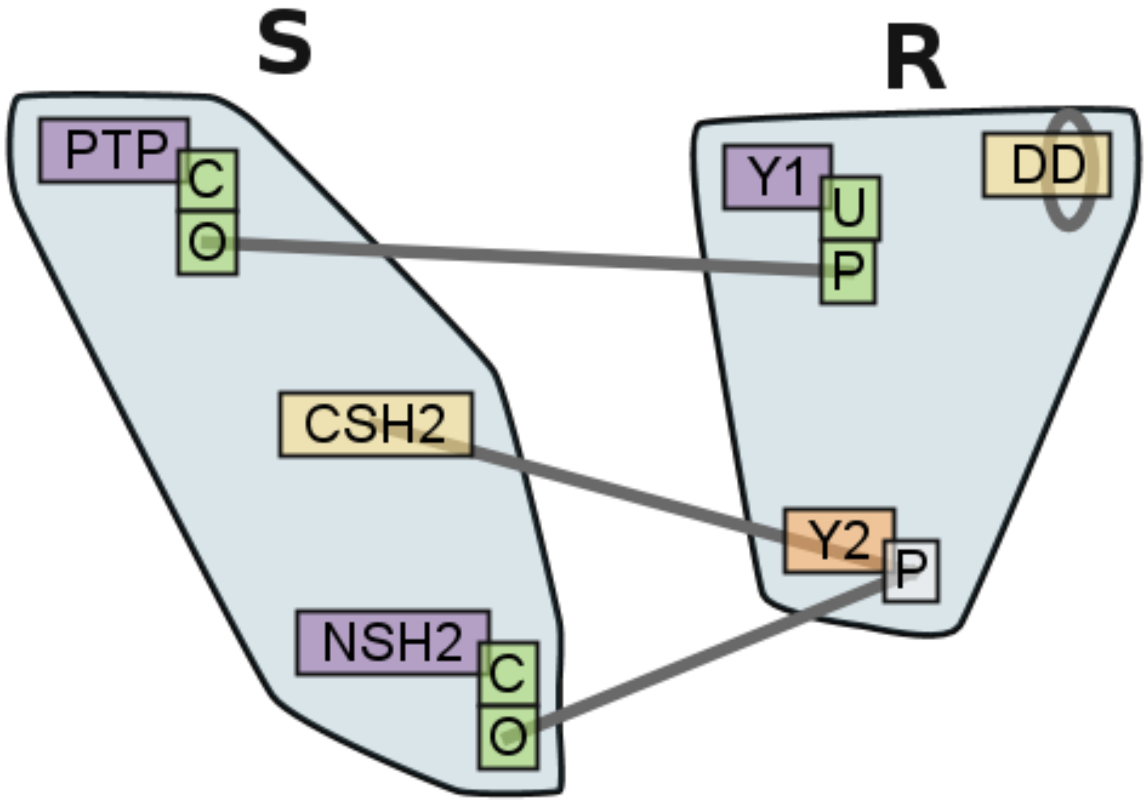


Moving towards realistic cell-signaling applications: Fitting a large model of growth factor signaling



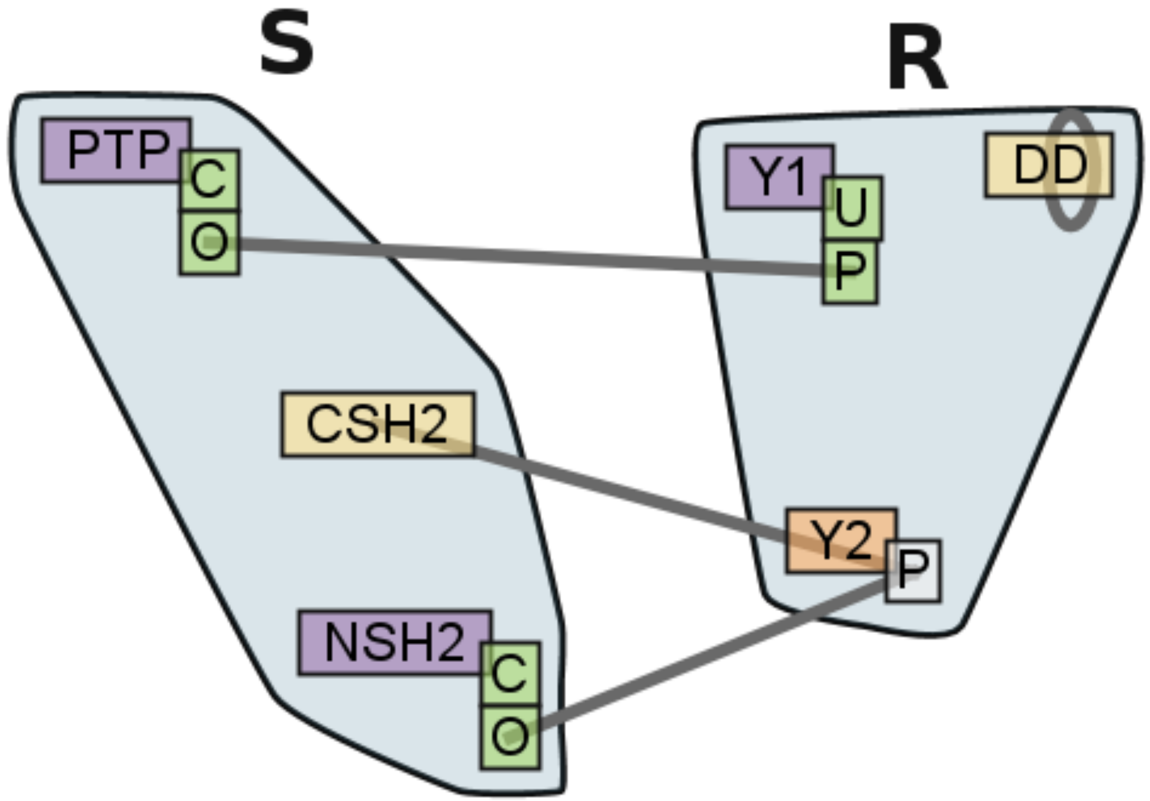
Moving towards realistic cell-signaling applications: Fitting a large model of growth factor signaling

