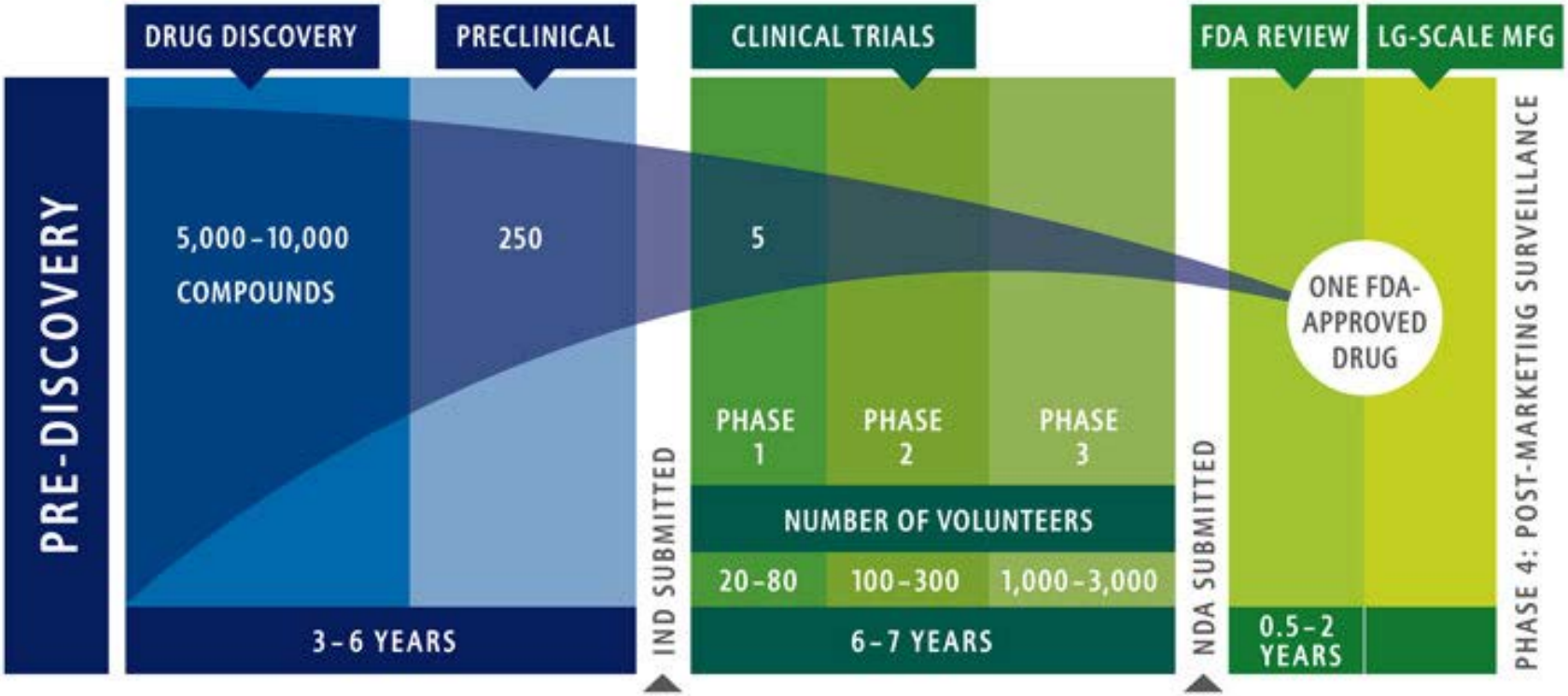
Systems biology: from models to drug candidates

Jan Hasenauer

Research Group Data-driven Computational Modeling,
Institute of Computational Biology, Helmholtz Zentrum München

Department of Mathematics, University of Technology Munich

Drug Discovery and Development: A LONG, RISKY ROAD



Source: Pharmaceutical Research and Manufacturers of America

How can this fraction be improved?

$\Rightarrow$ Better utilisation of public prior knowledge and information obtained during the drug development process.

Available resources / databases

Databases for experiment data:

- The Human Genome Atlas
- The Human Proteome Atlas
- The Cancer Genome Atlas (TCGA)
- The Genotype-Tissue Expression (GTEx) Project
- etc.

Databases for molecular interaction:

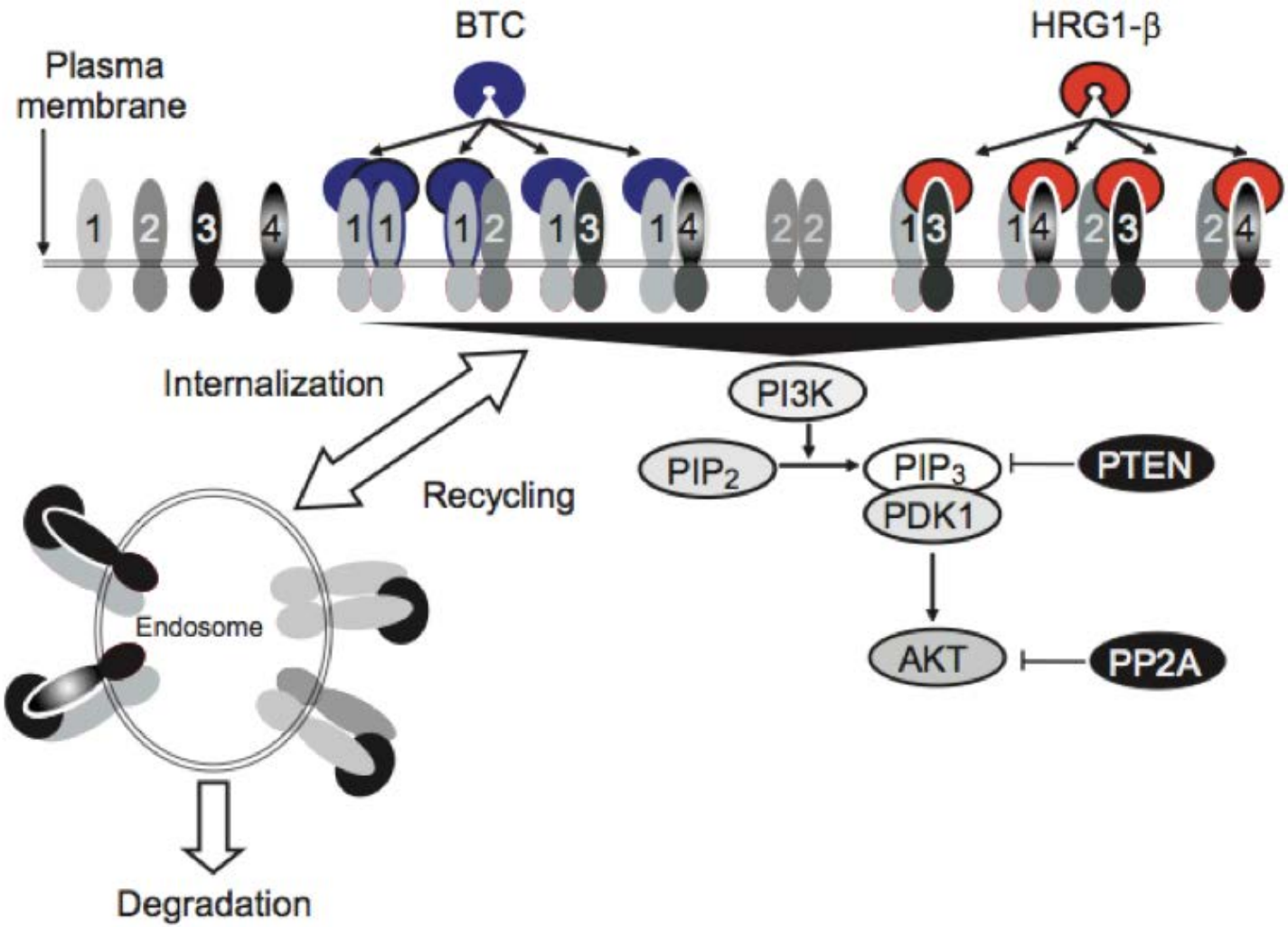
- REACTOME: curated pathway database
- STRING: database of protein-protein interactions
- etc.

Databases for mathematical models:

- BioModels
- cellML
- etc.

Massive datasets, but how can they be integrated / exploited?

Model-based drug design



B. Schöberl et al., Science Signaling, 2009

Outline

Outline

Types of models

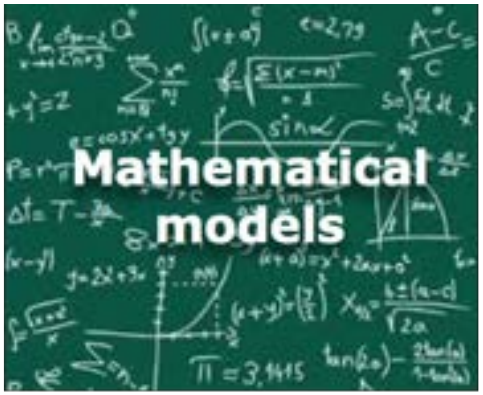
Model 'models'

Mouse 'models'

Car 'models'

Shoe 'models'

Mathematical models



statistical models

mechanistic models

A model is "a representation of the **essential aspects** of an existing system (or a system to be constructed) which **presents knowledge** of that system in **usable form**."

P. Eykhoff, Wiley and Sons, (1974)

Some applications of mathematical models

Integration

- Use of heterogeneous datasets
- Storage of available information
- Development of coherent description

Analysis

- Identification of mechanisms
- Assessment of causalities

Hypothesis testing / validation

- Comparison of competing hypotheses
- Verification of predicted model properties

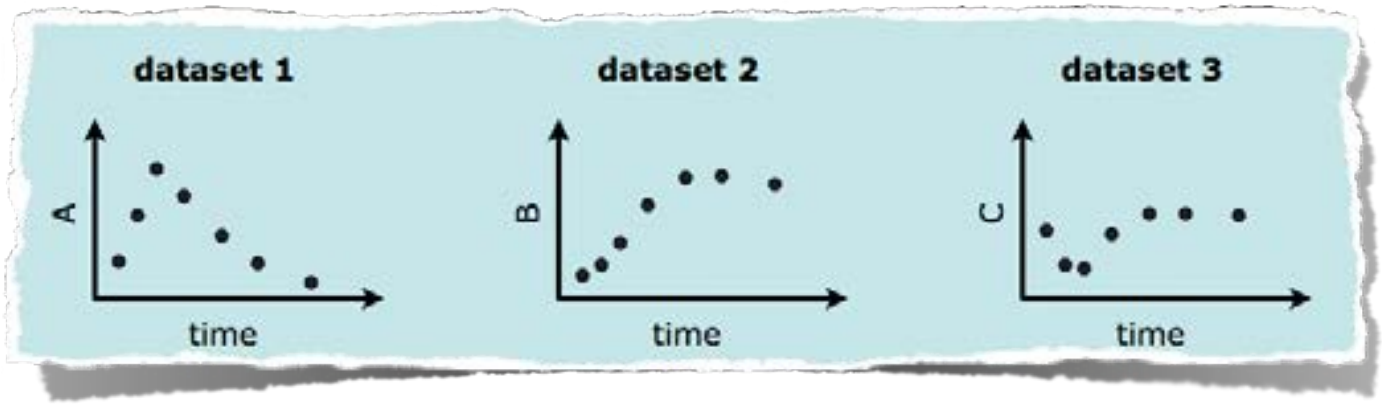
Prediction

- Prediction of response for new condition / treatment
- Optimisation of condition / treatment

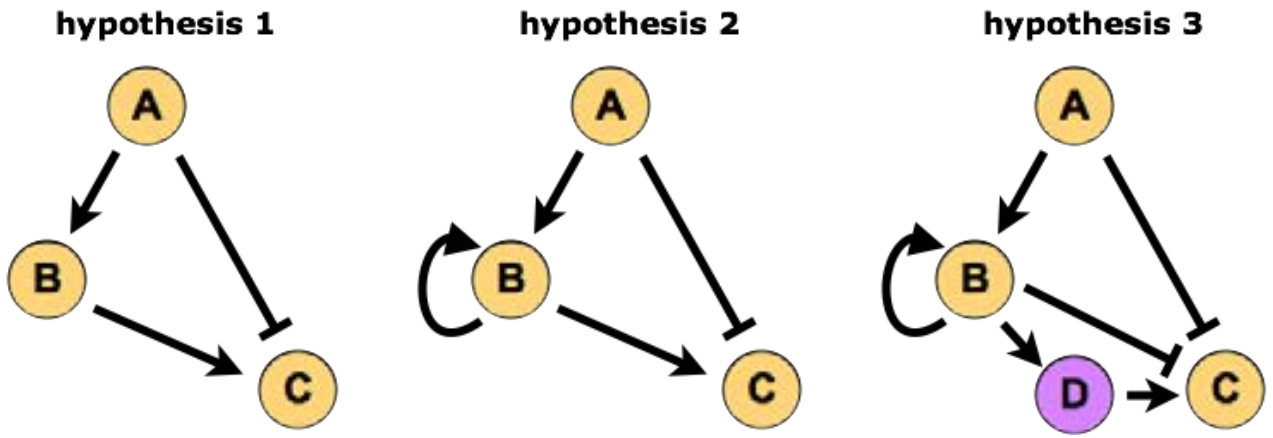
$\rightarrow$ **identification of drug targets or combination therapies**

Comparison of competing hypotheses

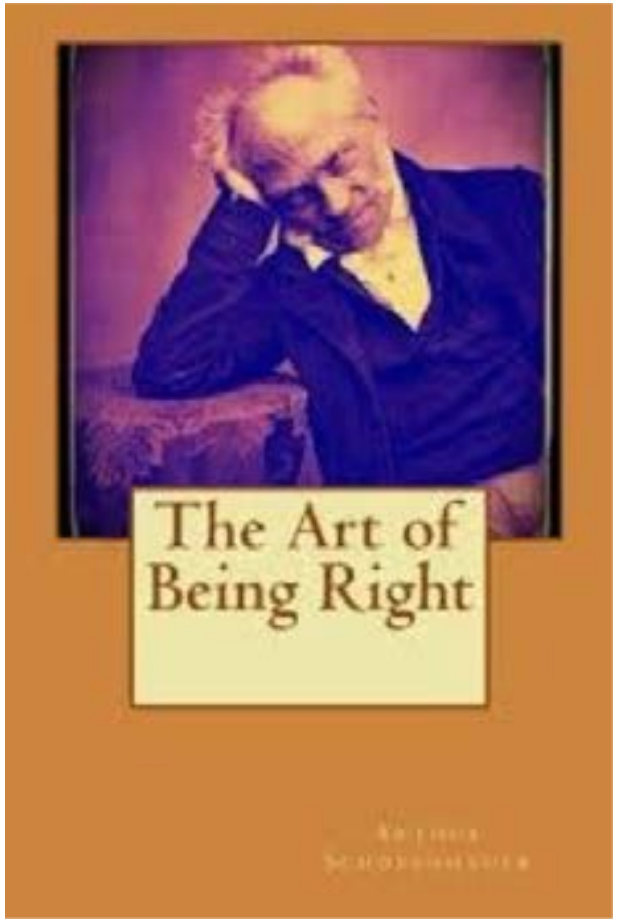
Data



Biological hypothesis



Consistent reasoning



The Art of Being Right:
38 Ways to Win an Argument

is an acidulous and sarcastic treatise written by the German philosopher Arthur Schopenhauer in sarcastic deadpan.

Key challenge of verbal arguments:

Language is imprecise, vage and leaves room for interpretation.

Outline

Biochemical species and reactions

Chemical species

A chemical species X_i, $i = 1, \ldots, n_x$ is an ensemble of chemically identical molecular entities.

Chemical reactions

A chemical reaction R_j, $j = 1, \ldots, n_r$ is a process that results in the interconversion of chemical species.

$$\underbrace{\sum_{i=1}^{n_t} \nu_{i,j} X_i}_{\text{educts/reactants}} \quad \overset{k_{+j}}{\underset{k_{-j}}{\rightleftharpoons}} \quad \underbrace{\sum_{i=1}^{n_t} \eta_{i,j} X_i}_{\text{products}}$$

with

- stoichiometric coefficients $\nu_{i,j}, \eta_{i,j} \in \mathbb{N}_0$ and
- reaction rate constants $k_{+j}, k_{-j} \in \mathbb{R}_+$.

