# Constrained Evolutionary Search for Model Parameters.
# Case Studies in Thymocyte Dynamics

Daniela Zaharie

Department of Computer Science,
West University of Timişoara, Romania
dzaharie@info.uvt.ro

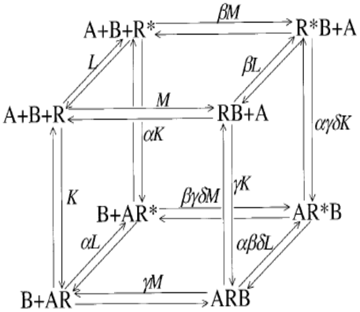
ECODAM - Iasi, 25 June 2013

# Outline

1. Some problems requiring parameter estimation
2. Constrained evolutionary search of the parameter space
3. Case study: modelling a perturbed thymocyte dynamics
4. Exploring the output of the evolutionary search

# A parameterized model in pharmacology

## Allosteric two-state model of receptor activation



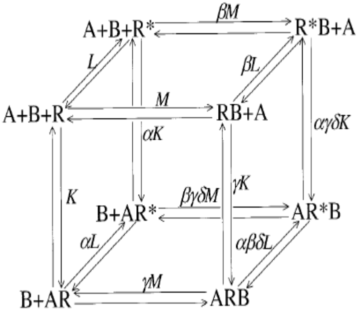

- $L$ - receptor isomerization constant
- $K$ ($M$) - equilibrium dissociation constant for $A$ ($B$)
- $\gamma$ - binding cooperativity for $A$ and $B$
- $\delta$ - activation cooperativity $A$ and $B$
- $\alpha$, $\beta$ - intrinsic efficacies

- $[A]$ ($[B]$)-orthosteric (allosteric) ligand concentration
- $[R]_a$ ($[R]_t$)- activated (total) receptor concentration

$$\frac{[R]_a}{[R]_t} = \frac{KL(M+\beta)[B] + \alpha L(M+\beta\gamma\delta[B])[A]}{K(M(1+L)+(1+\beta L)[B]) + (M(1+\alpha L)+\gamma(1+\alpha\beta\delta L)[B])[A]}$$

# A parameterized model in pharmacology

## Allosteric two-state model of receptor activation




What is known?

- Experimental values for $[A]$, $[B]$, $[R]_a/[R]_t$

What is required?

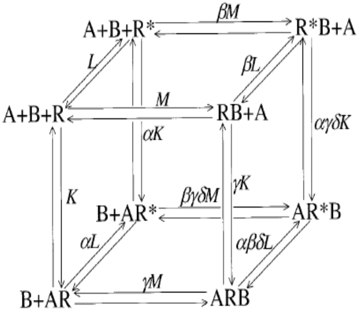
- $K$, $L$, $M$, $\alpha$, $\beta$, $\gamma$, $\delta$

Typical approach:

- minimize the mean squared error ($MSE$)

# A parameterized model in pharmacology

## Allosteric two-state model of receptor activation




What is known?

- Experimental values for $[A]$, $[B]$, $[R]_a/[R]_t$

What is required?

- $K$, $L$, $M$, $\alpha$, $\beta$, $\gamma$, $\delta$

Typical approach:

- minimize the mean squared error (*MSE*)

Particularities:

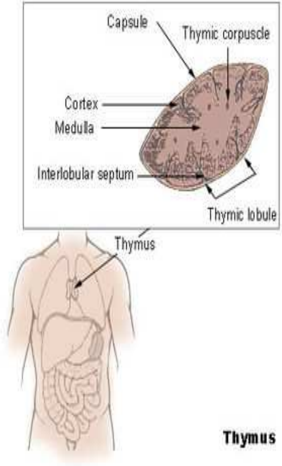
- explicit relationship between *MSE* and the parameters
- not easy to establish initial approximations for the parameters
- possible several equally good sets of parameters $\Longrightarrow$ multi-modal optimization

[D.Roche et al, 2013]