- 24 parameters



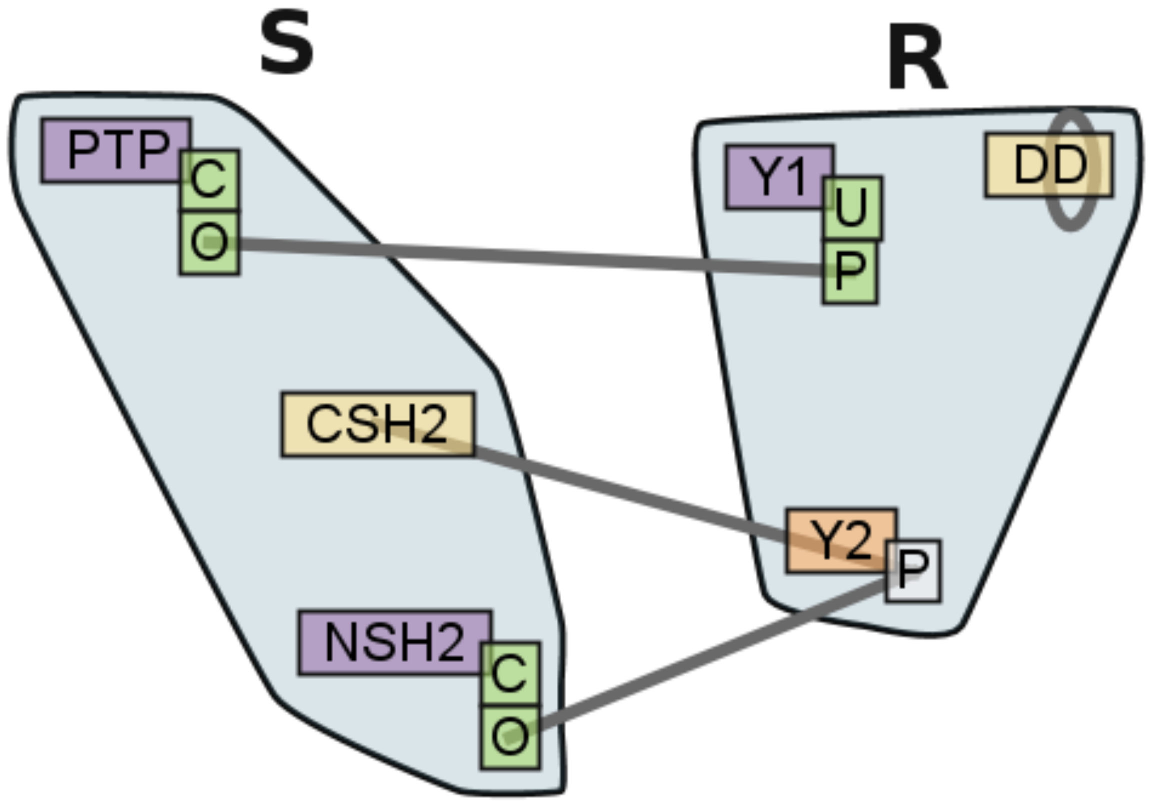
Moving towards realistic cell-signaling applications: Fitting a large model of growth factor signaling

- 24 parameters
- 1037 reactions



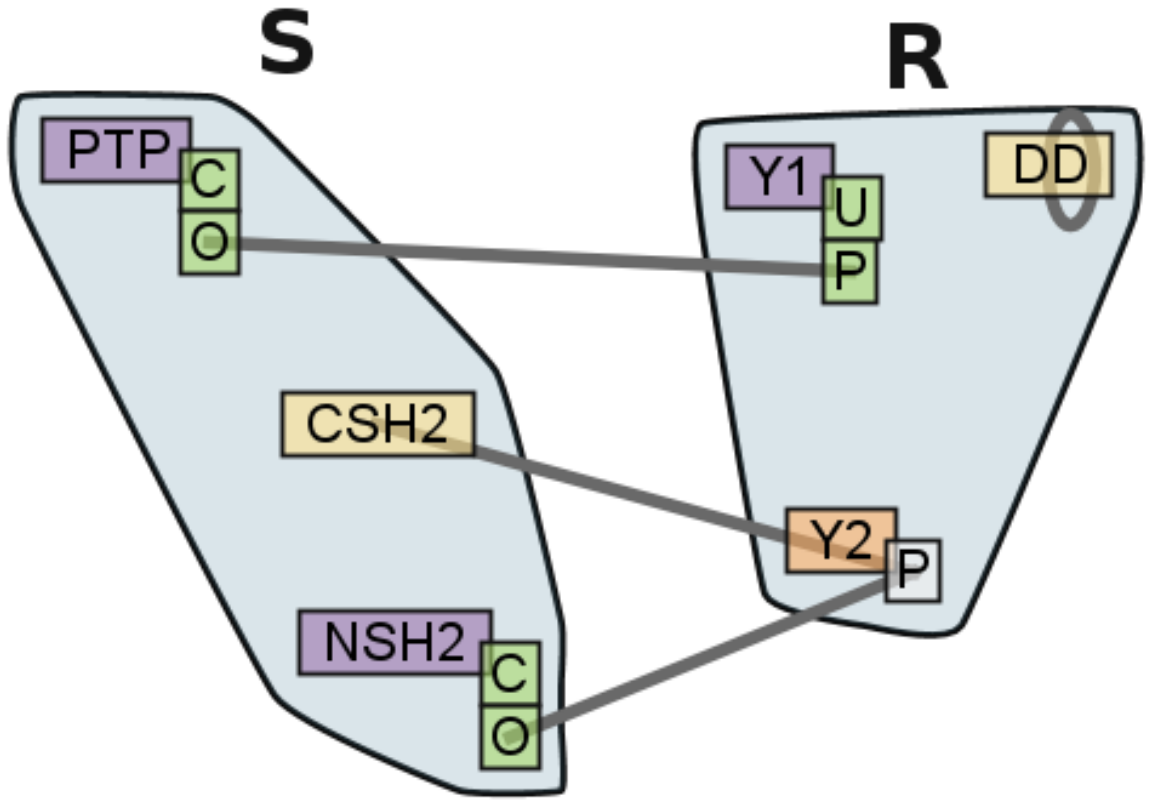
Moving towards realistic cell-signaling applications: Fitting a large model of growth factor signaling