Reaction kinetics

Chemical reaction R_j:

$$\underbrace{\sum_{i=1}^{n_t} \nu_{i,j} X_i}_{\text{educts/reactants}} \quad \underset{k_{-j}}{\overset{k_{+j}}{\rightleftharpoons}} \quad \underbrace{\sum_{i=1}^{n_t} \eta_{i,j} X_i}_{\text{products}}$$

Mass action kinetics

The reaction rate $v_j(x)$ of reaction R_j assuming mass action kinetics is

$$v_j(x) \quad = \quad k_{+j} \prod_{i=1}^{n_x} x_i^{\nu_{i,j}} - k_{-j} \prod_{i=1}^{n_x} x_i^{\eta_{i,j}}$$

with x_i denoting the concentration of X_i.

Dynamics of biochemical reaction networks

Reaction Rate Equation (RRE)

The temporal evaluation of the concentration vector x is captured by

$$\begin{aligned} \frac{dx(t,\theta)}{dt} &= f(x(t,\theta),\theta,t), \qquad & x(0) = x_0(\theta) \\ &= S \cdot v(x(t,\theta),\theta,t), \end{aligned}$$

with

- stoichiometric matrix

$$S = \begin{pmatrix} \eta_{1,1} - \nu_{1,1} & \eta_{1,2} - \nu_{1,2} & \cdots \\ \eta_{2,1} - \nu_{2,1} & \eta_{2,2} - \nu_{2,2} & \cdots \\ \vdots & \vdots & \end{pmatrix}$$

- reaction flux vector $v(x,\theta,t)$
- initial condition $x_0(\theta)$ (e.g., steady state)
- parameter vector θ (e.g., kinetic rates k_{-j} and k_{+j})

Example: Enzyme kinetics (1)

$$R_1: \quad A + E \overset{k_{+1}}{\underset{k_{-1}}{\rightleftharpoons}} AE$$

$$R_2: \quad AE \overset{k_{+2}}{\rightarrow} B + E$$

with biochemical species

- A: substrate
- E: enzyme
- AE: substrate-enzyme-complex
- B: product

and rate parameters k_{+1}, k_{-1} and k_{+2}.

Questions:

1. What are the state variables?
2. What are the reaction fluxes?
3. What is the stoichiometric matrix?
4. What is the ODE model?

Example: Enzyme kinetics (2)

State variables

- x_1: concentration of substrate
- x_2: concentration of enzyme
- x_3: concentration of substrate-enzyme-complex
- x_4: concentration of product

Reaction fluxes

$$v_1(x,\theta) = k_{+1}x_1x_2 - k_{-1}x_3$$
$$v_2(x,\theta) = k_{+2}x_3$$

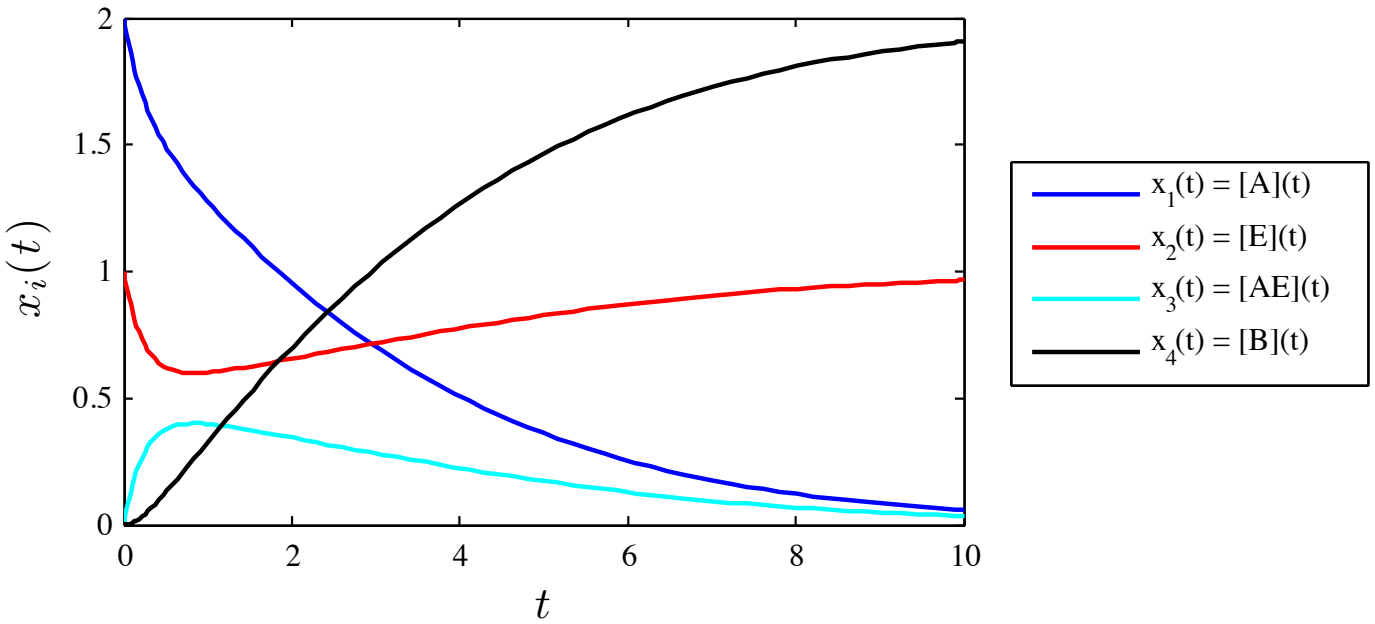
ODE model

$$\frac{d}{dt}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} -1 & 0 \\ -1 & +1 \\ +1 & -1 \\ 0 & +1 \end{pmatrix} \begin{pmatrix} v_1(x,\theta) \\ v_2(x,\theta) \end{pmatrix} = \underbrace{\begin{pmatrix} -k_{+1}x_1x_2 + k_{-1}x_3 \\ -k_{+1}x_1x_2 + k_{-1}x_3 + k_{+2}x_3 \\ +k_{+1}x_1x_2 - k_{-1}x_3 - k_{+2}x_3 \\ +k_{+2}x_3 \end{pmatrix}}_{=f(x)}$$

Numerical simulation

Aim

Evaluation of the time-dependent state of a process for a given set of rate parameters and initial conditions.



The simulation results

- provide a prediction of the dynamics of the process and
- and depend on the parameters of the process.

Euler's explicit method for solving ODEs

Approach

Forward finite-difference approximation of time derivative,

$$\frac{dx(t)}{dt} \approx \frac{x(t+\Delta_t) - x(t)}{\Delta t},$$

with time step Δ_t

For $x_k = x(k\Delta_t)$, we obtain the explicit method

$$\frac{x_{k+1} - x_k}{\Delta_t} = Sv(x_k) \quad \Leftrightarrow \quad x_{k+1} = x_k + \Delta_t Sv(x_k)$$

Euler's explicit method

Input: $x_0 \leftarrow x_0(\theta)$
for $k = 0, \ldots, k_{\max}$ **do**
  $x_{k+1} \leftarrow x_k + \Delta_t Sv(x_k)$
end

Advances methods for solving ODEs

Single-step methods:

- Heun (2nd order)
- Runge-Kutta (arbitrary order)

Multi-step methods:

- Adams-Bashforth (arbitrary order)
- Backward differentiation (arbitrary order)

Multi-derivative methods:

- Hermite-Obreschkoff methods
- Fehlberg methods

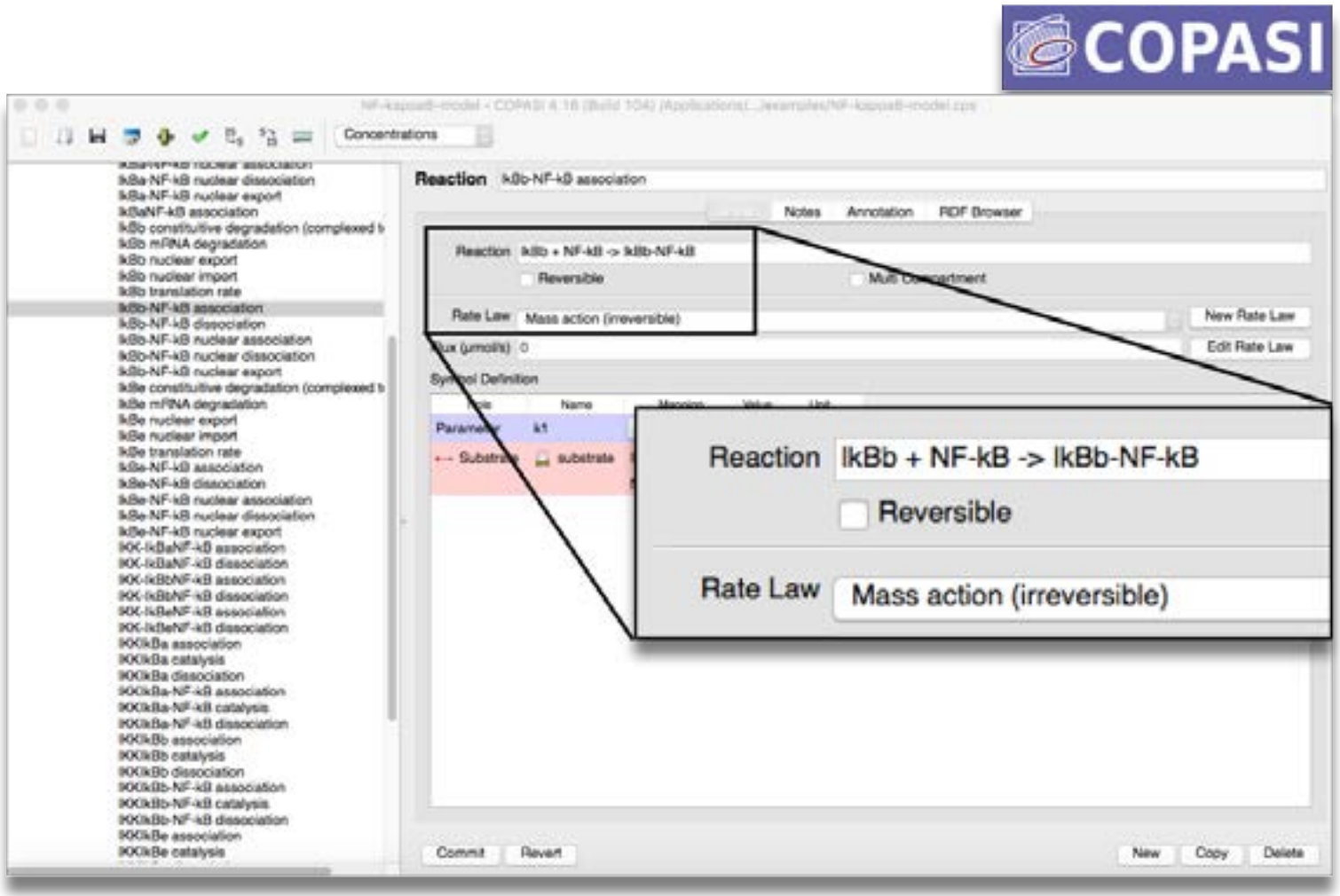
Software for numerical toolboxes

SUNDIALS: SUite of Nonlinear and DIfferential/ALgebraic Equation

- CVODES for ODEs
- IDAS for DAEs

Software tools

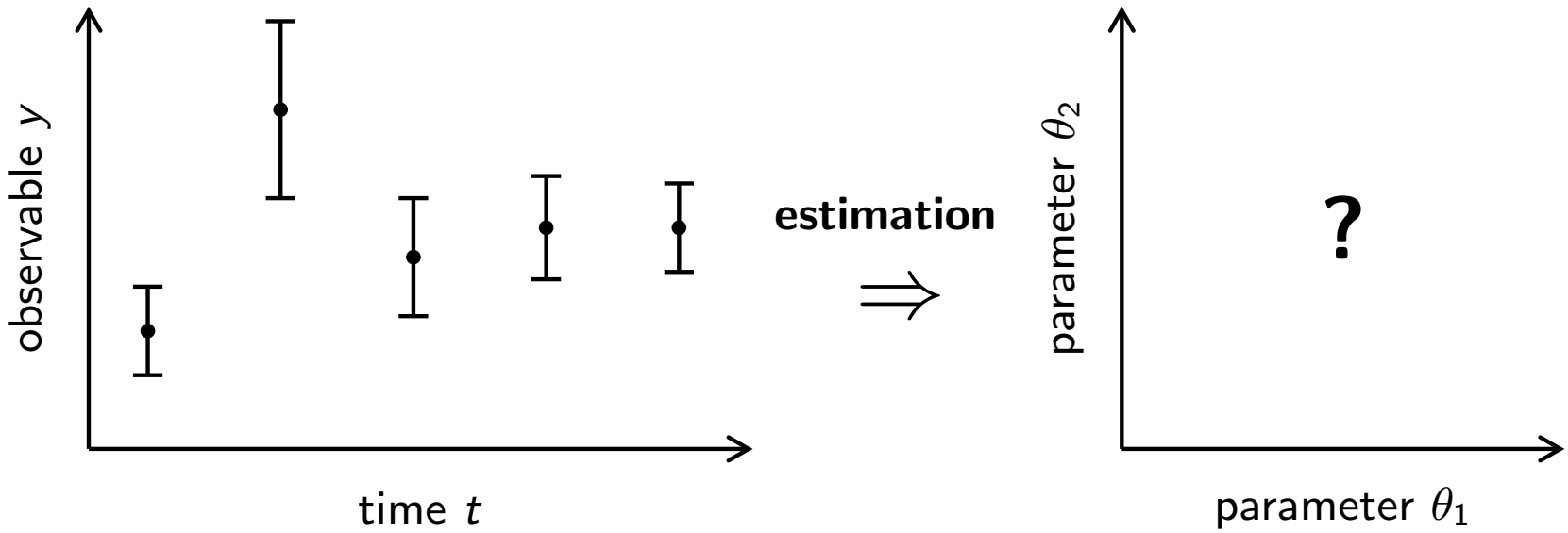
- **CellDesigner:** Editor for drawing gene-regulatory and biochemical networks
- **Copasi:** Software for modelling, simulation and analysis of biochemical networks



Outline

Parameter estimation

The prediction of mechanistic models depends in the model parameters θ (e.g. rate constants k_{+j} and k_{-j}). The parameters are however mostly unknown.



Inverse problem

Estimation of model parameters from available experimental data.

Observable in biochemical processes

Observable

A function of state variable and/or parameters which can be measured is called **observable**,

$$y(t, \theta) = h(x(t, \theta), \theta, t).$$

with $h : \mathbb{R}^{n_x} \times \mathbb{R}^{n_\theta} \times \mathbb{R}_+ \mapsto \mathbb{R}^{n_y}$.

Examples for observation functions:

- absolute data: $y_j = x_l$ or $y_j = x_{l_1} + x_{l_2}$
- relative data: $y_j = s_l x_l$ or $y_j = s_l \left(x_{l_1} + x_{l_2}\right)$
- saturated measurement: $y_j = \frac{x_l}{K_l + x_l}$

Examples for experimental approaches:

- Western blots
- Flow cytometry
- RNAseq

Experimental data

Measurement

An experimental assessment of an observable is called **measurement**. Measurements are in general subject to noise yielding noise-corrupted measurement data,

$$\bar{y} \sim p(\bar{y}|y, \varphi),$$

with noise parameters φ.