# A parameterized model in immunology

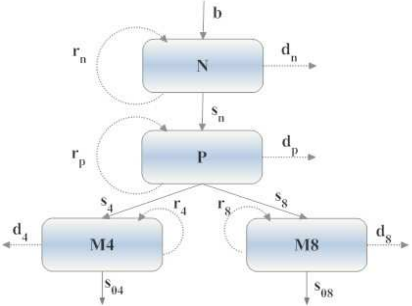
## Compartmental models of the thymus

- the thymus is an import organ of the immune system
- its main role is to produce various types of thymocytes (T cells):
  - double negative T cells (N population)
  - double positive T cells (P population)
  - single positive T cells (M4 and M8 populations)
- there are several processes involving the T cell populations:
  - cell proliferation (growth)
  - cell death (involution)
  - cell maturation and differentiation (transfer)



# A parameterized model in immunology

## Compartmental models of the thymus




**Parameters:**

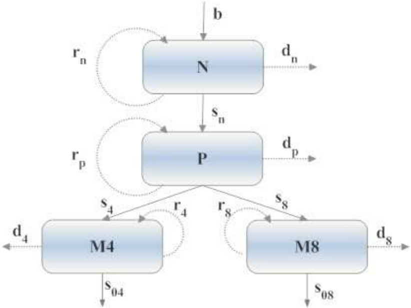
- proliferation rates: $r_n$,$r_p$,$r_4$,$r_8$
- death rates: $d_n$,$d_p$,$d_4$,$d_8$
- transfer rates: $s_n$,$s_4$,$s_8$,$s_{o4}$,$s_{o8}$
- bone marrow inflow rate: $b$
- carrying capacities: $K$, $K_n$

**Example:** Mehr's model [Mehr et al, 1996]

$$\begin{cases} \dot{N}(t) &= r_n N(t)(1 - N(t)/K_n) - d_n N(t) - s_n N(t) + b(1 - N(t)/K_n) \\ \dot{P}(t) &= r_p P(t)(1 - Z(t)/K) - d_p P(t) - (s_4 + s_8)P + s_n N(t) \\ \dot{M}_4(t) &= r_4 M_4(t)(1 - Z(t)/K) - d_4 M_4(t) - s_{04} M_4(t) + s_4 P(t) \\ \dot{M}_8(t) &= r_8 M_8(t)(1 - Z(t)/K) - d_8 M_8(t) - s_{08} M_8(t) + s_8 P(t) \\ Z(t) &= N(t) + P(t) + M_4(t) + M_8(t) \end{cases}$$

# A parameterized model in immunology

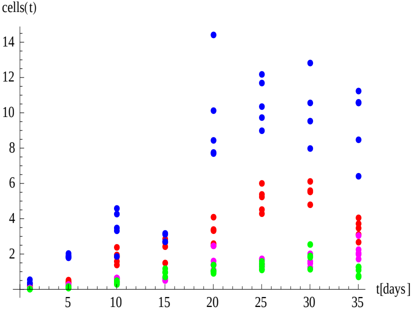
## Compartmental models of the thymus




### What is required?

- proliferation/ death/ transfer/ inflow rates
- carrying capacities ($K_n$, $K$)

### What is known ?




Experimental estimates of the number of cells during their evolution ($N$-red, $P$-blue, $M_4$-magenta, $M_8$-green)

# A parameterized model in immunology

## Compartmental models of the thymus

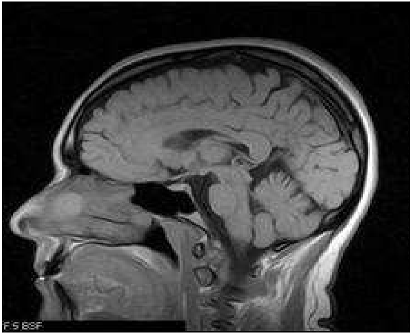
Particularities:

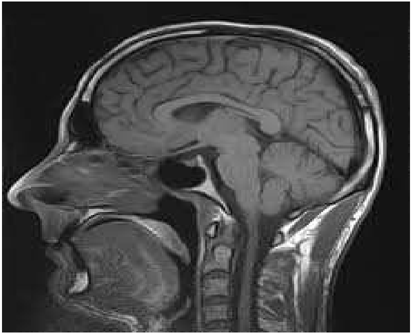
- search for parameters minimizing the mean squared error betwen the numerically estimated solutions of the ODE and the experimental data
- "semi-transparent" model: no explicit analytical relationship between *MSE* and the parameters to be estimated
- constraints:
  - $K_n < K$ (relationship between the carrying capacities)
  - desired evolution of the populations of cells (e.g. steady-state or involution)