- 24 parameters
- 1037 reactions
- 149 ODEs

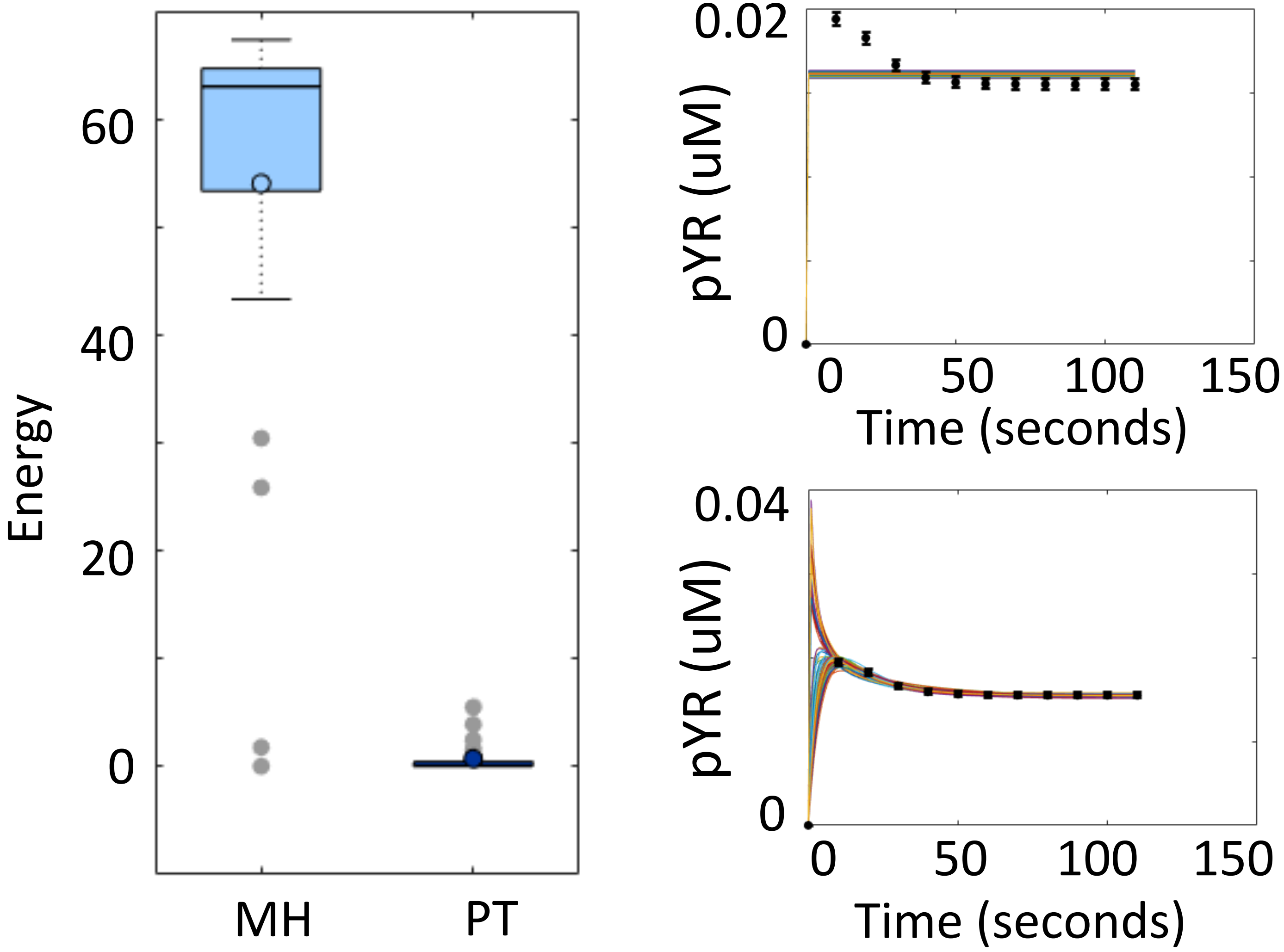


Moving towards realistic cell-signaling applications: Fitting a large model of growth factor signaling

- 24 parameters
- 1037 reactions
- 149 ODEs
- pYR combines 136 model species



Moving towards realistic cell-signaling applications: Fitting a large model of growth factor signaling



Summary:

Summary:

- For simple models PT typically outperformed MH w.r.t convergence speed and sampling efficiency

Summary:

- For simple models PT typically outperformed MH w.r.t convergence speed and sampling efficiency

- For more complex models PT consistently found global minima whereas MH often gets stuck in a local minimum

Summary:

- For simple models PT typically outperformed MH w.r.t convergence speed and sampling efficiency

- For more complex models PT consistently found global minima whereas MH often gets stuck in a local minimum

- PT outperformed ABC on a simple mRNA self-regulation model

Summary:

- For simple models PT typically outperformed MH w.r.t convergence speed and sampling efficiency

- For more complex models PT consistently found global minima whereas MH often gets stuck in a local minimum