Example:

- additive noise:
  - independent: $\bar{y}_{jk} = y_j(t_k, \theta) + \epsilon_{jk}$ with $\epsilon_{jk} \sim \mathcal{N}(0, \sigma_{jk}^2)$
  - correlated noise: $\bar{y}_k = y(t_k, \theta) + \epsilon_{jk}$ with $\epsilon_k \sim \mathcal{N}(0, \Sigma_k)$
- multiplicative noise:
  - standard formulation: $\bar{y}_{jk} = y_j(t_k, \theta) \cdot \nu_{jk}$ with $\nu_{jk} \sim \log \mathcal{N}(0, \sigma_{jk}^2)$
  - transformed output: $\log \bar{y}_{jk} = \log y_j(t_k, \theta) + \epsilon_{jk}$ with $\epsilon_{jk} \sim \mathcal{N}(0, \sigma_{jk}^2)$

Experimental data

$$\mathcal{D} = \{(t_k, \bar{y}_k)\}_{k=1}^{n_t}$$

Statistical description

Likelihood

The conditional probability of data given the model

$$p(\mathcal{D}|\theta) = p(\bar{y}|y(\cdot,\theta),\varphi(\theta)) \qquad \text{“general model”}$$

$$= \prod_{k=1}^{n_t}\prod_{j=1}^{n_y} p(\bar{y}_{jk}|y_j(t_k,\theta),\varphi_{jk}(\theta)) \qquad \text{“assuming independence”}$$

Example:

- Independent, additive normally distributed measurement noise:

$$p(\mathcal{D}|\theta) = \prod_{k=1}^{n_t}\prod_{j=1}^{n_y} \frac{1}{\sqrt{2\pi}\sigma_{jk}(\theta)} \exp\left\{-\frac{1}{2}\left(\frac{\bar{y}_{jk}-y_j(t_k,\theta)}{\sigma_{jk}(\theta)}\right)^2\right\}$$

- Independent, multiplicative log-normally distributed measurement noise:

$$p(\mathcal{D}|\theta) = \prod_{k=1}^{n_t}\prod_{j=1}^{n_y} \frac{1}{\sqrt{2\pi}\bar{y}_{jk}\sigma_{jk}(\theta)} \exp\left\{-\frac{1}{2}\left(\frac{\log\bar{y}_{jk}-\log y_j(t_k,\theta)}{\sigma_{jk}(\theta)}\right)^2\right\}$$

Remark: The factor $1/\bar{y}_{jk}$ is relevant for model selection.

Maximum Likelihood & Maximum A Posteriori estimator

Maximum Likelihood (ML) estimator

The ML estimate $\theta^{ml} \in \Omega \subseteq \mathbb{R}_+^{n_\theta}$ maximises the likelihood,

$$\theta^{ml} = \arg\max_{\theta \in \Omega} p(\mathcal{D}|\theta), \quad \text{subject to } \mathcal{M}(\theta).$$

Bayes's theorem:

$$p(\theta|\mathcal{D}) = \frac{p(\mathcal{D}|\theta)p(\theta)}{p(\mathcal{D})}$$

with

- $p(\theta|\mathcal{D})$: posterior probability of parameters given data
- $p(\mathcal{D}|\theta)$: conditional probability of data given model / likelihood
- $p(\theta)$: prior probability
- $p(\mathcal{D})$: marginal probability of data

Maximum A Posterior (MAP) estimator

The MAP estimate $\theta^{ml} \in \Omega \subseteq \mathbb{R}_+^{n_\theta}$ maximises the posterior probability,

$$\theta^{map} = \arg\max_{\theta \in \Omega} \{p(\theta|\mathcal{D}) \propto p(\mathcal{D}|\theta)p(\theta)\}, \quad \text{subject to } \mathcal{M}(\theta).$$

Reformulation of optimisation problem

Log-transformation of likelihood function

The ML estimate is the minimiser of the negative log-likelihood function $J(\theta)$,

$$\theta^{ml} = \arg\min_{\theta \in \Omega} \{J(\theta) := -\log p(\mathcal{D}|\theta)\}, \quad \text{subject to } \mathcal{M}(\theta).$$

Properties:

- Maxima of $p(\mathcal{D}|\theta)$ are minima of $J(\theta)$.
- $J(\theta)$ usually "(locally) more convex" than $-p(\mathcal{D}|\theta)$.
- Evaluation of $J(\theta)$ numerically more robust as sum instead of product.

Log-transformation of parameters

For positive parameters $\theta \geq 0$, the ML estimate $\theta^{ml} = \exp(\xi^{ml})$ can be obtained by optimising the log-transformed parameter $\xi = \log(\theta)$,

$$\xi^{ml} = \arg\min_{\xi \in \log(\Omega)} J(\exp(\xi)), \quad \text{subject to } \mathcal{M}(\exp(\xi)).$$

Properties: $J(\exp(\xi))$ usually "(locally) more convex" than $J(\theta)$.

Relation: Maximum Likelihood (ML) $\leftrightarrow$ Least squares (LS)

Negative log-likelihood function for independent, additive normally distributed measurement noise:

$$p(\mathcal{D}|\theta) = \prod_{k=1}^{n_t} \prod_{j=1}^{n_y} \frac{1}{\sqrt{2\pi}\sigma_{jk}(\theta)} \exp\left\{ -\frac{1}{2} \left(\frac{\bar{y}_{jk} - y_j(t_k, \theta)}{\sigma_{jk}(\theta)} \right)^2 \right\}$$

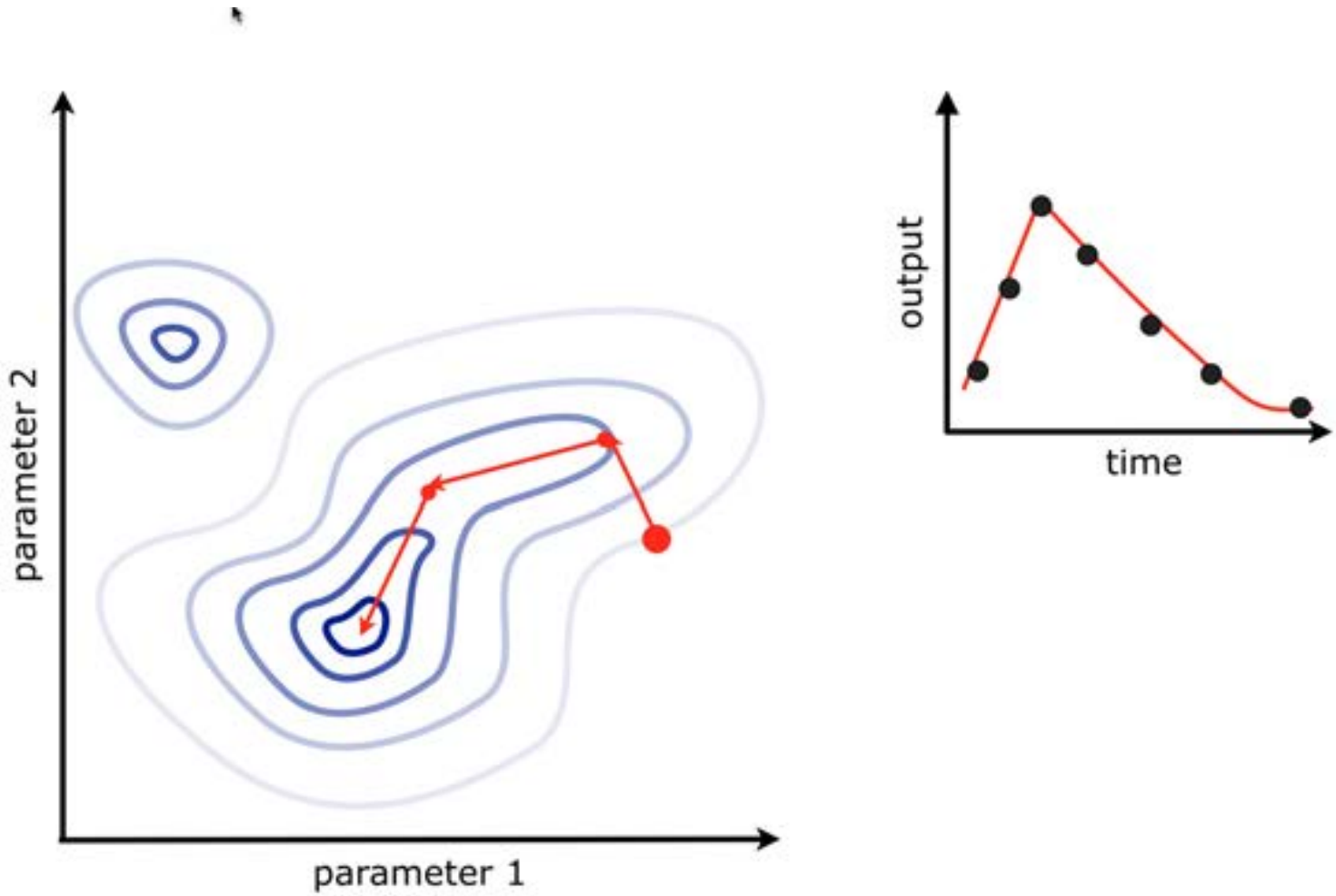
$$\Rightarrow J(\theta) = -\log p(\mathcal{D}|\theta) = \frac{1}{2} \sum_{k=1}^{n_t} \sum_{j=1}^{n_y} \log(2\pi\sigma_{jk}^2(\theta)) + \left(\frac{\bar{y}_{jk} - y_j(t_k, \theta)}{\sigma_{jk}(\theta)} \right)^2$$

For known noise variance $\sigma_{jk}^2(\theta) \neq \mathrm{fnc}(\theta)$, the term $\log(2\pi\sigma_{jk}^2(\theta))$ is constant and the objective function can be simplified to

$$J(\theta) = \frac{1}{2} \sum_{k=1}^{n_t} \sum_{j=1}^{n_y} \left(\frac{\bar{y}_{jk} - y_j(t_k, \theta)}{\sigma_{jk}} \right)^2 .$$

$\rightarrow$ objective function of the **weighted least squares estimator**

Illustration of local optimisation



Local optimisation

Optimisation problem

$$\min_{\theta} J(\theta)$$

Goal of local optimiser:

Construction of sequence $\theta^{(0)}$, $\theta^{(1)}, \theta^{(2)}, \ldots$ along which objective function decreases, $J(\theta^{(0)}) > J(\theta^{(1)}) > J(\theta^{(2)}) > \ldots$, using local properties of the objective function.

Optimizer path construction

$$\theta^{(k+1)} = \theta^{(k)} + t^{(k)} \Delta^{(k)}$$

in which

- search direction $\Delta^{(k)} \in \mathbb{R}^{n_t}$ and
- step length $t^{(k)} \in \mathbb{R}_+$.

Steepest descent methods

Search direction

Direction with strongest (local) decrease for given vector norm $||\cdot||$:

$$\Delta^{(k)} = \arg \min_{v, ||v||=1} \nabla_\theta J(\theta^{(k)})^T v.$$

Special cases:

- Euclidean norm $||v|| := v^T v$:

  $$\Delta^{(k)} = -\nabla_\theta J(\theta^{(k)}) \quad \rightarrow \quad \text{gradient descent}$$

- General quadratic norm $||v|| := v^T P v$ with positive definite matrix P:

  $$\Delta^{(k)} = -P^{-1} \nabla_\theta J(\theta^{(k)})$$

Step length

Minimisation of $J(\theta)$ along $\theta = \theta^{(k)} + t^{(k)} \Delta^{(k)}$ using

- exact line search $t^{(k)} := \arg \min_{t>0} J(\theta^{(k)} + t\Delta^{(k)})$ or
- backtracking line search ($\rightarrow$ successive geometric reduction of $t^{(k)}$).

Implementation of steepest descent methods

Input: starting point $\theta^{(0)}$
set $k \leftarrow 0$
repeat
  calculate steepest descent direction $\Delta^{(k)}$
  calculate step size $t^{(k)}$ by line search
  set $\theta^{(k+1)} \leftarrow \theta^{(k)} + t^{(k)}\Delta^{(k)}$
  set $k \leftarrow k + 1$
until *stopping criterion holds*;

Common stopping criteria

- Small change of objective function: $||J(\theta^{(k+1)}) - J(\theta^{(k)})||_2 < \epsilon_J$
- Small change of parameters: $||\theta^{(k)} - \theta^{(k-1)}||_2 < \epsilon_\theta$
- Sumber of function evaluations: $k > N$

Remark: The calculation of the descent direction and the step size requires the numerical simulation of the ODE.

Newton-Raphson and trust-region method

Idea: Next point as optimum of local approximation

$$J(\theta) \approx J(\theta^{(k)}) + \nabla_\theta J|_{\theta^{(k)}} \left(\theta - \theta^{(k)}\right) + \frac{1}{2}\left(\theta - \theta^{(k)}\right)^T \left(\nabla^2_\theta J|_{\theta^{(k)}}\right) \left(\theta - \theta^{(k)}\right)$$

Newton-Raphson method

$$\theta^{(k+1)} = \theta^{(k)} - \left(\nabla^2_\theta J|_{\theta^{(k)}}\right)^{-1} \nabla_\theta J|_{\theta^{(k)}}$$

Properties:

- (Local) quadratic convergence
- Positive definite Hessian required $\rightarrow$ *Levenberg-Marquardt algorithm*
- Local approximation might not be satisfactory

Trust-region method

In trust-region methods the next point $\theta^{(k+1)}$ is the minimiser of the quadratic approximation of the objective function with the trust region,

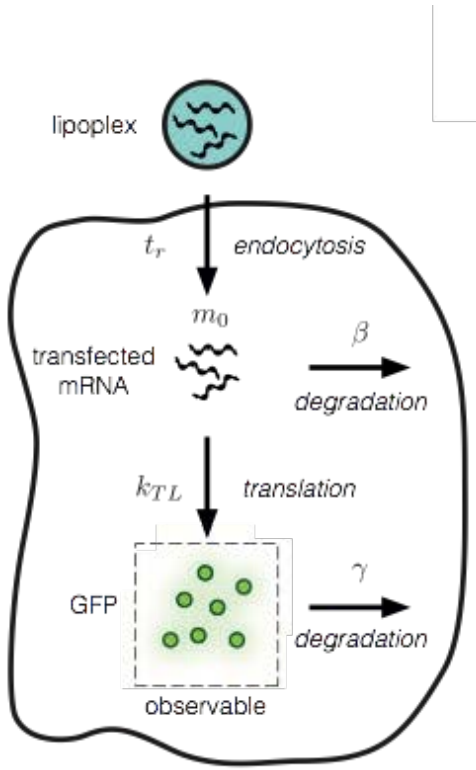
$$\theta^{(k+1)} = \theta^{(k)} + \Delta \quad \text{with} \quad \Delta = \underset{||\Delta||_2 \leq r^{(k)}}{\arg\min} \nabla_\theta J|_{\theta^{(k)}} \Delta + \frac{1}{2}\Delta^T \left(\nabla^2_\theta J|_{\theta^{(k)}}\right) \Delta,$$

with the trust-region radius $r^{(k)}$ being updated based on the prediction accuracy.