[PNII-ID-PCE: REVISAL - Modeling and simulation of the dynamics of thymocyte populations and cells of the thymus medulla under normal and pathological situations, 2012-2014]

# A parameterized software module

## Image registration





- Registration: Find a transformation which maps pixels of one image to corresponding pixels of the other image

[D. Gil et al., 2013]

# A parameterized software module

## Image registration

Particularities:

- What is available: a software module implementing a registration algorithm and maximal error threshold
- What is required: parameters of the registration algorithm such that the error is below the threshold and the execution time of the algorithm is minimized
- ”opaque” model: almost nothing is known about the algorithm inside the software module (it receives the parameters and provides the error and the running time)
- constraints on parameters (e.g $p_1 < p_2$) and on the error ($err < \epsilon$)
- the values provided for the execution time can be uncertain

# Summary: types of parameterized models

## "Transparent" models

- the search criterion depends in an explicit way on the parameters
- gradient information could be available

# Summary: types of parameterized models

## ”Transparent” models

- the search criterion depends in an explicit way on the parameters
- gradient information could be available

## ”Semi-transparent” models

- the search criterion depends only in an implicit way on the parameters
- its evaluation requires numerical solutions of some equations

# Summary: types of parameterized models

## ”Transparent” models

- the search criterion depends in an explicit way on the parameters
- gradient information could be available

## ”Semi-transparent” models

- the search criterion depends only in an implicit way on the parameters
- its evaluation requires numerical solutions of some equations

## ”Opaque” models

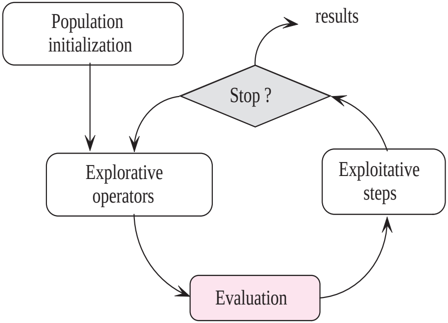
- the evaluation of the search criterion is based on a ”black box” module
- nothing is known about the influence of the parameters on the search criterion

# Outline

# Why using an evolutionary search?

- it requires minimal knowledge on the model
- if properly designed it ensures a good exploration of the parameter space
- it can be used for:
  - multimodal/ multiobjective/ constrained optimization

  and can provide several results of similar quality




Population-based stochastic search

- Exploration: mutation and crossover (reproduction)
- Exploitation: selection

# Which search method to use?

Many options:

- Evolution Strategies (ES)
- Covariance Matrix Adaptation ES (CMA-ES)
- Particle Swarm Optimization (PSO)
- Differential Evolution (DE)
- Ant Systems (AS)
- Harmony Search (HS)
- Artificial Bees Colonies (ABC)
- other nature inspired metaheuristics (e.g. firefly, cuckoo, bacterial foraging, gravitational search)

# Which search method to use?

Many options:

- Evolution Strategies (ES)
- Covariance Matrix Adaptation ES (CMA-ES)
- Particle Swarm Optimization (PSO)
- Differential Evolution (DE)
- Ant Systems (AS)
- Harmony Search (HS)
- Artificial Bees Colonies (ABC)
- other nature inspired metaheuristics (e.g. firefly, cuckoo, bacterial foraging, gravitational search)

Selection criteria:

- appropriateness - ability to deal with the problem characteristics
- simplicity - easy to understand/ implement
- competitiveness - good behavior for test functions
- availability - easy to find implementations

# Common types of constraints

## From simple to complex constraints

- Bounding box constraints: $x \in [a, b]$

### Repairing rules:

- iterate the reproduction operator until the offspring satisfies the constraint
- use a symmetry based rule, i.e. when $x \notin [a, b]$ iterate:

$$x' = \begin{cases} b - (x - b) & \text{if } x > b \\ a + (a - x) & \text{if } x < a \end{cases}$$

  until $x' \in [a, b]$.