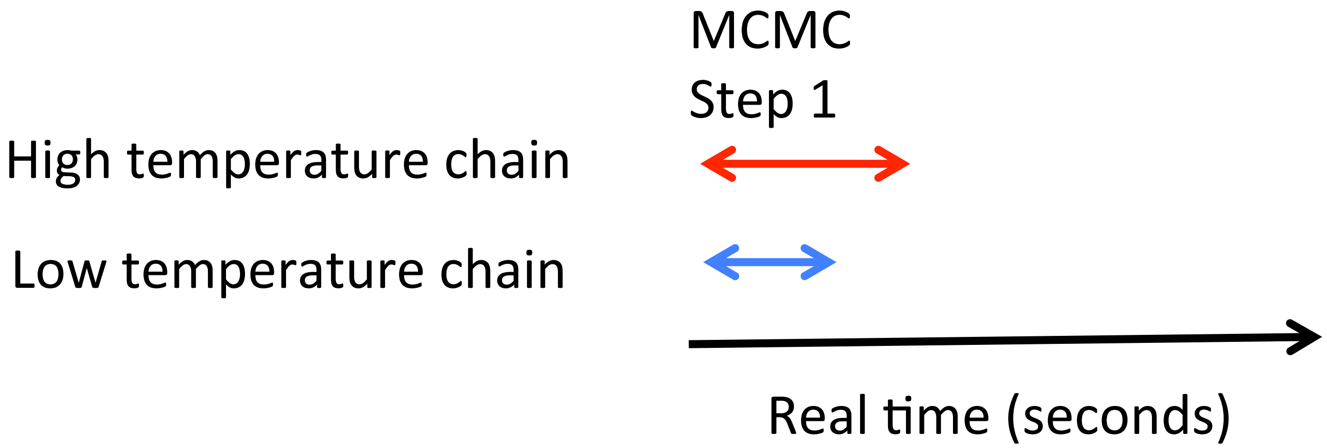
- PT outperformed ABC on a simple mRNA self-regulation model