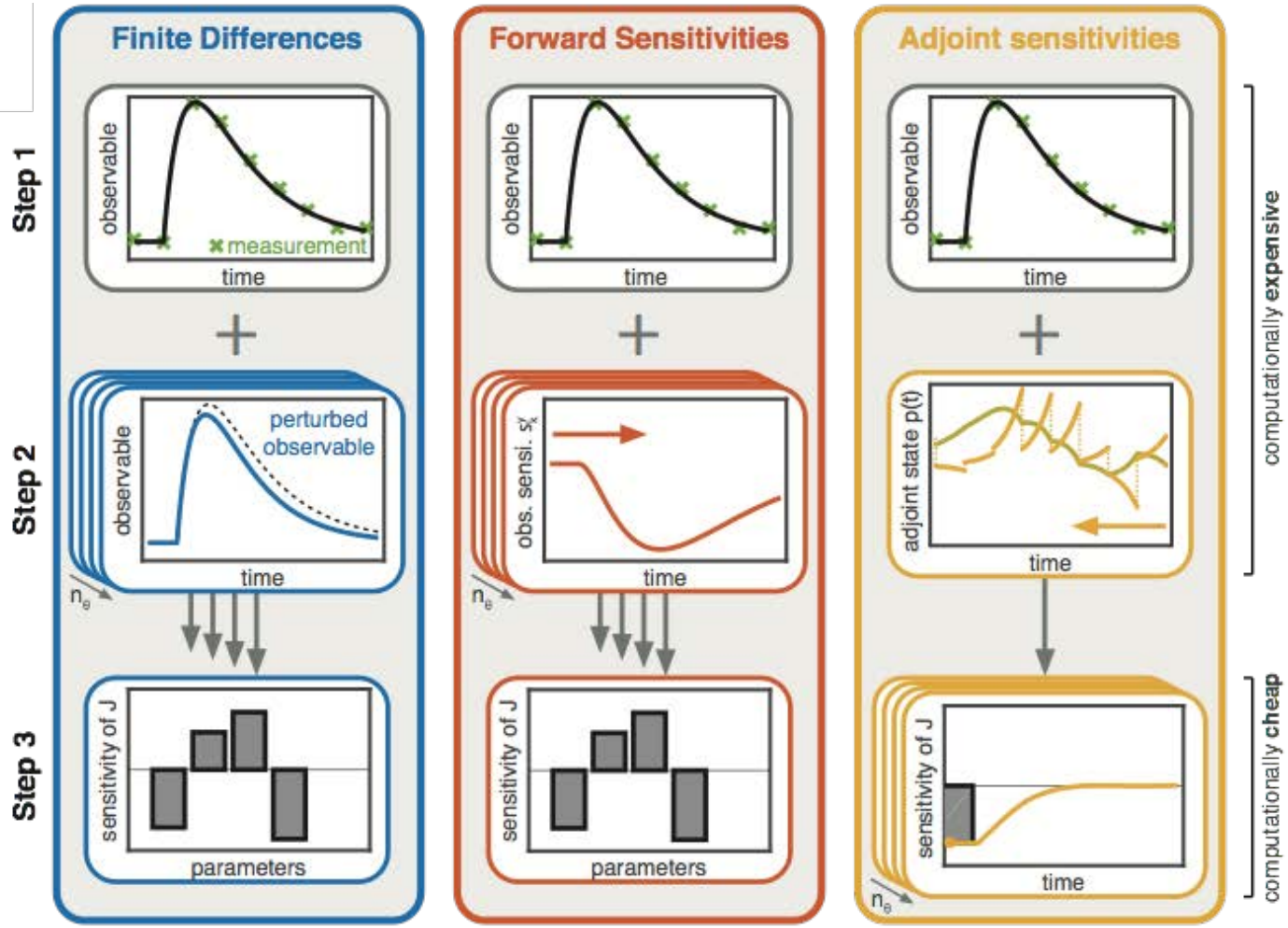
Implementation of trust-region methods

Input: starting point $\theta^{(0)}$, initial trust-region radius $r^{(0)}$
Input: tuning parameters $\gamma_2 > 1 > \gamma_1 > 0$ and $1 > \nu > 0$
set $k \leftarrow 0$
repeat
  calculate gradient and Hessian
  calculate update step $\Delta \leftarrow \underset{||\Delta||_2 \leq r^{(k)}}{\arg\min} \nabla_\theta J|_{\theta^{(k)}} \Delta + \frac{1}{2} \Delta^T \left(\nabla^2_\theta J|_{\theta^{(k)}} \right) \Delta$
  calculate prediction accuracy $\rho \leftarrow \dfrac{J(\theta^{(k)} + \Delta) - J(\theta^{(k)})}{\nabla_\theta J|_{\theta^{(k)}} \Delta + \frac{1}{2} \Delta^T \left(\nabla^2_\theta J|_{\theta^{(k)}} \right) \Delta}$
  if $\rho \geq \nu$ **then**
    set $\theta^{(k+1)} \leftarrow \theta^{(k)} + \Delta$
    set $r^{(k+1)} \leftarrow \gamma_2 r^{(k)}$
  else
    set $\theta^{(k+1)} \leftarrow \theta^{(k)}$
    set $r^{(k+1)} \leftarrow \gamma_1 r^{(k)}$
  end
  set $k \leftarrow k + 1$
until *stopping criterion holds*;

Illustration of sensitivity analysis methods

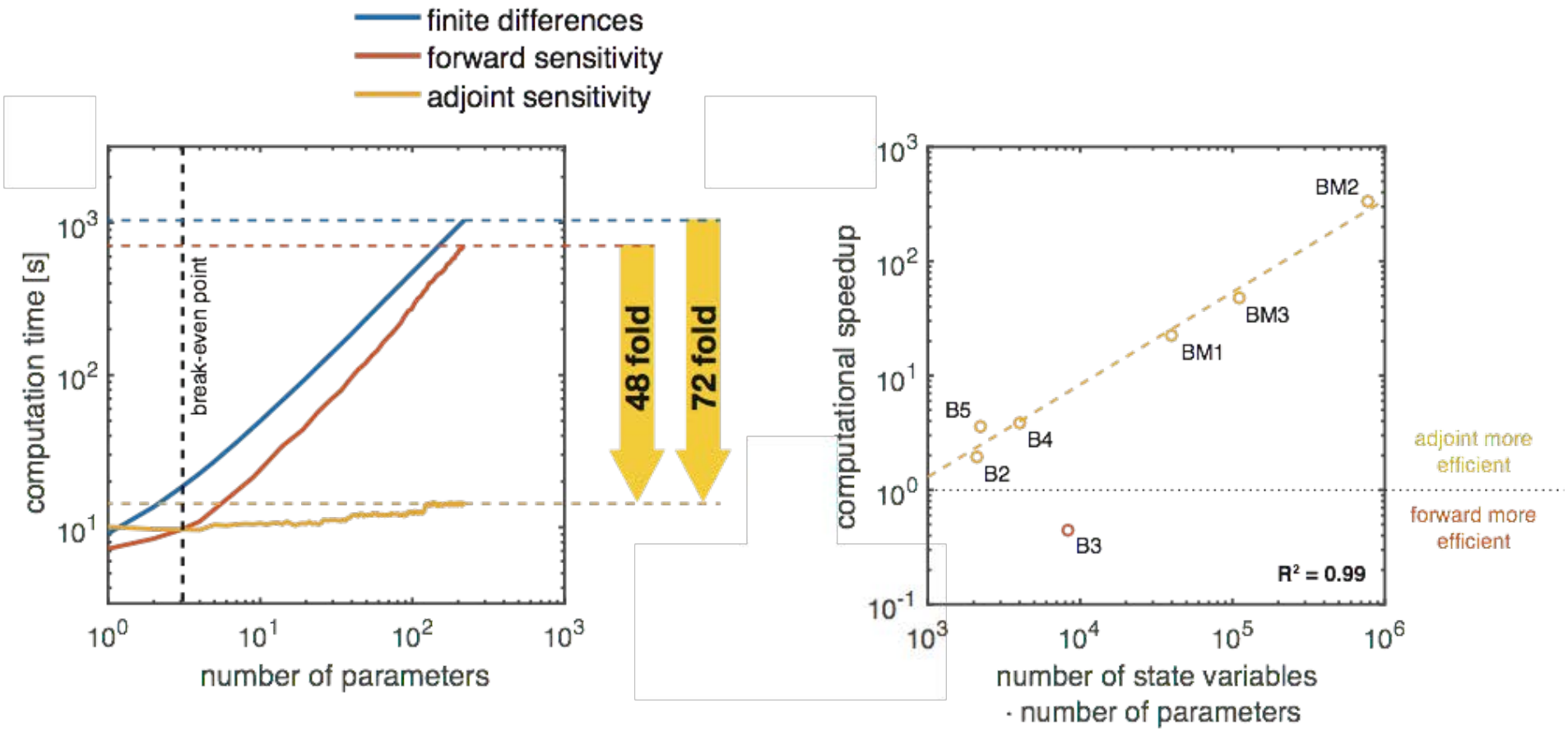


$$\frac{[\text{mRNA}]}{dt} = -\beta[\text{mRNA}] + m_0\delta(t - t_r)$$
$$\frac{d[\text{GFP}]}{dt} = k_{TL}[\text{mRNA}] - \gamma[\text{GFP}]$$
$$y = [\text{GFP}]$$



Remark: If you are working with log-parameter, do not forget the chain rule!

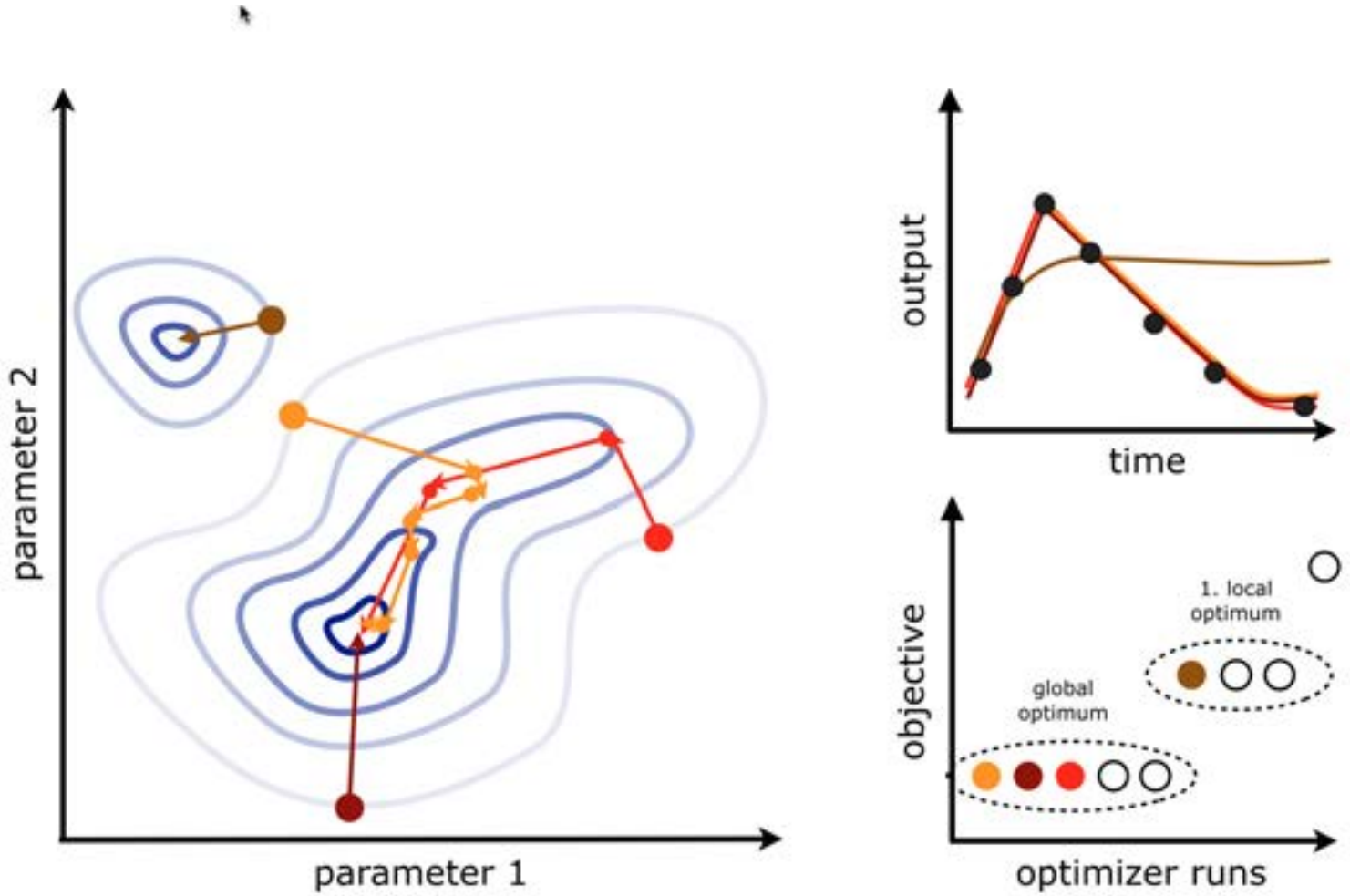
Comparison of sensitivity analysis methods



Properties:

- Forward and adjoint sensitivity analysis provide high accuracy.
- Adjoint sensitivity analysis provide good scalability.

Illustration of global optimisation using multi-start local optimisation



Global optimisation using multi-start local optimisation

Properties of good local optimiser: Robust and efficient convergence to the local optimum in those basin of attraction the local optimiser has been it initialised.

Multi-start local optimisation

Idea: Coverage of all basins of attraction. $\Rightarrow$ *global optimal point*

Algorithm:

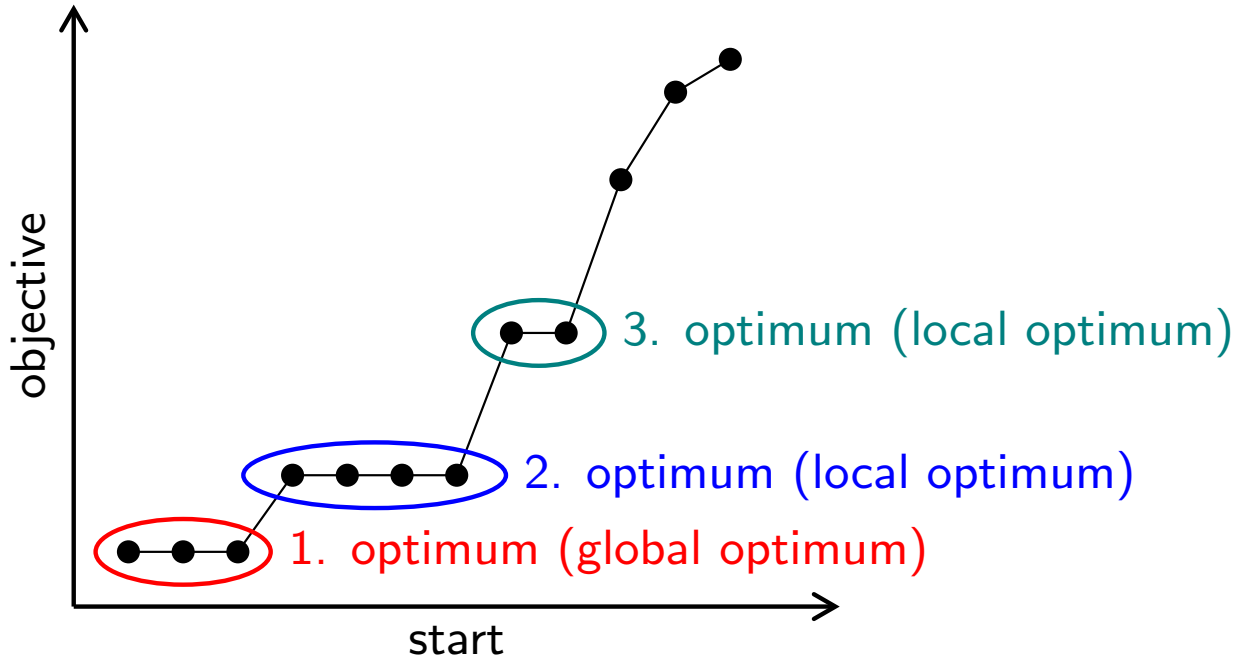
Input: Number of starting points M, probability distribution of initial condition $p(\hat{\theta}^{(0)})$
for $i = 1$ *to* M **do**
  Draw initial point $\hat{\theta}^{(i0)} \sim p(\hat{\theta}^{(0)})$.
  Compute locally optimal point $\hat{\theta}^{(i*)}$ starting from $\hat{\theta}^{(i0)}$.
end

Common choices for sapling distribution $p(\hat{\theta}^{(0)})$:

- uniform distribution
- latin hypercube (with M points)

Global optimisation using multi-start local optimisation

Analysis of fitting results using sorted optimiser runs:



Remark: Same optimum iff same parameter vector.

Properties of multi-start local optimisation: Large number of samples requires if many local optima or if the global optimum has a small basin of attraction.

Global optimisation using alternative methods

Alternative global optimisation algorithms:

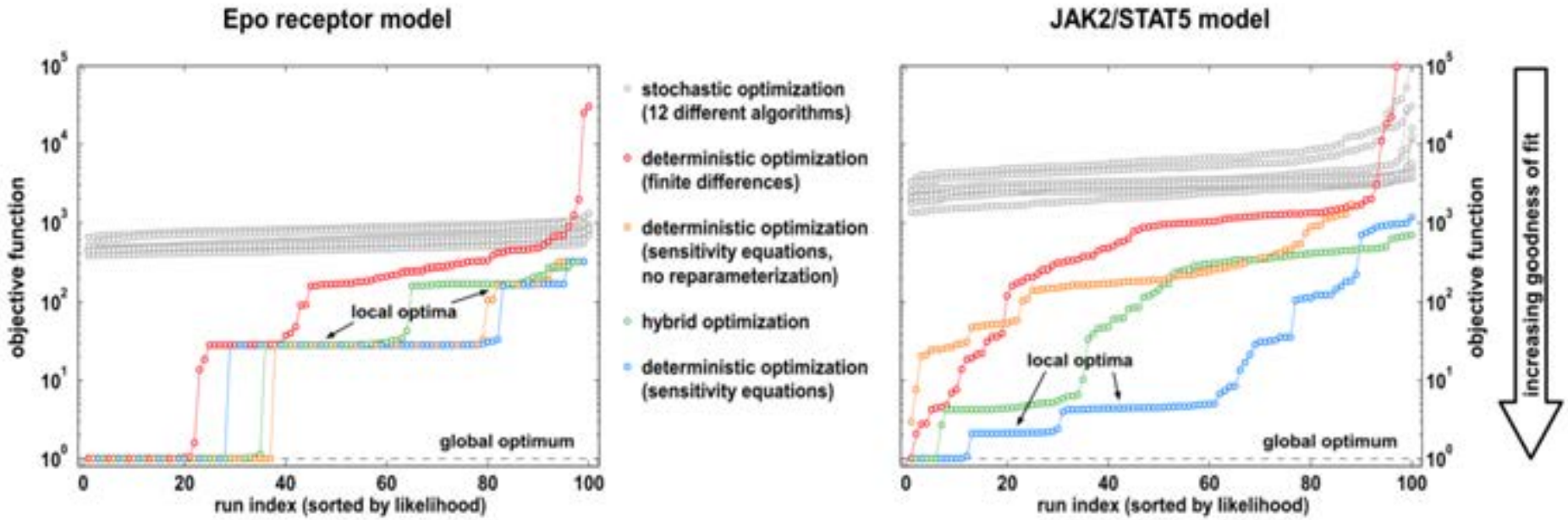
- Deterministic methods
  - branch-and-bound
  - interval optimisation
- Stochastic, thermodynamic methods
  - simulated annealing
  - evolutionary algorithms
  - swarm-based optimisation algorithms
- Hybrid stochastic-deterministic methods

Common claim:

Many global optimisation procedure better than multi-start optimisation.