- select randomly an element in the search range

# Common types of constraints

## From simple to complex constraints

**Constraints on the set of feasible values: $x \in [a, b] \cap \mathbb{Z}$**

- use specific operators to generate new elements
- use operators for search in continuous spaces + apply a rounding function

**Simple inequality constraints: $x < y$**

- instead of searching for $x$ and $y$
- search for $x$ and $\delta > 0$ (such that $y = x + \delta$)

# Common types of constraints

## From simple to complex constraints

### General form of a constrained optimization problem

Find $x$ which minimizes $f(x)$ subject to

- $g_i(x) \leq 0, \quad i = 1, \ldots, m$
- $h_j(x) = 0, \quad j = 1, \ldots, p$ (usually transformed in $|h_j(x)| < \epsilon$)

### Types of constraints

- $g_i$ and $h_j$ depend explicitely on $x$ - easy to check if the constraint is satisfied
- there is no explicit dependence between $g_i$, $h_j$ and $x$ (e.g. the constraint could just say that the estimated model behaves in a given way) - only a degree of constraint satisfaction can be estimated

# Dealing with constraints in evolutionary search

## Penalty method

### Main idea:

combine the objective function with a penalty function measuring the degree of violating the constraints

$$F(x) = f(x) + \sum_{i=1}^{m} r_i \max(0, g_i(x)) + \sum_{j=1}^{p} c_j |h_j(x)|$$

### Advantage:

- easy to implement

### Disadvantage:

- the choice of the penalty weights is not obvious

# Dealing with constraints in evolutionary search

## Feasibility rules

**Main idea (Deb's feasibility rule):**

use separate objective value ($f$) and penalty value (degree of constraint violation - $\phi$) when compare two elements; $x$ is better than $x'$ if:

- $x$ and $x'$ are both feasible and $f(x) < f(x')$
- $x$ is feasible and $x'$ is not feasible
- $x$ and $x'$ are both unfeasible and $\phi(x) < \phi(x')$

**Advantages:**

- easy to implement and to couple with various search algorithms
- it does not require parameters

# Dealing with constraints in evolutionary search

## Feasibility rules

### Main idea (Deb's feasibility rule):

use separate objective value ($f$) and penalty value = degree of constraint violation ($\phi$) when compare two elements; $x$ is better than $x'$ if:

- $x$ and $x'$ are both feasible and $f(x) < f(x')$
- $x$ is feasible and $x'$ is not feasible
- $x$ and $x'$ are both unfeasible and $\phi(x) < \phi(x')$

### Disadvantage:

- separating the constraints and the objective function can lead to diversity loss (because they strongly favour the feasible solutions)

# Dealing with constraints in evolutionary search

## Stochastic ranking

### Main idea:

decides randomly which selection criterion to use (objective or penalty function)

$x$ is better than $x'$ if

$$\begin{cases} ((\phi(x) = \phi(x') = 0) \text{ or } (\text{rand}(0,1) < P_f)) \text{ and } (f(x) < f(x'))) \\ \phi(x) < \phi(x') \end{cases}$$

### Advantages:

- it limits the diversity loss (by accepting promising but unfeasible candidates)

# Dealing with constraints in evolutionary search

## Stochastic ranking

### Main idea:

decides randomly which selection criterion to use (objective or penalty function)

$x$ is better than $x'$ if

$$\begin{cases} ((\phi(x) = \phi(x') = 0) \text{ or } (\mathrm{rand}(0,1) < P_f)) \text{ and } (f(x) < f(x'))) \\ \phi(x) < \phi(x') \end{cases}$$

### Disadvantages:

- it requires the specification of a parameter ($P_f$, e.g. $P_f = 0.45$)