- PT efficiently reduced a moderately complex negative feedback model

Limitations and fixes

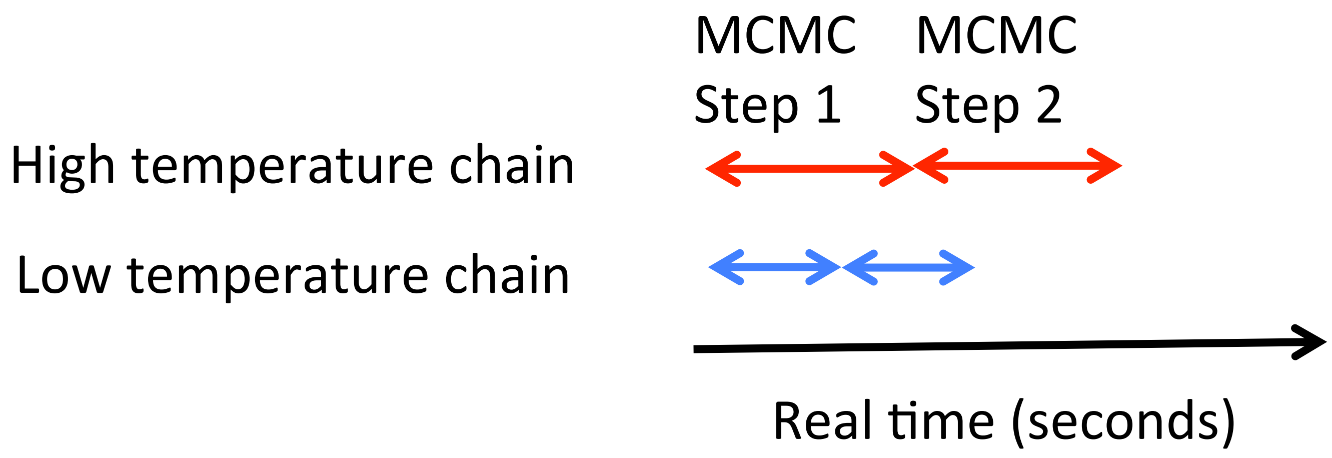
Limitations and fixes

- Synchronization has a cost



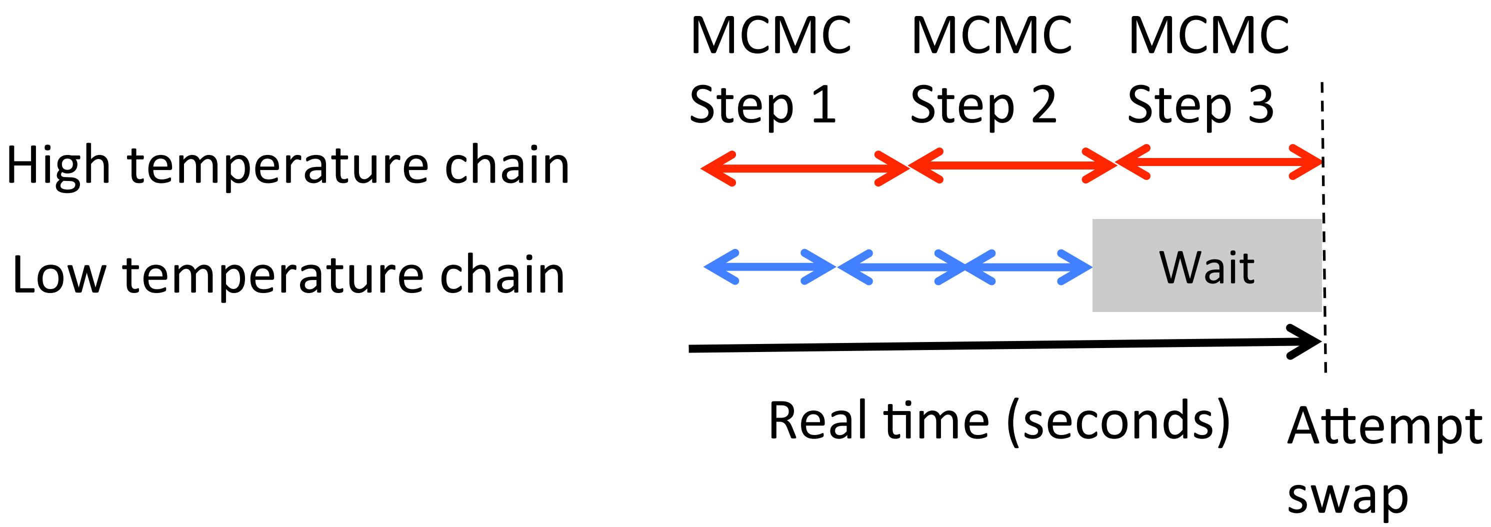
Limitations and fixes

- Synchronization has a cost



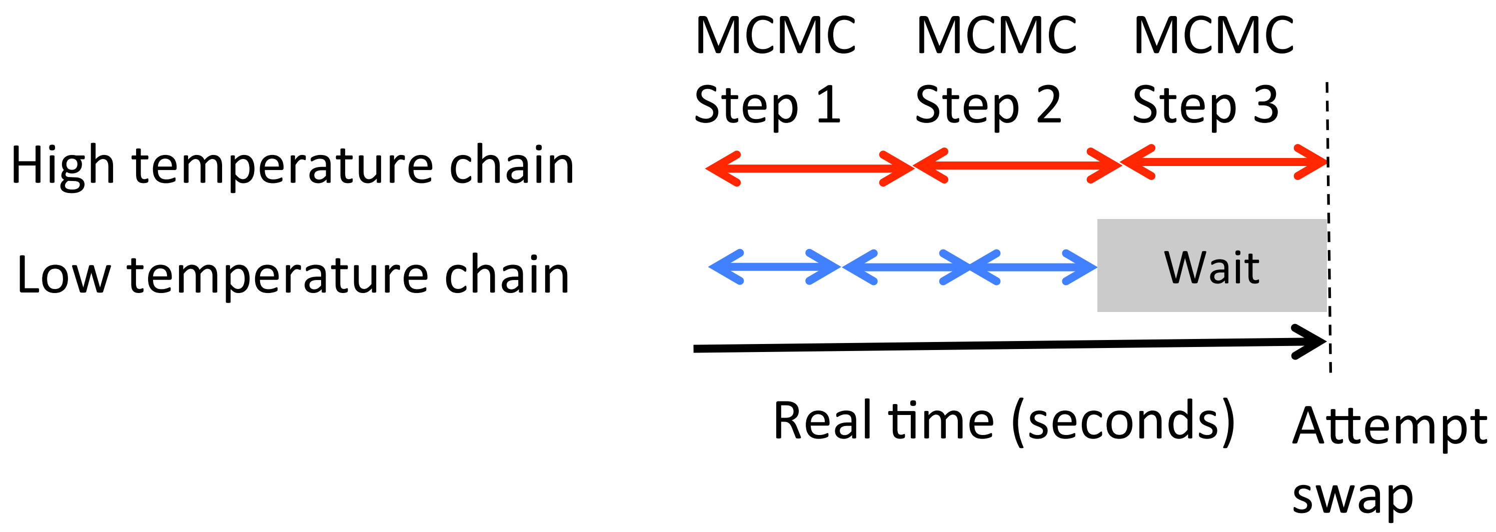
Limitations and fixes

- Synchronization has a cost



Limitations and fixes

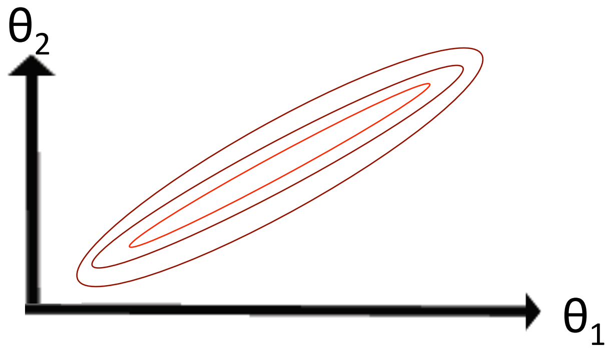
- Synchronization has a cost



Solution: asynchronous swapping

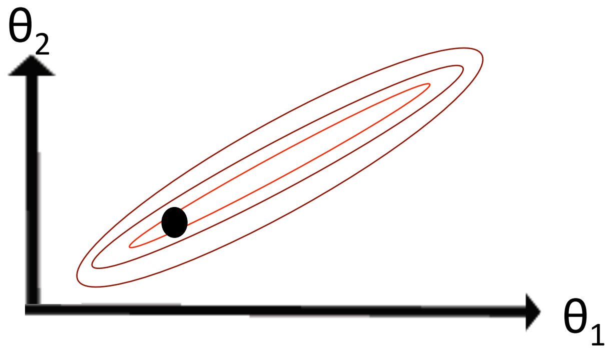
Limitations and fixes

- Proposal function does not leverage parameter correlations



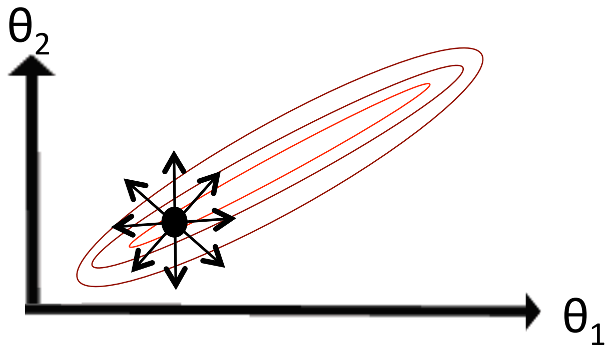
Limitations and fixes

- Proposal function does not leverage parameter correlations



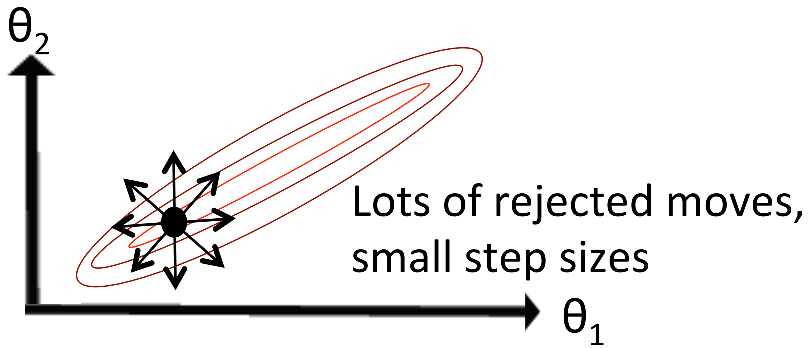
Limitations and fixes

- Proposal function does not leverage parameter correlations