Personal finding: Multi-start local optimisation is for medium- and large-scale parameter estimation problems in systems / computational biology much better than other methods, assuming that the local optimiser works well.

Comparison of global optimisation methods



Raue et al., PLoS ONE, 2013

Observation 1: Only multi-start local optimisation achieves convergence for high-dimensional problems.

Observation 2: Reparameterisation (i.e., log-transformation) is not only important for the efficiency but also the convergence of optimisers.

Software packages

Data2Dynamics (D2D)

- Simple handling of model, experimental data and experimental conditions
- Highly automised
- Multi-experiment fitting for ODEs
- Parameter optimisation and uncertainty analysis

Advanced MATLAB Interface for CVODE and IDAS (AMICI)

- 1st and 2nd order forward sensitivity analysis
- 1st (and soon 2nd) order adjoints sensitivity analysis
- Support for state and output events

Parameter EStimation TOolbox (PESTO)

- Parameter optimisation
- Uncertainty analysis
- Flexible interface

MEtaheuristics for systems biology and bioinformatics Global Opt. (MEIGO)

- Flexible toolbox for global optimisation
- Support of integer variables

Outline

Assessment of residual distribution

Independent, additive normally distributed measurement noise:

$$\bar{y}_{jk} = \bar{y}_j(t_k, \theta^{\text{true}}) + \epsilon_{jk} \quad \text{with} \quad \epsilon_{jk} \sim \mathcal{N}(0, \sigma_{jk}^2)$$

$$\Rightarrow \text{residual } r_{jk} = \frac{\bar{y}_{jk} - \bar{y}_j(t_k, \theta^{\text{true}})}{\sigma_{jk}} \sim \mathcal{N}(0, 1)$$

Evaluation of residual distribution

Visualisation of residual distribution for best fit and comparison with $\mathcal{N}(0, 1)$.

Properties:

- Qualitative insights:
  - measured error variances $\rightarrow$ over-/underfitting & distribution assumption
  - estimated error variances $\rightarrow$ distribution assumption
- Extensions towards (temporal) correlation straight forward

Assessment of objective function value (1)

Case: (Unbiased) estimated of the error variances from experimental data

Negative log-likelihood function

$$J(\theta) = \underbrace{\frac{1}{2}\sum_{k=1}^{n_t}\sum_{j=1}^{n_y}\log\left(2\pi\sigma_{jk}^2\right)}_{=C(\text{known constant})} + \underbrace{\frac{1}{2}\sum_{k=1}^{n_t}\sum_{j=1}^{n_y}\left(\frac{\bar{y}_{jk} - y_j(t_k,\theta)}{\sigma_{jk}}\right)^2}_{=\text{weighted quadratic distance}}$$

Objective function distribution for true parameter, $\theta = \theta^{\text{true}}$

$$2J(\theta^{\text{true}}) - C = \sum_{k=1}^{n_t}\sum_{j=1}^{n_y} r_{jk}^2 \sim \chi^2(\cdot|n_t \cdot n_y)$$

Remark: There exists θ^{ml} with $J(\theta^{ml}) < J(\theta^{\text{true}})$, implying that θ^{true} is suboptimal.

Approximate objective function distribution for optimal ML estimate, $\theta^{ml} = \theta^{\text{true}}$

$$2J(\theta^{ml}) - C \sim \chi^2(n_t \cdot n_y - n_\theta).$$

Assessment of objective function value (2)

Case: measured error variances

Evaluation of objective function value

Comparison of objective function value achieved for experimental data with theoretical distribution.

$$\int_0^{2J(\theta^{ml})-C} \chi^2(j|n_t \cdot n_y - n_\theta)dj \begin{cases} \in (0.05, 0.95) & \Rightarrow \text{reasonable fit} \\ > 0.95 & \Rightarrow \text{underfitting (model too simple)} \\ < 0.05 & \Rightarrow \text{overfitting (model too complex)} \end{cases}$$

Remark 1: Improved approximation and extension for the case of estimated error variances via parametric and non-parameteric bootstrapping.

Remark 2: Underfitting can also be the result of the use of an inappropriate optimiser.

Outline

The “vehicle problem” car freaks face

Task: Given only a few information, find the right vehicle.

Vehicle 1
Lamborghini
Aventador lp700-4

Vehicle 2
the tractor of my uncle

Vehicle 3
mini cooper

Dataset 1
0 - 100 km/h: 2.7 sec

Dataset 2
4.2 l / 100 km

Dataset 3
4 wheels

Problem: Data are often not sufficient to solve the inverse problem.
⇒ identifiability problems and uncertainties

Resolving a crime

Task: Find the murderer.

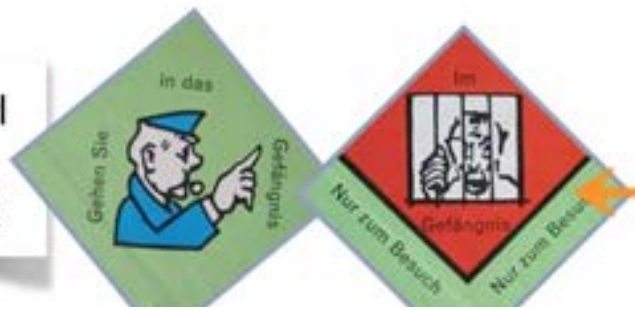
Data:
- dead body
- fingerprints
- foodprints
- DNA
- ...

Prior information:
- regular suspects
- ...
-

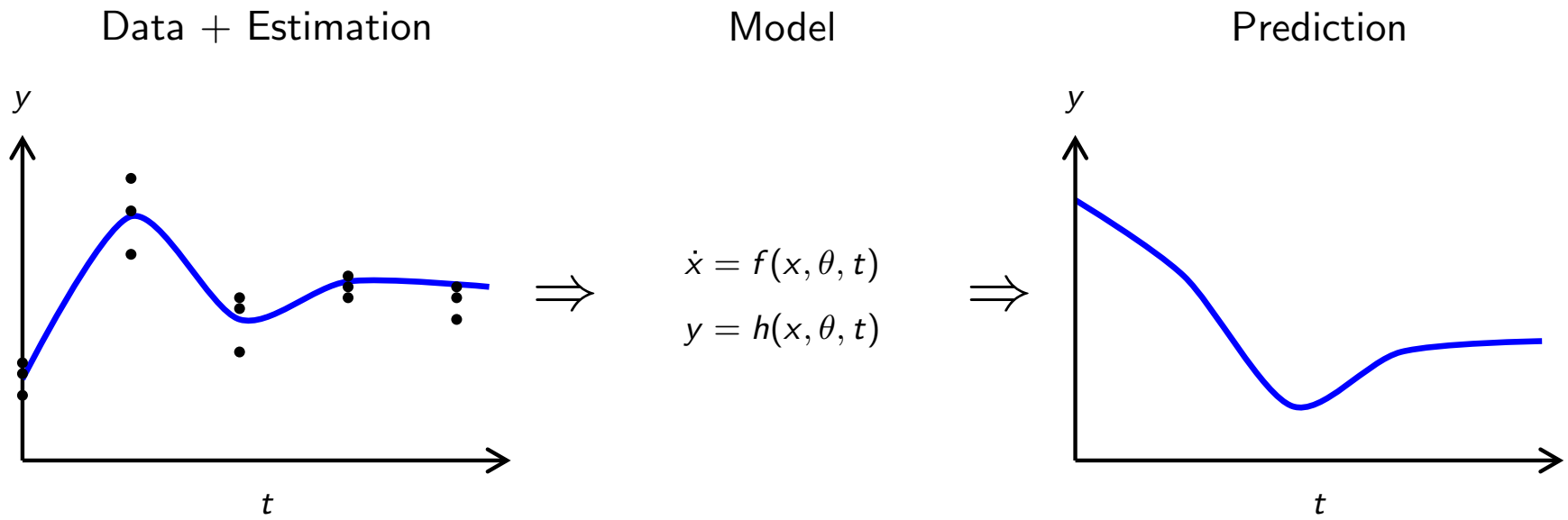
Estimator
(clever detectives)

Uncertainty / Identifiability analysis
(judge)

Problem of general importance
(8 million viewers)!



Model and prediction uncertainties



Optimisation-based approach

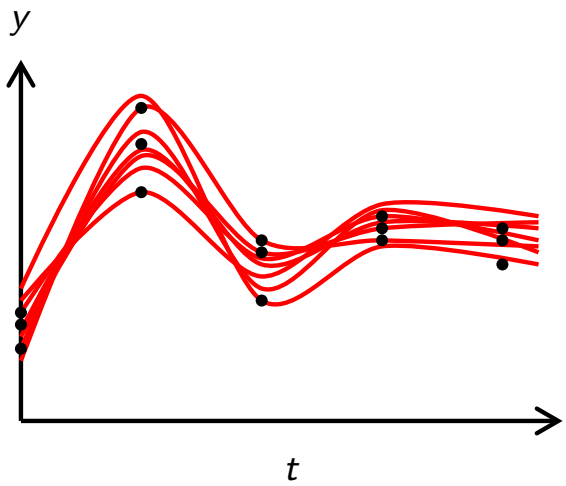
1. Calculation of *optimal* parameter
   (e.g. least-square, or maximum-likelihood)

$\Rightarrow$ *optimal* model of the process

2. Model falsification/rejection and prediction using *optimal* model

Model and prediction uncertainties

Data + Estimation



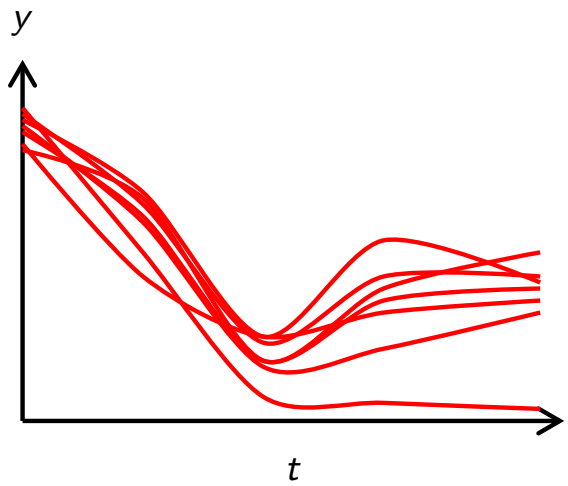
$\Rightarrow$

Model

$$\dot{x} = f(x, \theta^i, t)$$
$$y = h(x, \theta^i, t)$$
$$i \in \{1, \ldots, N\}$$

$\Rightarrow$

Prediction



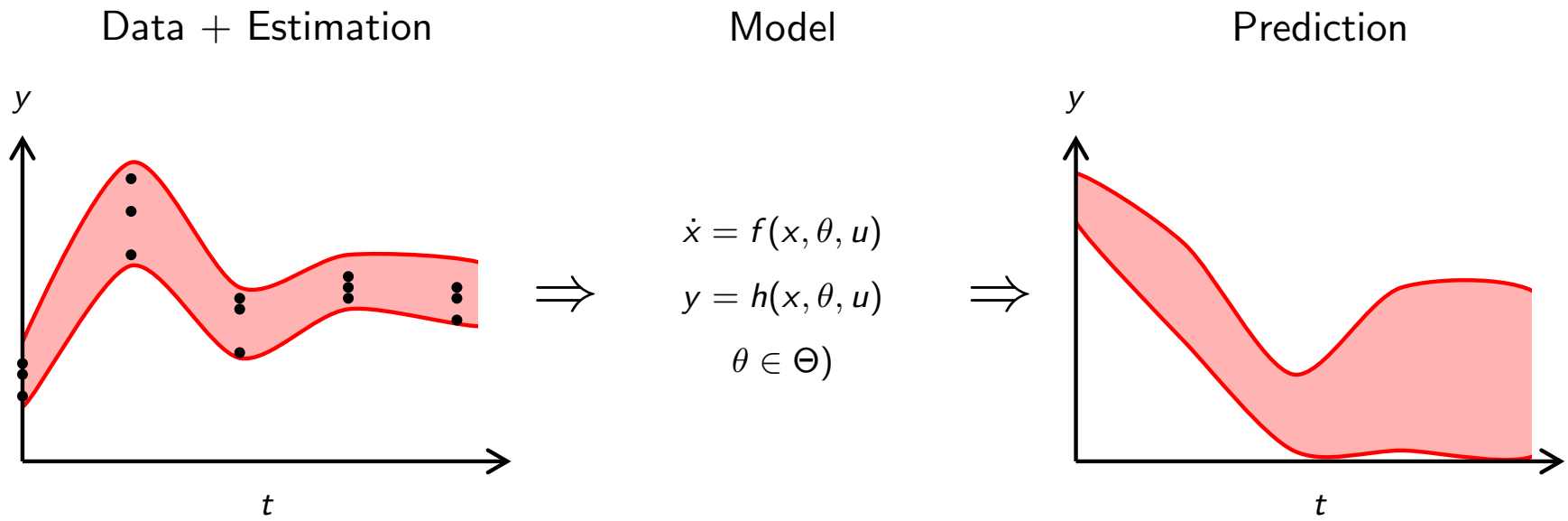
Sampling-based approach

1. Calculation of a sample of *good* parameters
   (e.g. Bayesian methods, or Bootstrapping)

$\Rightarrow$ sample of *good* models of the process

2. Model falsification/rejection and prediction using a sample of models

Model and prediction uncertainties



Set-based approach

1. Calculation of the set of *consistent* parameters
   (e.g. interval arithmetics, SOS, or convex optimisation)

$\Rightarrow$ all *consistent* models of the process (with a certain structure)

2. Model falsification/rejection and prediction using the set of models

Illustration of confidence interval and confidence region

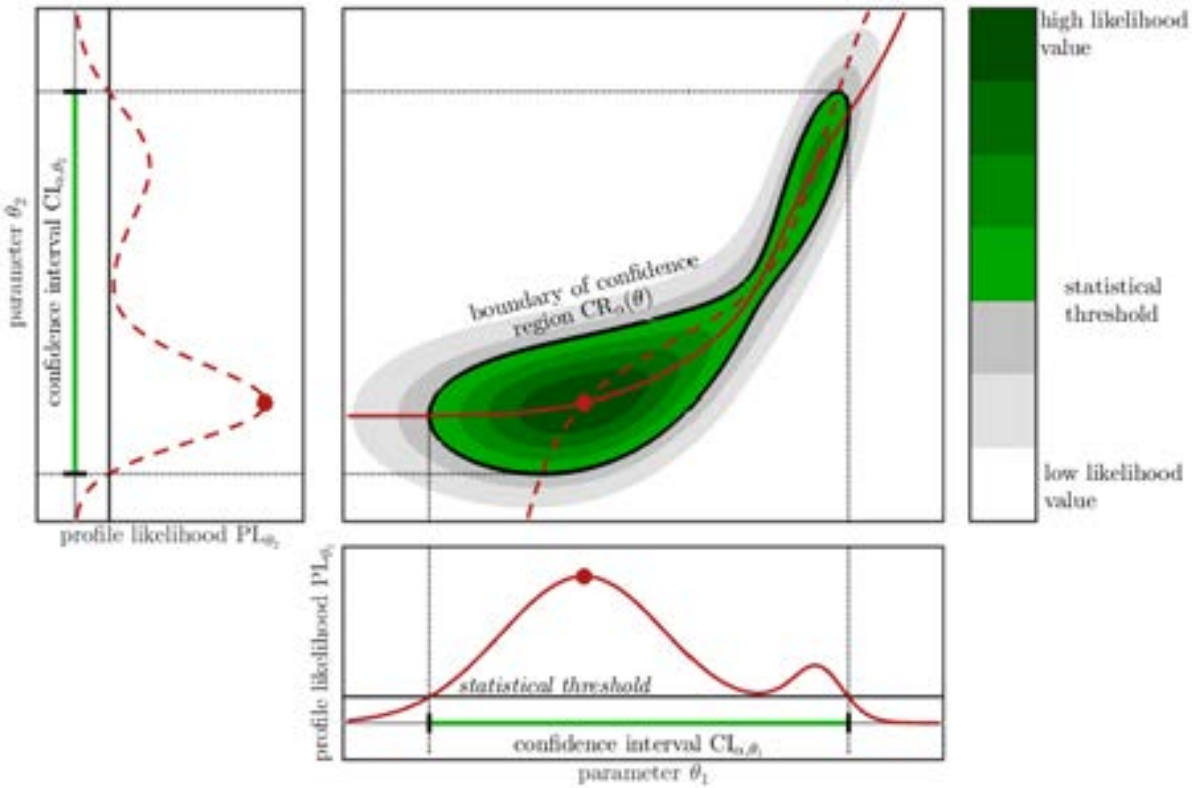


Illustration of confidence region, confidence intervals, profile likelihoods and their relation. (big panel) Likelihood function landscape (shading), confidence region (■) and profile likelihood path θ_c (θ_1:━;θ_2:▪▪). (small panels) Profile likelihood ratio (━) and confidence interval (━) for θ_1 and θ_2. The relation of different quantities is indicated using dotted lines. The significance threshold is indicated in all three figures as solid black line.