# Outline

# Case study: perturbed thymocyte dynamics

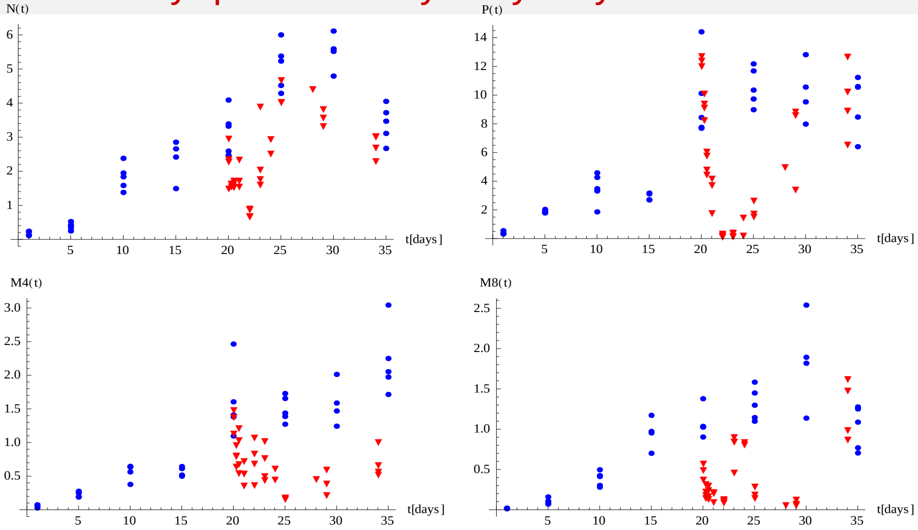
## Context

- the dynamics of thymocytes is very sensitive to any pathological situation associated with an increase of glucocorticoids (e.g. diabetes,obesity, infections)
- by administrating a particular glucocorticoid (dexamethasone - DXM) one can induce a transient involution of thymus activity similar with that common in pathological situations
- it would be useful to model this transient perturbation and to extract information on the impact DXM has on various mechanisms (cell proliferation, death and differentiation)

## Experimental data [dr. F. Mic, UMF Timisoara]

- more than 70 experiments on young and adult mice (before and after a treatment with DXM)

# Case study: perturbed thymocyte dynamics




Normal thymus data

Thymus under treatment

# Case study: perturbed thymocyte dynamics

## What is known ?

- DXM induces a significant depletion of thymocytes, especially of DP cells
- after treatment the thymus rebounds and the pre-treatment level is almost reached after 14 days

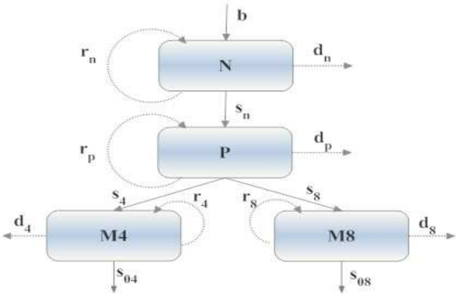
## What would be useful to know?

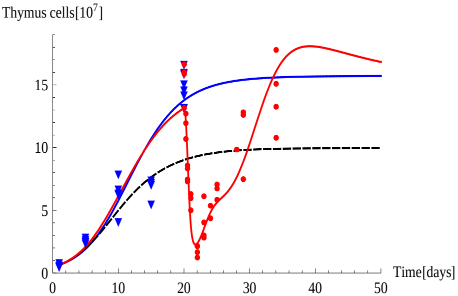
- can the perturbation induced by DXM be modelled through transiently perturbed rates/ mechanisms?
- which of the mechanisms (proliferation, death, differentiation) is most affected?
- when does the perturbation on each mechanism reaches the maximal value?

# Case study: perturbed thymocyte dynamics

## Possible approach

- start from a model describing the normal dynamics (e.g. a multi-compartmental model)
- introduce transient perturbations into the model (e.g perturbed rates, transient inhibition of proliferation)



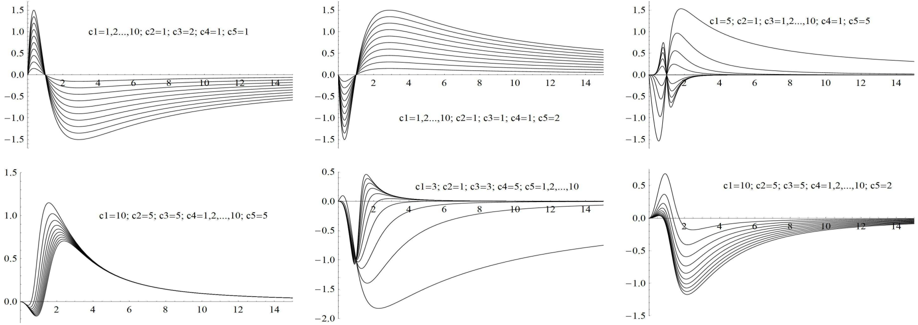



# Perturbed rates

- Main idea: additive perturbation of rates
- Perturbing functions family

$$\xi(C;t) = \frac{c_1}{t^{c_3}+c_2} - \frac{c_1 c_4/c_2}{t^{c_5}+c_4}, c_i > 0$$