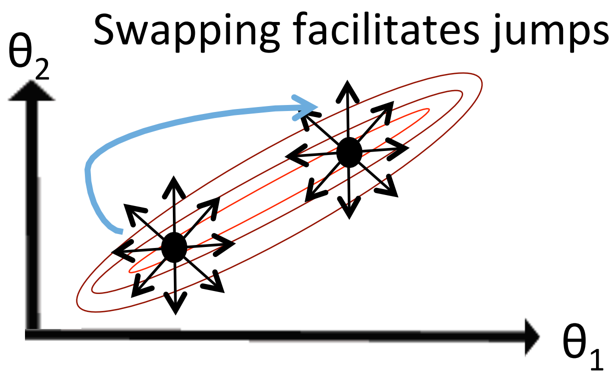
Limitations and fixes

- Proposal function does not leverage parameter correlations



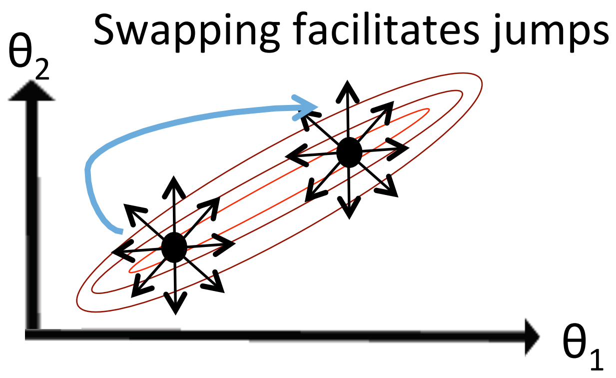
Limitations and fixes

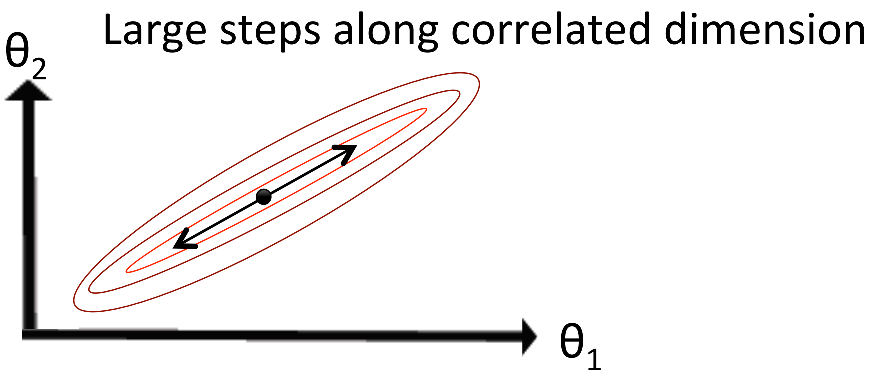
- Proposal function does not leverage parameter correlations



Limitations and fixes

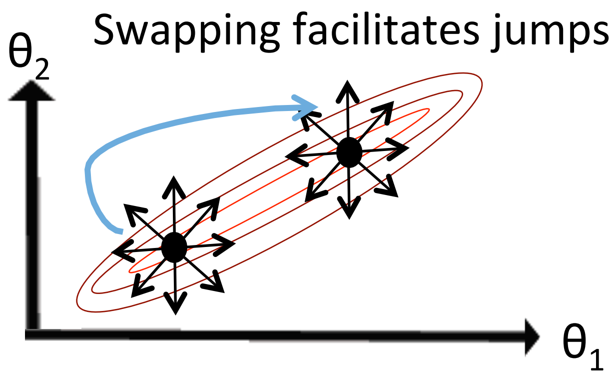
- Proposal function does not leverage parameter correlations

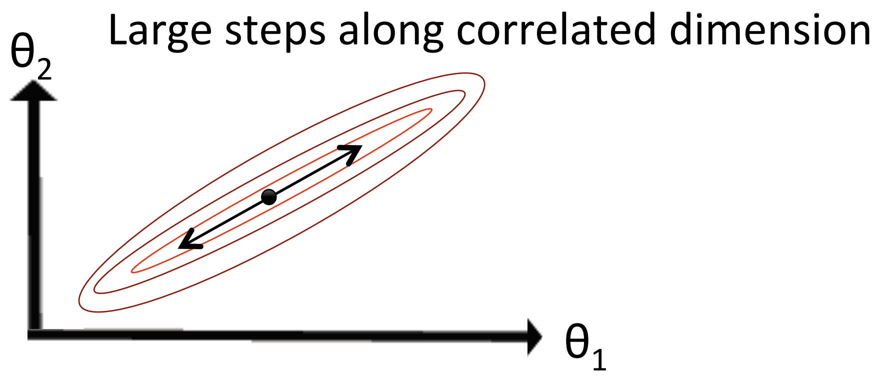




Limitations and fixes

- Proposal function does not leverage parameter correlations





Solution:
Hessian guidance*
Multi-chain Monte Carlo**

*H. Eydgahi et al. Properties of cell death models calibrated and compared using Bayesian approaches. Molecular Systems Biology 2014.
** Zhang, L. A. et al. APT-MCMC, a C++/Python implementation of Markov Chain Monte Carlo for parameter identification. Comput. Chem. Eng. 2018.