Confidence region

Goal: Assessment of parameters and prediction uncertainties for a given set of experimental data.

Confidence region

For the parameter vector $\theta \in \Theta$ we define the confidence region to the confidence level α as

$$\begin{aligned} \mathrm{CR}_\alpha &= \left\{ \theta \in \Theta \,\middle|\, \frac{\mathcal{L}_\mathcal{D}(\theta)}{\mathcal{L}_\mathcal{D}(\hat{\theta})} \geq \exp\left(-\frac{\Delta_\alpha}{2}\right) \right\}, \\ &= \left\{ \theta \in \Theta \,\middle|\, 2\left(J(\theta) - J(\hat{\theta})\right) \leq \Delta_\alpha \right\}, \end{aligned}$$

with Δ_α denoting the αth-percentile of the χ^2 distribution with one degree of freedom.

Property: the likelihood-based confidence region should contain the true parameter in $(1-\alpha)$ when calculated for a large number or experimental replicates. This estimate is however for nonlinear systems often merely a rough approximation.

Confidence interval

Model property $g(\theta)$, e.g.

- individual parameter: $g(\theta) = \theta_j$
- state x_j a time point T: $g(\theta) = x_j(T, \theta)$

Confidence interval

The confidence interval for a model property $g(\theta)$ is the projection of CR_α onto $g(\theta)$,

$$\mathrm{CI}_{\alpha, g(\theta)} = P_{g(\theta)}\mathrm{CR}_\alpha = \{c \,|\exists \theta \in \mathrm{CR}_\alpha \wedge g(\theta) = c\}\,.$$

Properties:

- Confidence intervals provide a measure for the uncertainty of $g(\theta)$.
- Confidence intervals to a confidence level α can be bounded or unbounded.

Practical identifiability

Practical identifiability

A model property $g(\theta)$ is called practically identifiable if its confidence interval $\mathrm{CI}_{\alpha,g(\theta)}$ is bounded; otherwise it is called practically non-identifiable.

Remark 1: The practical identifiability depends on the confidence level α.

Remark 2: Structural identifiability $\not\Rightarrow$ practical identifiability

Profile likelihood

Profile likelihood

For the model property $g(\theta)$ we define the profile likelihood as

$$\mathrm{PL}_{g(\theta)}(c) = \max_{\theta \in \Theta} \mathcal{L}_{\mathcal{D}}(\theta) \text{ subject to } g(\theta) = c.$$

For values c outside the range of $g(\theta)$, $\mathrm{PL}_{g(\theta)}(c) = 0$.

From the profile likelihood the confidence interval for $g(\theta)$ follows as

$$\mathrm{CI}_{\alpha, g(\theta)} = \left\{ c \,\middle|\, \frac{\mathrm{PL}_{g(\theta)}(c)}{\mathcal{L}_{\mathcal{D}}(\hat{\theta})} \geq \exp\left(-\frac{\Delta_\alpha}{2}\right) \right\}.$$

Remark:

- Profile likelihoods facilitate the calculation of confidence intervals without the evaluation of the confidence region or its projection.
- Profile likelihood based confidence intervals are also called "finite sample confidence intervals".

Illustration of profile likelihood calculation

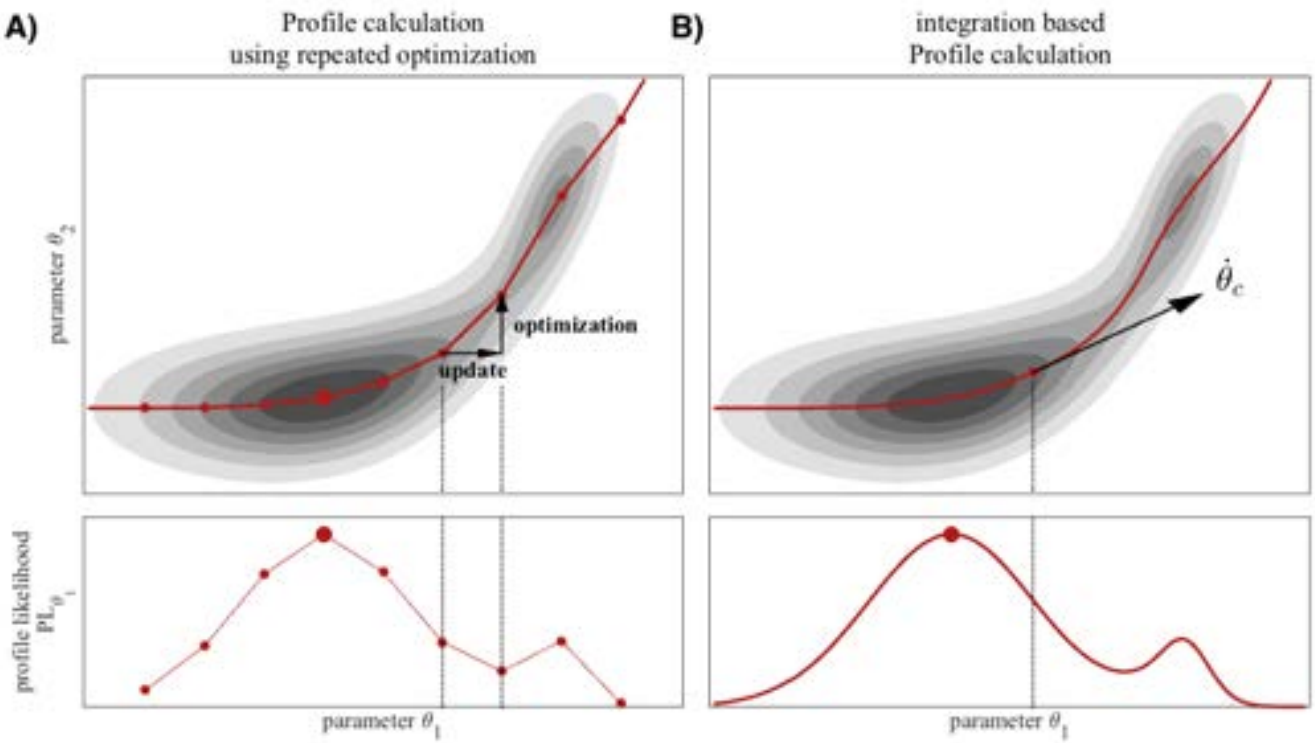


Illustration of optimization based profile likelihood calculation (upper panels) and integration based profile likelihood calculation (lower panels). (upper panel in A) Optimization based profile likelihood calculation for likelihood function (shading) using update and re-optimization step (arrows). (upper panel in B) Evaluation points (•) and approximation of the profile likelihood (▬). (lower panel in A) Integration based profile likelihood calculation for likelihood function (shading) using continuous system with derivative (arrow) tangential to the parameter trajectory (▬). (lower panel in B) Profile likelihood (▬) obtained using integration based method.

Method for optimisation-based profile likelihood calculation

Profile likelihood

Sequence of constraint optimisation problems,

$$\min_{\theta \in \Theta} J(\theta) \text{ subject to } g(\theta) = c,$$

for values c which are either on a grid or chosen adaptively.

Implementation as sequence of local optimisation problems with starting point

1. **0th order proposal:** the optimal point for c_{l-1}, $\theta^{(0)}_{c_l} = \theta_{c_{l-1}}$, or
2. **1st order proposal:** the linear extrapolation based on the optimal points for c_{l-1} and c_{l-2},

$$\theta^{(0)}_{c_l} = \theta_{c_{l-1}} + \frac{c_l - c_{l-1}}{c_{l-1} - c_{l-2}}(\theta_{c_{l-1}} - \theta_{c_{l-2}}).$$

Properties:

- Large number of local optimisations.
- (Relatively) efficient and robust implementation. (see D2D and PESTO)
- Potentially initialisation at multiple local optima required which are above the statistical threshold.

Implementation of optimisation-based profile likelihood calculation

Input: parameter interval of interests $[g_{\min}, g_{\max}]$, step size Δc, maximum likelihood estimate $\hat{\theta}$

set: $c_0 = g(\hat{\theta})$
set: $\theta(c_0) = \hat{\theta}$
set: $\mathrm{PL}_{g(\theta)}(c) = \mathcal{L}_{\mathcal{D}}(\hat{\theta})$

% Profile for increasing values of c
set: $c = c_0$ **repeat**
  update: $c = c + \Delta c$
  optimise: $\max_{\theta_{j \neq i}} \mathcal{L}_{\mathcal{D}}(\theta)$ for fixed $g(\theta) = c \Rightarrow \theta(c)$ and $\mathrm{PL}_{g(\theta)}(c)$
until $c + \Delta c \notin [g_{\min}, g_{\max}]$ *or* $\mathrm{PL}_{g(\theta)}(c)$ *below threshold*;

% Profile for decreasing values of c
set: $c = c_0$ **repeat**
  update: $c = c - \Delta c$
  optimise: $\max_{\theta_{j \neq i}} \mathcal{L}_{\mathcal{D}}(\theta)$ for fixed $g(\theta) = c \Rightarrow \theta(c)$ and $\mathrm{PL}_{g(\theta)}(c)$
until $c - \Delta c \notin [g_{\min}, g_{\max}]$ *or* $\mathrm{PL}_{g(\theta)}(c)$ *below threshold*;

Illustration of parameter sampling

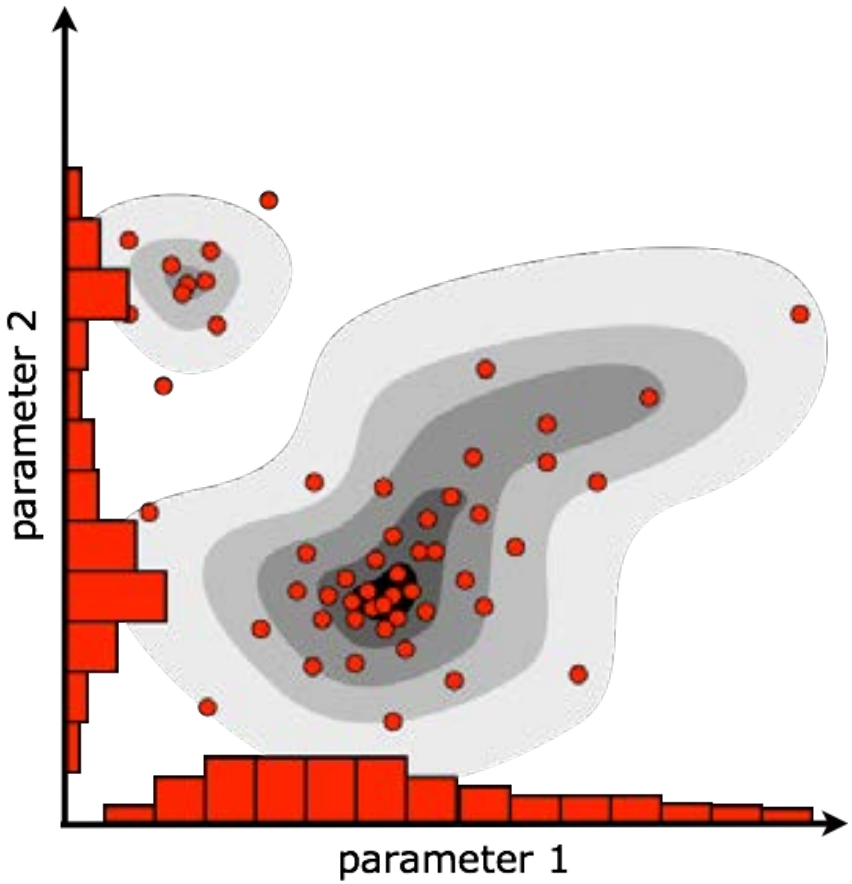
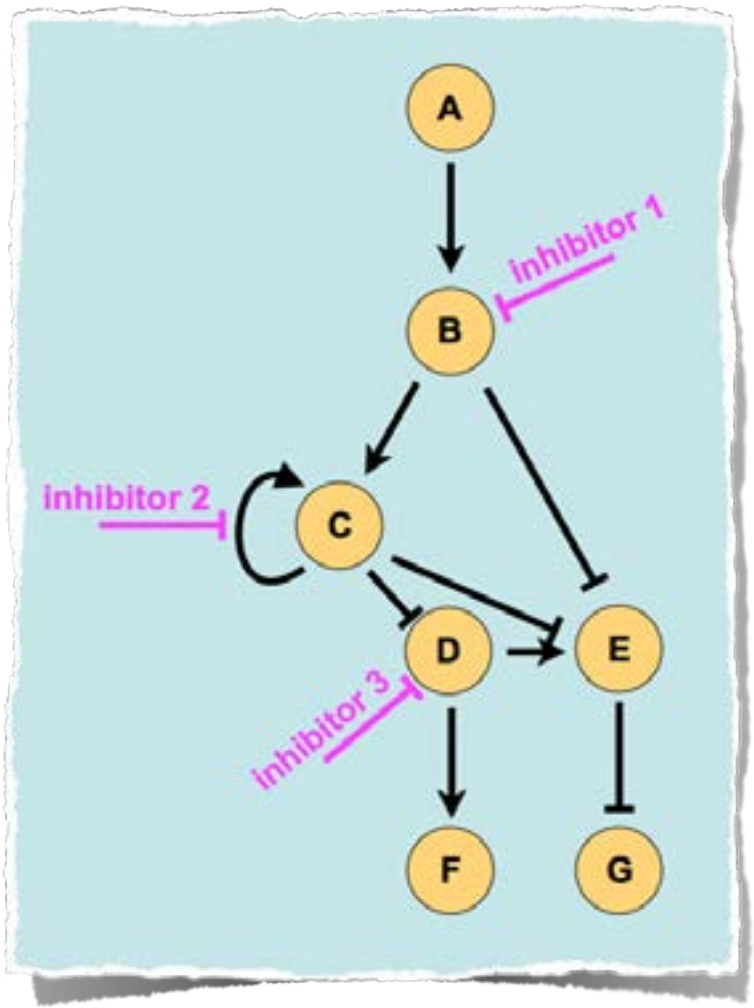


Illustration of sampling results. Likelihood function landscape (shading), samples (●) and histogram of samples (■),

Outline

Selection of drug targets



Starting point:

Definition of the aim of the pharmaceutical intervention, e.g. inhibition / activation of a molecular species influencing the phenotype.

Question:

How to best inhibit/activate a certain component, e.g. inhibit F and activate G?