- Perturbed rates: $r + \xi(C;t)$



# Parameters estimation

- Optimization problem: find the parameters which
  - minimize the mean squared error (MSE)
  - satisfy constraints concerning the positivity of all perturbed rates and the vanishing of the perturbation

$$MSE(x) = \frac{1}{4n} \sum_{\pi \in \{N,P,M_4,M_8\}} \left( \frac{1}{\max_{j=\overline{1,n}} \{\bar{\pi}_j^2\}} \sum_{j=1}^{n} (\pi(x;t_j) - \bar{\pi}_j)^2 \right)$$

- $n$ = number of experimental values
- $\bar{\pi}$ = experimental values corresponding to each of the four populations
- $\pi(x;t)$ = numerically estimated solutions

# Constraints on parameters

- For each perturbed rate, two constraints should be satisfied

$$r + \xi(C;t) \geq 0 \text{ for all } t \in [t_a, t_f]$$

$$|\xi(C;t_f)| < \epsilon_f$$

- $t_a$ = time moment when the perturbation starts (treatment administration moment)
- $t_f$ = time moment when the perturbation should be small enough (the effect of treatment is passed)
- $\epsilon_f$ = small enough value (negligible perturbation)

# Constraints on parameters

## Main difficulty in dealing with the constraints

- the constraints on positivity of rates cannot always be checked exactly
- in some cases, sufficient conditions for positivity can be found but usually they are not also necessary:

$$r \geq \max\{c_1/(c_2^2 + c_2), c_1 c_4/(c_2^2 + c_2 c_4 + c_2)\}$$

- need to define a degree of constraint satisfaction

$S_p^j(C_j) = 1$ if the sufficient condition is satisfied, otherwise

$$S_p^j(C_j) = \frac{\text{card}\{t \in T_h | r_j + \xi(C_j; t) > 0\} - \delta}{\text{card}(T_h)}$$

$$T_h = \{t_a, t_a + h, \ldots, t_f\}, \quad h > 0 - \text{discretization step}, \quad \delta > 0$$

# Constraints on parameters

Positivity constraint:

$$S_p^j(C_j) = \frac{\mathrm{card}\{t \in T_h | r_j + \xi(C_j;t) > 0\} - \delta}{\mathrm{card}(T_h)}$$

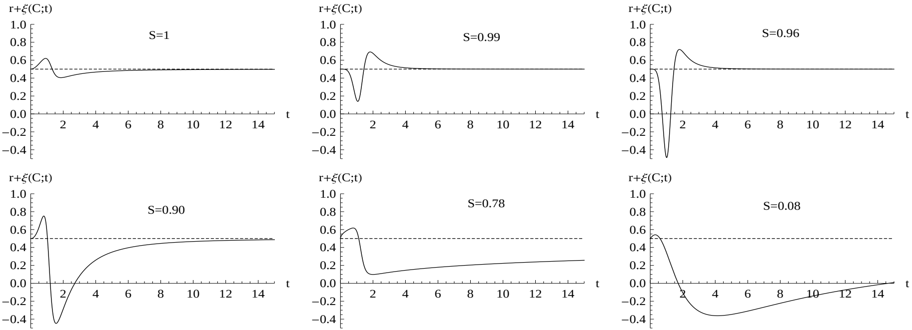
Perturbation vanishing constraint:

$$S_v^j(C_j) = \begin{cases} 1 & \text{if } |\xi(C_j;t_f)| \leq \epsilon_f \\ 1 - \min\{1, |\xi(C_j;t_f)|\} & \text{otherwise} \end{cases}$$

Combined constraints satisfaction degree:

$$S(C) = \prod_{j=1}^{q} S_p^j(C_j) S_v^j(C_j) \in [0,1]$$

# Constraints on parameters




Constraints handling: different ways of combining the satisfaction degree with MSE

# Various comparison rules

*$x$ is considered better than $x'$ if one condition is satisfied*

**Rule A ($\theta$ - threshold for the satisfaction degree, e.g.$\theta = 0.99$):**

- $S(x) \geq \theta$ and $S(x') < \theta$;
- $S(x) \geq \theta$ and $S(x') \geq \theta$ and $MSE(x) < MSE(x')$;
- $S(x) < \theta$ and $S(x') < \theta$ and $S(x) \geq S(x')$