Limitations and fixes

- Method is only moderately parallel, does not fully leverage large number of nodes on typical modern day clusters

Limitations and fixes

- Method is only moderately parallel, does not fully leverage large number of nodes on typical modern day clusters

Solution:
Combine results from multiple PT chains
Run multiple chains at each temperature level*

.
* Zhang, L. A. et al. APT-MCMC, a C++/Python implementation of Markov Chain Monte Carlo for parameter identification. Comput. Chem. Eng. 2018.

Software: pTempEst

pTempEst

https://github.com/RuleWorld/ptempest

Software: pTempEst

pTempEst

https://github.com/RuleWorld/ptempest

MATLAB package, Supports models in SBML, BNGL, file formats

Software: pTempEst

pTempEst

https://github.com/RuleWorld/ptempest

MATLAB package, Supports models in SBML, BNGL, file formats

Required software:

MATLAB

Software: pTempEst

pTempEst

https://github.com/RuleWorld/ptempest

MATLAB package, Supports models in SBML, BNGL, file formats

Required software:

- MATLAB

Recommended software:

- BioNetGen (http://bionetgen.org)
  - -> writeMfile, writeMexfile
- SUNDIALS CVODE integrator

Software: pTempEst

pTempEst

https://github.com/RuleWorld/ptempest

MATLAB package, Supports models in SBML, BNGL, file formats

Required software:

MATLAB

Recommended software:

BioNetGen (http://bionetgen.org)

-> writeMfile, writeMexfile

SUNDIALS CVODE integrator

Features: Adaptive step sizes, temperatures, user-defined objective functions and likelihood functions, variety of in-built options for likelihoods and priors

Future development

- Python implementation of parallel tempering that addresses observed limitations and is closely interfaced with BioNetGen

Future development

- Python implementation of parallel tempering that addresses observed limitations and is closely interfaced with BioNetGen

- BNG ODE model export to C with swig wrappers for fast integration in python

```
writeCfile({}),
writeCfile({swig=>1})
```

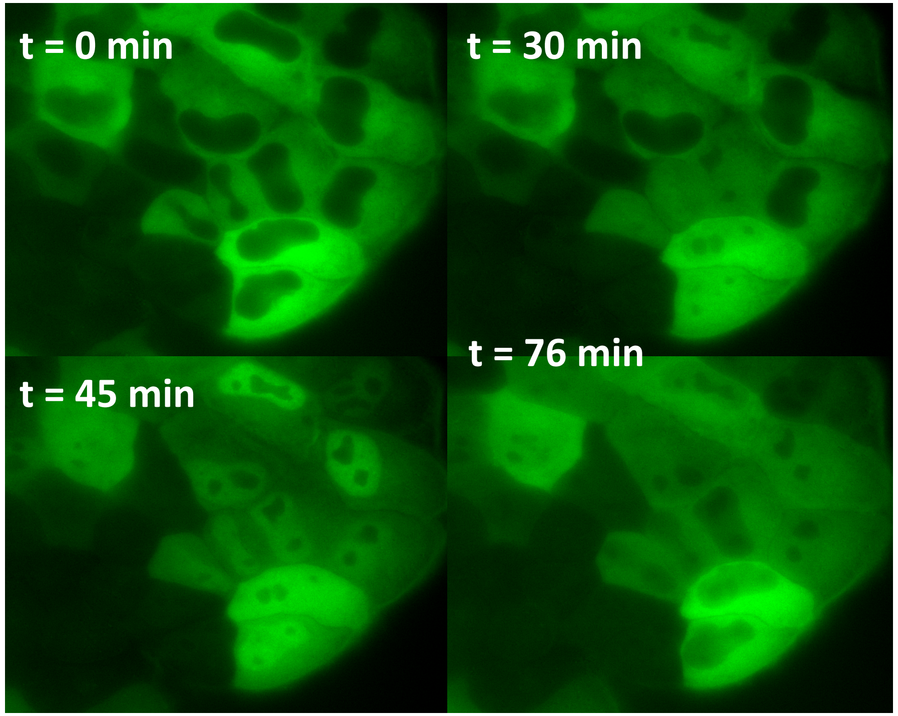
Future development

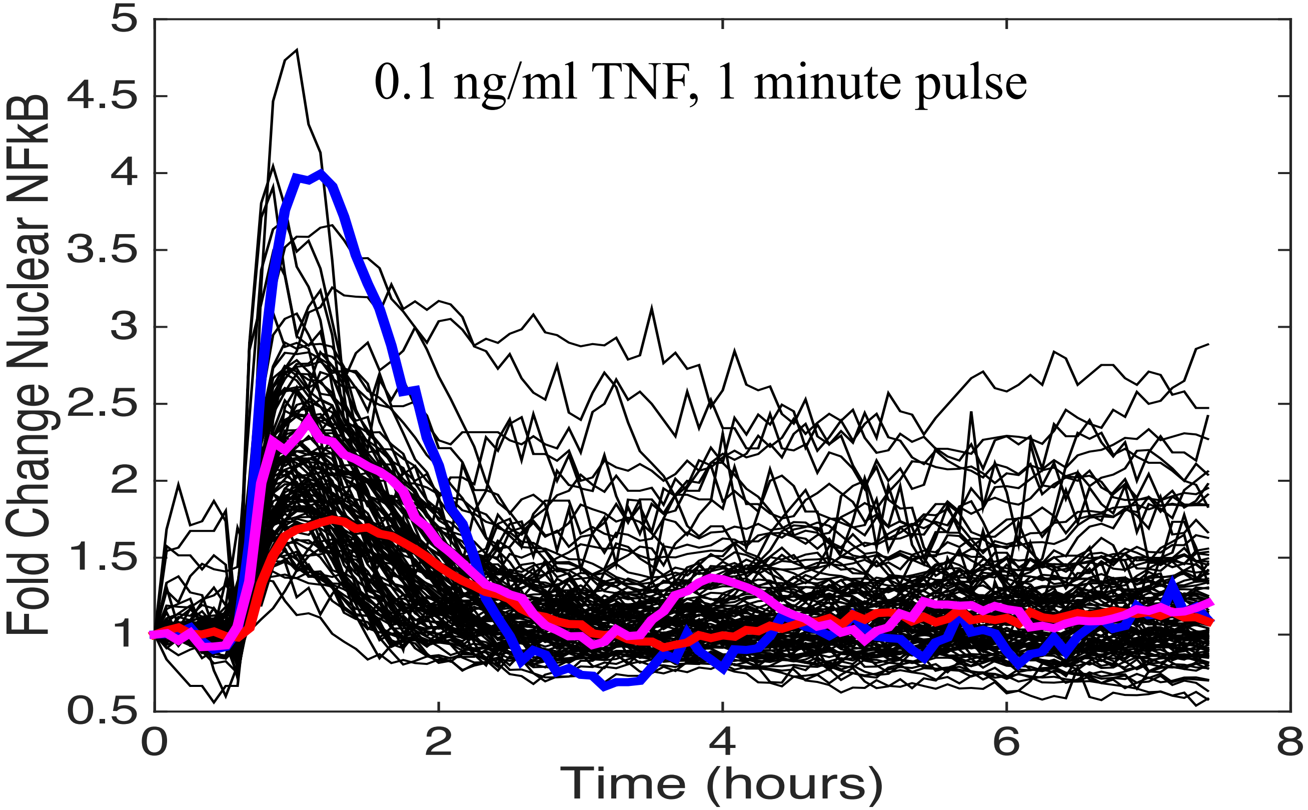
- Improving tool accessibility and model communication:

SBML support enables conversion of models in different formats to BNG to be used with the parameter estimation tools being developed

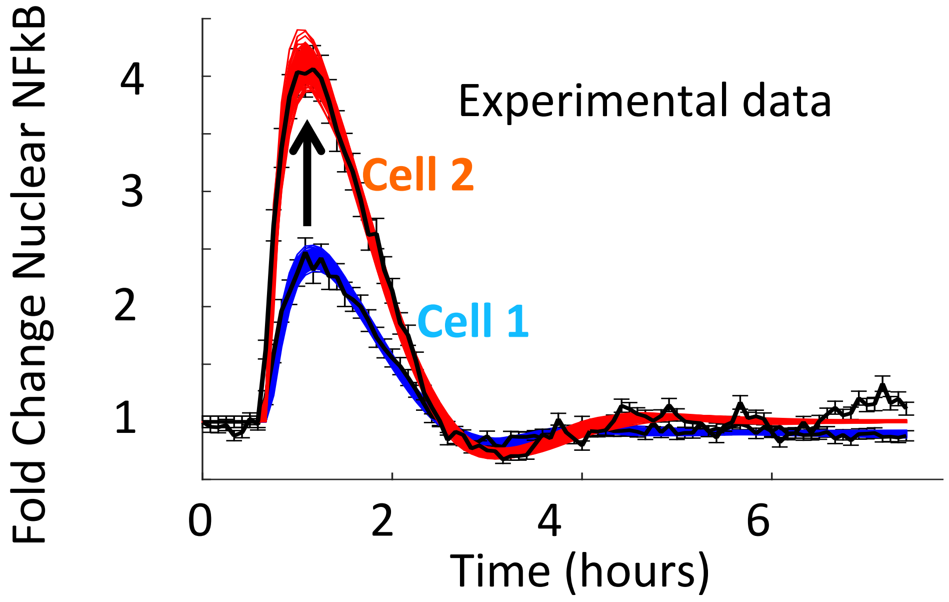
Currently developing support for SBML-Multi

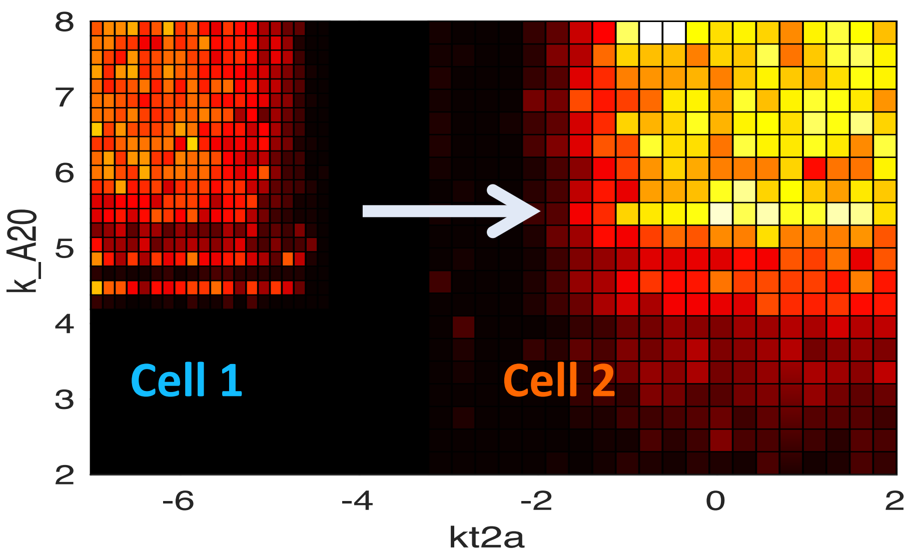
Applying parameter estimation to understand variability in NFkB signaling





Applying parameter estimation to understand variability in NFkB signaling





Acknowledgement

Advisors

Dr. James Faeder
Dr. Robin Lee

Faeder Lab

Kunal Aggarwal
Cihan Kaya
Dr. John Sekar
Dr. Jose-Juan Tapia
Dr. Robert Sheehan
Dr. Justin Hogg
Dr. Ali Sinan Saglam

Lee Lab

Chaitanya Mokashi
Gabriel Kowalczyk
David Schipper
Yue Guo
Dr. Augustin Cruz
Dr. Qiuhong Zhang

Funding sources

NIGMS-funded (P41GM103712) National Center for Multiscale Modeling of Biological Systems (MMBioS)

NIH grant R35-GM119462 to RECL