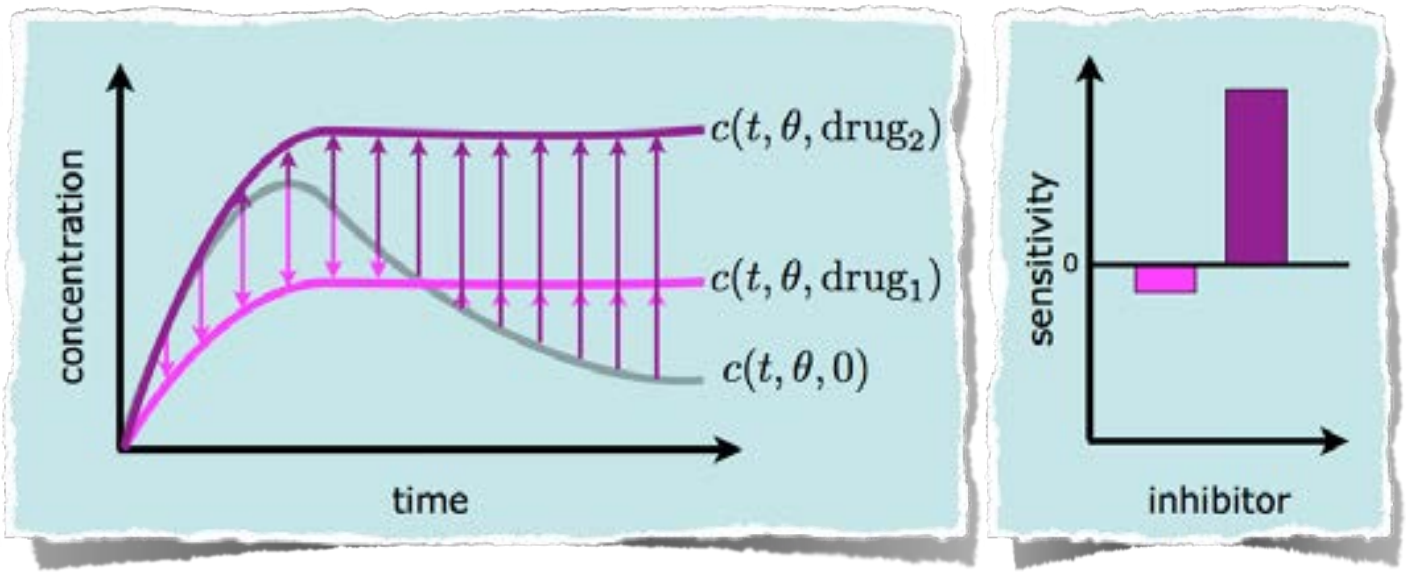
The answer depends on the quantitative properties of the network.

Analysis of the mathematical model to identify drug candidates

Question: Which drug has the largest impact on the property of interest, e.g., the activation of a certain transcription faction?

Approach

Implementation of the different drug candidates in the model and study of the (integrated) local / global sensitivity analysis with respect to the drug concentration.



Local sensitivity analysis

Model with drug d:

$$\frac{dx(t,d)}{dt} = f(x(t),\theta,t,d), \qquad x(0) = x_0(\theta) \quad \text{“state variables”}$$

$$c(t,d) = \int_0^T g(x(t),\theta,t,d))dt \qquad \text{“interesting network property”}$$

Sensitivity of state and network properties

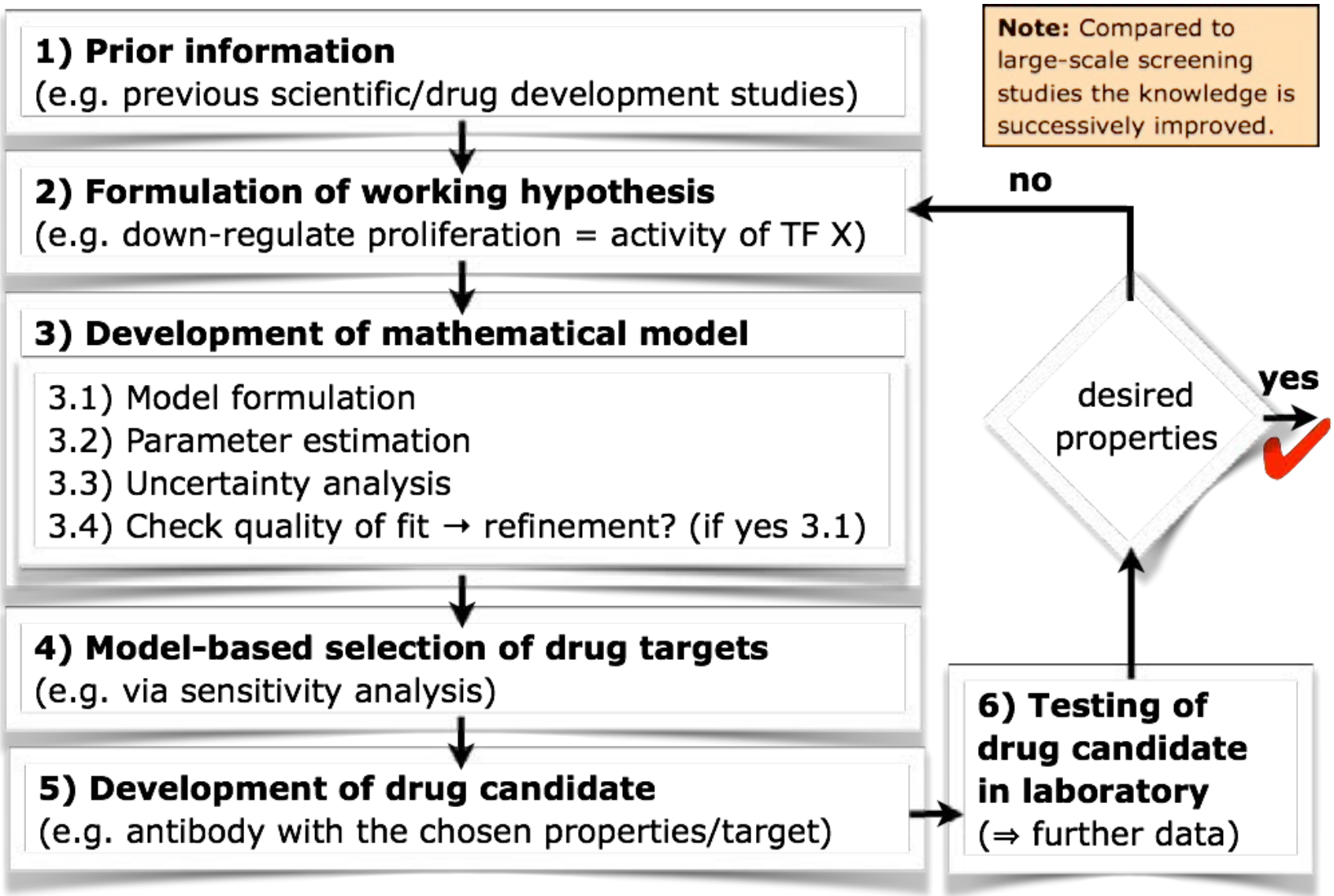
$$s^x(t,d) = \left(\frac{\partial x_1(t,\theta)}{\partial d},\ldots,\frac{\partial x_{n_x}(t,\theta)}{\partial d}\right)^T \quad \text{and} \quad s^c(t,d) = \left(\frac{\partial c_1(t,\theta)}{\partial d},\ldots,\frac{\partial c_{n_c}(t,\theta)}{\partial d}\right)^T$$

Evaluation of selection criterion

$$\dot{s}^x = \frac{\partial f}{\partial x}s^x + \frac{\partial f}{\partial d}, \qquad s^x(0) = \frac{\partial x_0}{\partial d}$$

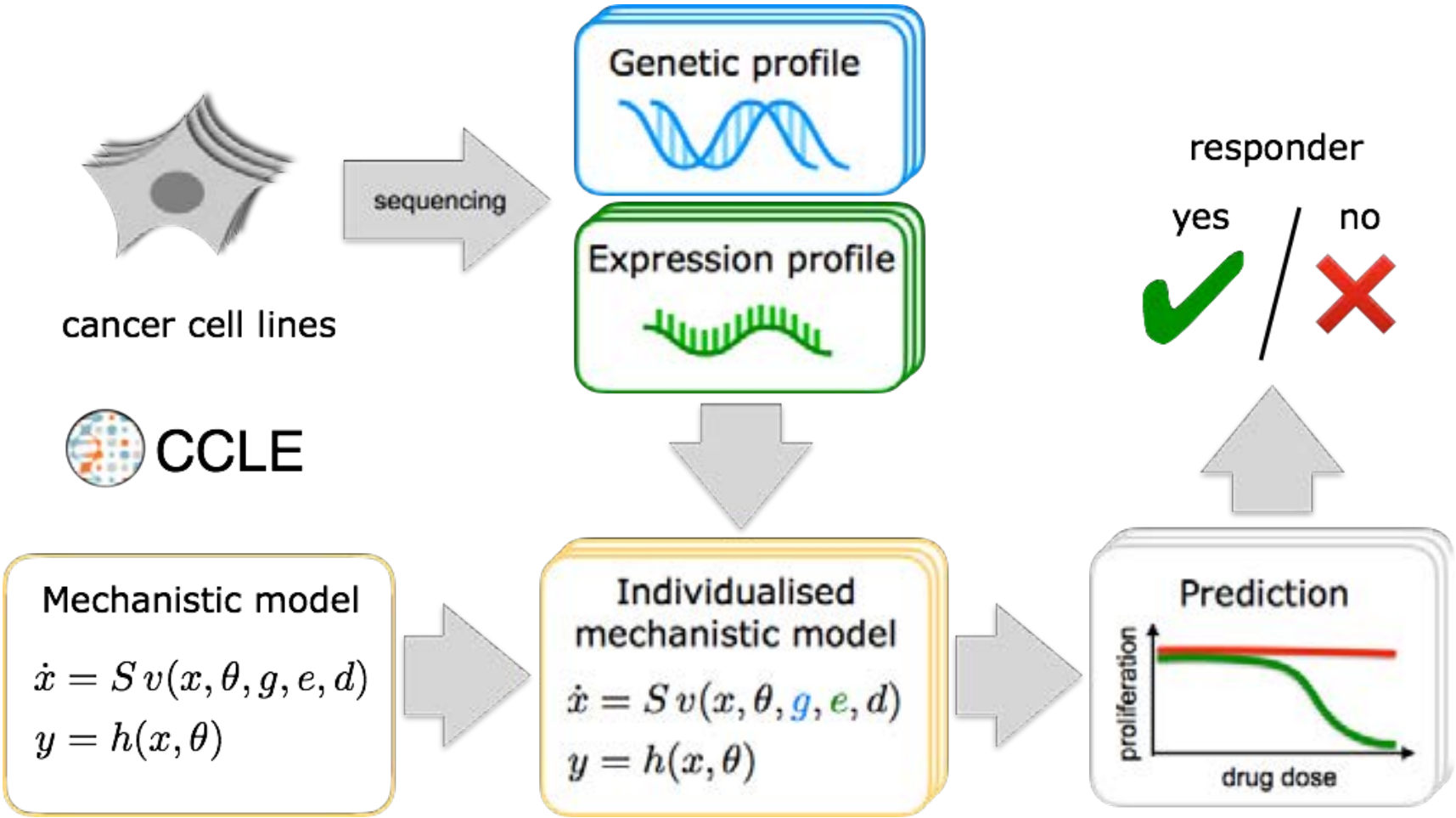
$$s^c = \int_0^T \frac{\partial g}{\partial x}s^x + \frac{\partial g}{\partial d}dt \qquad \leftarrow \text{“criterion for drug selection”}$$

Model-based drug development loop

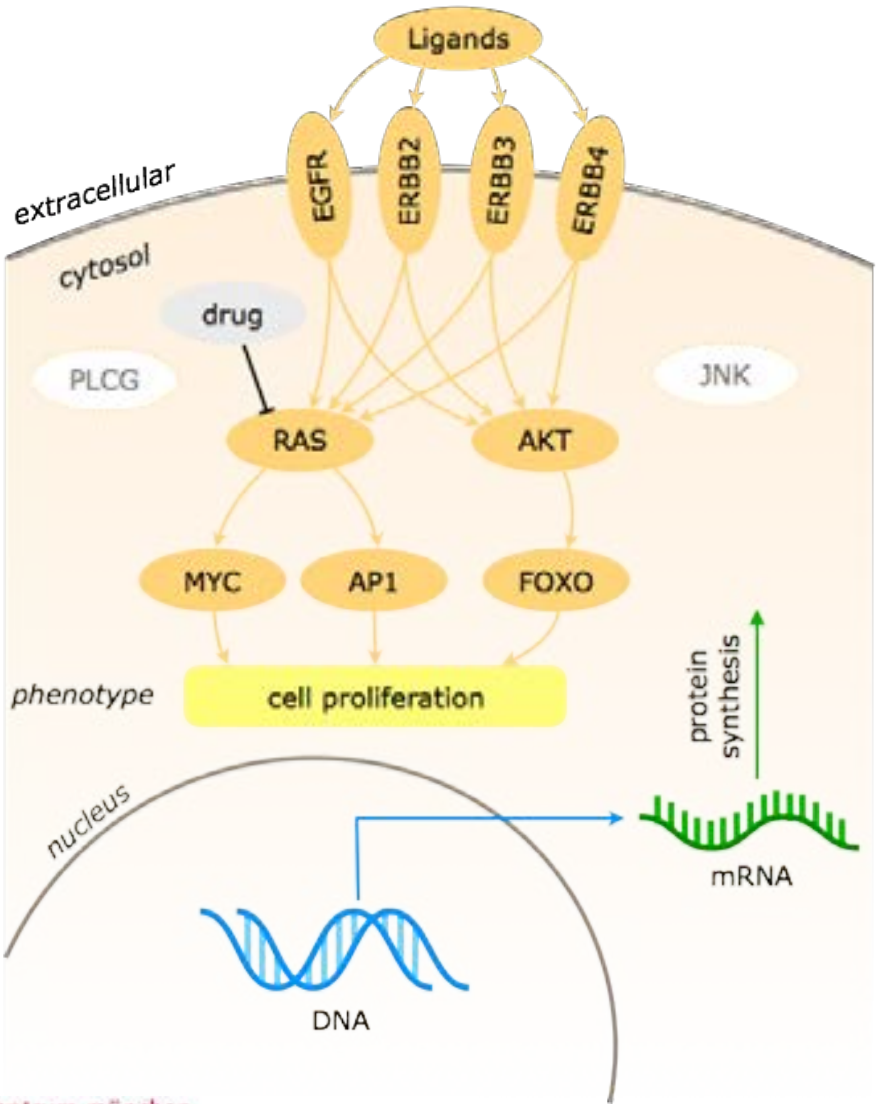


Outline

Paradigm of model-based prediction of personalised drug response



Structure of the individualisable model



Protein-protein interaction network

Mechanistic description:

$$\text{KRAS:GDP} \rightleftarrows \text{KRAS:GTP}$$

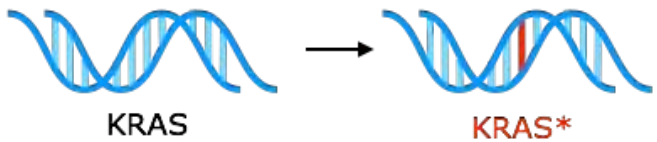
$$\text{RAF} + \text{KRAS:GTP} \rightarrow \text{pRAF} + \text{KRAS:GTP}$$

$$\text{KRAS*:GDP} \rightleftarrows \text{KRAS*:GTP}$$

$$\text{RAF} + \text{KRAS*:GTP} \rightarrow \text{pRAF} + \text{KRAS*:GTP}$$

Genetic profile

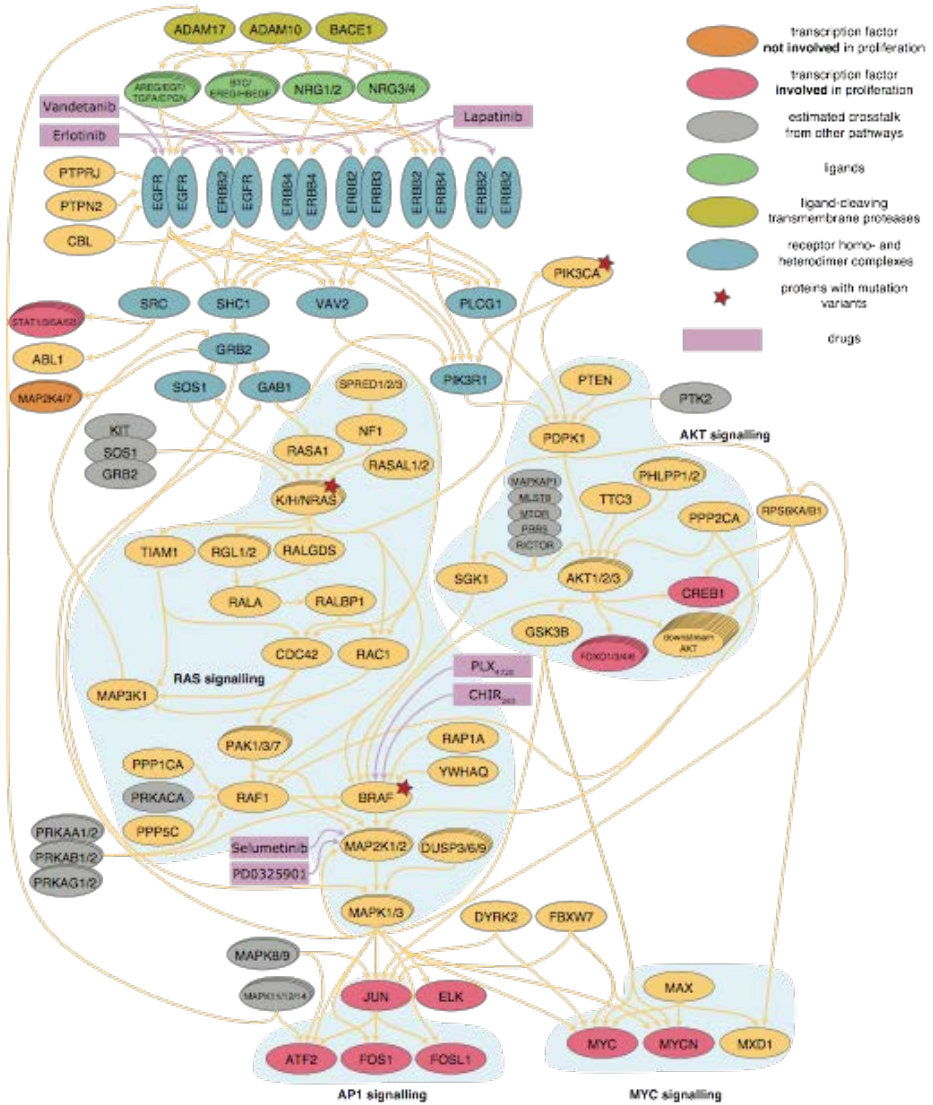
Mutation patterns:



Gene expression profile

Mechanistic description of protein synthesis:

Mechanistic model of some cancer signalling pathway



Model properties

Genes: 112
Mutant genes: 24

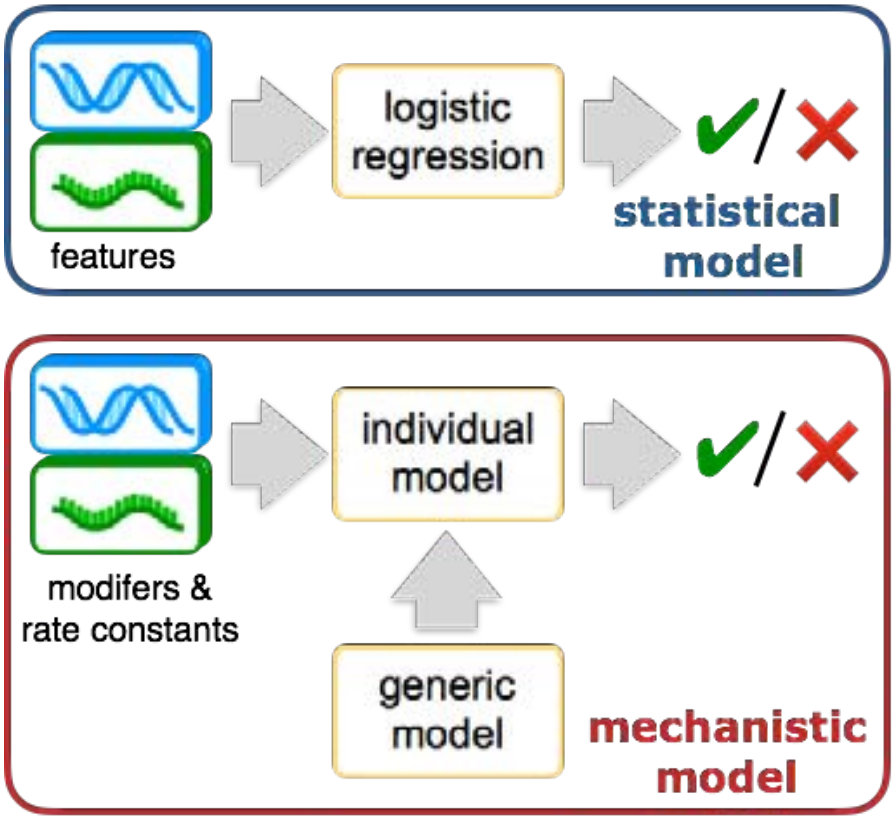
Reactions: 2704

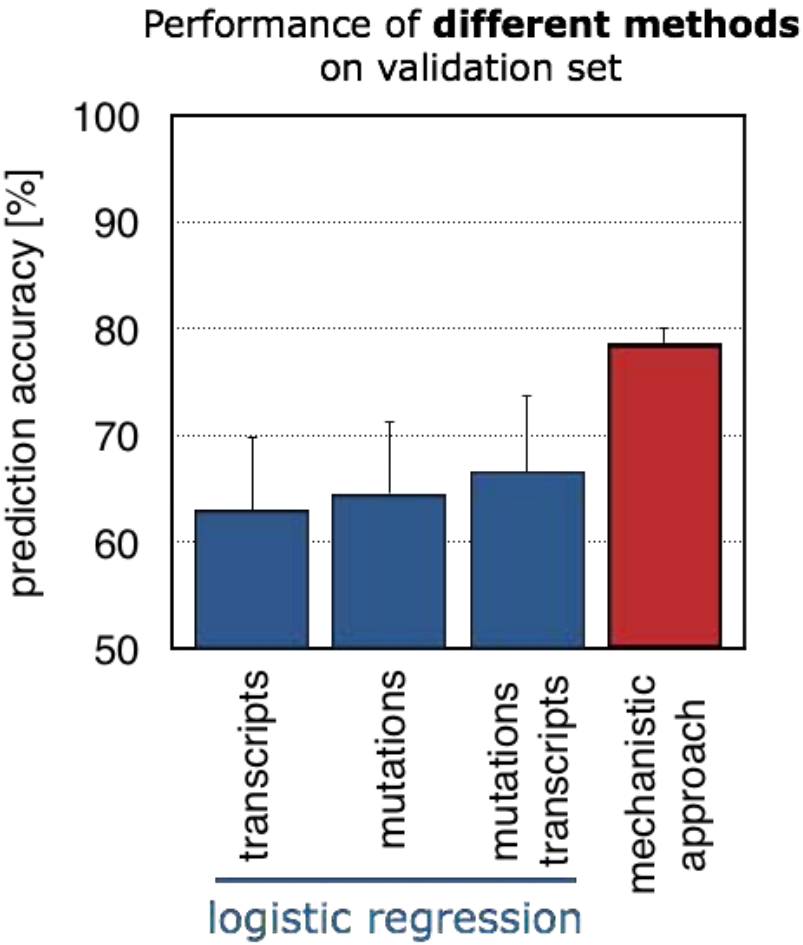
Drugs: 7

State variables: 1230
Parameters: 4256

TREND: Development of comprehensive disease maps (see http://disease-maps.org/).

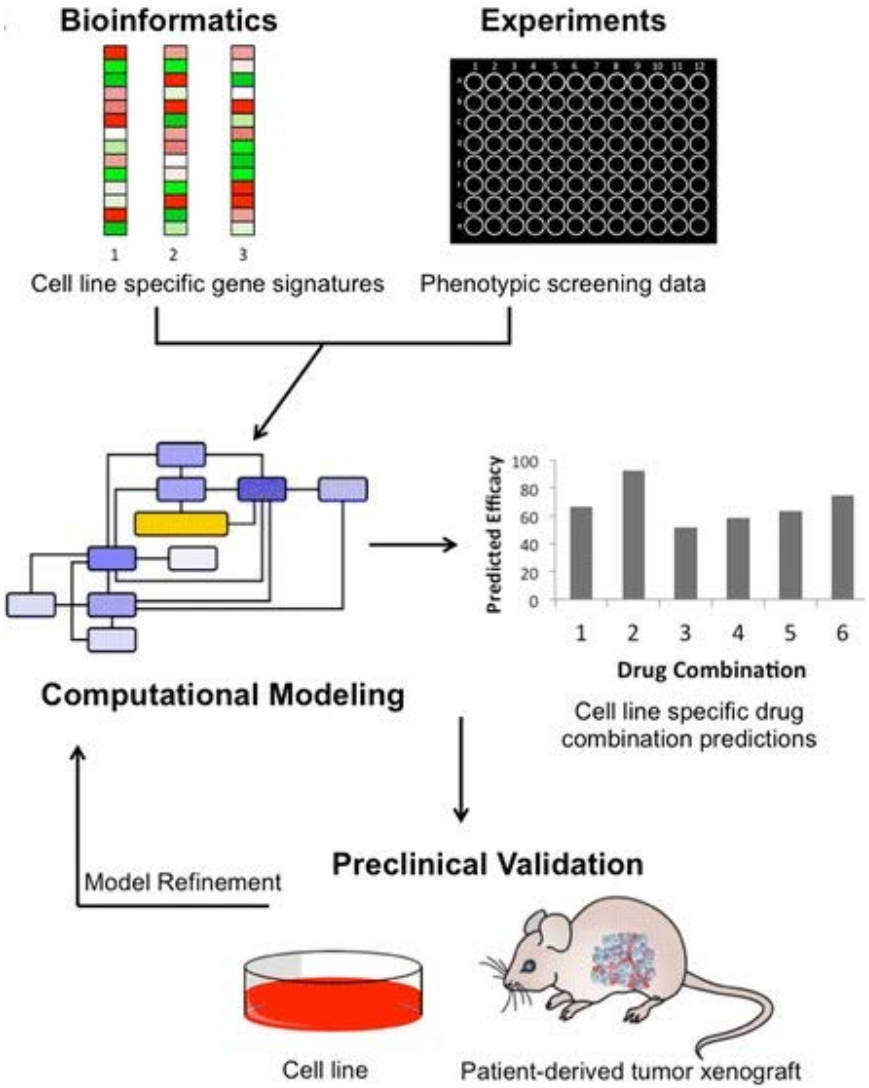
Comparison to mechanistic and statistical approaches





Outline

Where is the field going?



K. A Ryall & A. Choon Tan, Journal of Cheminformatics, 2015.

Take home messages

Why models?

- Consistent formulation
- Integration of knowledge
- Testable prediction
- Successive development
- AND: Available software tools make them easy to use!

How to develop / use them?

1. Modeling of biochemical reaction network
2. Estimation of unknown parameters from experimental data
3. Assessment of parameter and prediction uncertainties
4. Analysis of model with respect to drug targets / combination therapies
5. Experimental assessment predictions
6. Return (1)

Questions?

References

- **Overview / Review papers**

  A. Raue, M. Schilling, J. Bachmann, A. Matteson, M. Schelke, D. Kaschek, S. Hug, C. Kreutz, B. D. Harms, F. J. Theis, U. Klingmüller, and J. Timmer. Lessons learned from quantitative dynamical modeling in systems biology. *PLoS ONE*, 8(9):e74335, September 2015.

  J. Hasenauer, N. Jagiella, S. Hross, and F. J. Theis. Data-driven modelling of biological multi-scale processes. *Journal of Coupled Systems and Multiscale Dynamics*, 3(2):101–121, June 2015.

- **Software tools**

  A. Raue, B. Steiert, M. Schelker, C. Kreutz, T. Maiwald, H. Hass, J. Vanlier, C. Tönsing, L. Adlung, R. Engesser, W. Mader, T. Heinemann, J. Hasenauer, M. Schilling, T. Höfer, E. Klipp, F. J. Theis, U. Klingmüller, B. Schöberl, and J. Timmer. Data2Dynamics: a modeling environment tailored to parameter estimation in dynamical systems. *Bioinformatics*, 31(21):3558–3560, November 2015.

  S. Hoops, S. Sahle, R. Gauges, C. Lee, J. Pahle, N. Simus, M. Singhal, L. Xu, P. Mendes, and U. Kummer. COPASI – a COmplex PAthway SImulator. *Bioinformatics*, 22(24):3067–3074, October 2006.

  E. Balsa-Canto, and J. R. Banga. AMIGO, a toolbox for advanced model identification in systems biology using global optimization. *Bioinformatics*, 27(16):2311–2313, August 2011.

- **Optimisation methods**

  F. Fröhlich, B. Kaltenbacher, F. J. Theis, and J. Hasenauer. Scalable parameter estimation for genome-scale biochemical reaction networks. *PLoS Computational Biology*, 13(1):e1005331, January 2017.

  J. A. Egea, D. Henriques, T. Cokelaer, A. F. Villaverde, A. MacNamara, D. P. Danciu, J. R. Banga, and J. Saez-Rodriguez. MEIGO: an open-source software suite based on metaheuristics for global optimization in systems biology and bioinformatics. *BMC Bioinformatics*, 15:136, January 2014.

- **Identifiability and uncertainty analysis**

  S. Hug, A. Raue, J. Hasenauer, J. Bachmann, U. Klingmüller, J. Timmer, and F. J. Theis. High-dimensional Bayesian parameter estimation: Case study for a model of JAK2/STAT5 signaling. *Mathematical Biosciences*, 246(2):293–304, April 2013.

  A. Raue, C. Kreutz, T. Maiwald, J. Bachmann, M. Schilling, U. Klingmüller, and J. Timmer. Structural and practical identifiability analysis of partially observed dynamical models by exploiting the profile likelihood. *Bioinformatics*, 25(25):1923–1929, May 2009.

References

- **Outliers**

  C. Maier, C. Loos, and J. Hasenauer. Robust parameter estimation for dynamical systems from outlier-corrupted data. *Bioinformatics*, 2017, first online.

- **Unknown inputs**

  M. Schelker, A. Raue, J. Timmer, and C. Kreutz. Comprehensive estimation of input signals and dynamics in biochemical reaction networks. *Bioinformatics*, 28(18):i529—i534, September 2012.

- **Discrete events**

  F. Fröhlich, F. J. Theis, J. Rädler, and J. Hasenauer. Parameter estimation for dynamical systems with discrete events and logical operations. *Bioinformatics,* 2017, first online.

- **Steady state constraints**

  A. Fiedler, S. Raeth, F. J. Theis, A. Hausser, and J. Hasenauer. Tailored parameter optimization methods for ordinary differential equation models with steady-state constraints. *BMC Systems Biology*, 10:80, July 2016.

  M. Rosenblatt, J. Timmer and D. Kaschek. Tailored steady-state constraints for parameter estimation in non-linear ordinary differential equation models. *Frontiers in Cell and Developmental Biology*, 4(41), May 2016.

- **Stochastic models**

  F. Fröhlich, P. Thomas, A. Kazeroonian, F. J. Theis, R. Grima, and J. Hasenauer. Inference for stochastic chemical kinetics using moment equations and system size expansion. *PLoS Computational Biology*, 12(7):e1005030, July 2016.

  G. Lillacci, and M. Khammash. The signal within the noise: efficient inference of stochastic gene regulation models using fluorescence histograms and stochastic simulations. *Bioinformatics*, 29(18):2311–2318, July 2013.

  T. Toni, D. Welch, N. Strelkowa, A. Ipsen, and M. P. H. Stumpf. Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems. *Journal of the Royal Society Interface*, 6:187–202, July 2009.

  M. Komorowski, M. J. Costa, D. A. Rand, and M. P. H. Stumpf. Sensitivity, robustness, and identifiability in stochastic chemical kinetics models. *PNAS*, 108(21):8645–8650, May 2011.