(similar to Deb's rule)

**Rule B:**

- $S(x) \geq \theta$ and $S(x') < \theta$
- $S(x)S(x') = 0$ and $MSE(x) \leq MSE(x')$
- $S(x) \neq 0$, $S(x') \neq 0$ and $MSE(x)/S(x) \leq MSE(x')/S(x')$

# Various comparison rules

*$x$ is considered better than $x'$ if one condition is satisfied*

**Rule C:**

- $S(x) > 0$ and $S(x') = 0$
- $S(x) = 0$, $S(x') = 0$ and $MSE(x) \leq MSE(x')$
- $S(x) \neq 0$, $S(x') \neq 0$ and $MSE(x)/S(x) \leq MSE(x')/S(x')$.

# Various comparison rules

*$x$ is considered better than $x'$ if one condition is satisfied*

**Rule D ($S$ interpreted as a probability that the constraint is satisfied):**

- $U_1 \leq S(x)$ and $U_2 > S(x')$;
- $U_1 > S(x)$ and $U_2 > S(x')$ and $MSE(x) \leq MSE(x')$;
- $U_1 \leq S(x)$ and $U_2 \leq S(x')$ and $MSE(x)/S(x) \leq MSE(x')/S(x')$

($U_1$ and $U_2$ are random values uniformly distributed on $[0,1]$)

**Rule E (inspired by stochastic ranking):**

- $S(x) \geq \theta$, $S(x') \geq \theta$ and $MSE(x) < MSE(x')$
- $U < P_f$, $S(x)S(x') \neq 0$, $MSE(x)/S(x) < MSE(x')/S(x')$
- $S(x) \geq S(x')$

# Evolutionary search

Initialization: $x_i = U(a_i, b_i), \quad i = \overline{1,m}$

while ⟨ NOT termination ⟩ do

- Mutation:

$$y_i = x_{r_1} + F \cdot (x_{r_2} - x_{r_3}), \quad i = \overline{1,m}$$

- Crossover:

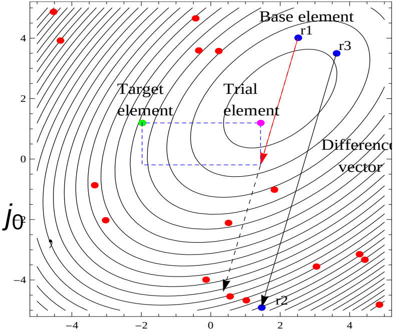
$$z_i^j = \begin{cases} y_i^j & \text{if } rand(0,1) < CR \text{ or } j = j_0 \\ x_i^j & \text{otherwise} \end{cases}$$

$$i = \overline{1,m}, j = \overline{1,n}$$

- Selection:

$$x_i(g+1) = \begin{cases} z_i & \text{if } f(z_i) \leq f(x_i(g)) \\ x_i^j & \text{if } f(z_i) > f(x_i(g)) \end{cases}$$

## Differential Evolution [Storn&Price, 1995]




$m$ - population size

$F \in (0,2)$ - scale factor

$CR \in [0,1]$ - crossover rate

$j_0$ - randomly selected component

# Evolutionary search

JADE algorithm [Zhang&Sanderson, 2009]:

- self-adaptive version of Differential Evolution

$$z_i^l = \begin{cases} x_i^l + F_i \cdot (x_{rbest}^l - x_i^l) + F_i \cdot (x_{r1}^l - x_{r2}^l) & \text{if } rand() \leq CR_i \\ x_i^l & \text{otherwise} \end{cases}$$

JADE particularities

- $x_{rbest}$ chosen from the $p\%$ best elements in the population
- $x_{r2}$ chosen from an archive consisting of elements discarded by selection
- $F_i$ generated using a Gaussian distribution
- $CR_i$ generated using a Cauchy distribution
- the parameters of these distributions are adjusted during the evolutionary process

# Numerical experiments

- Experimental dataset:
  - 232 values (number of cells in each of the four thymocyte populations)
  - collected from young and adult mice thymus either before or after a treatment administration
- number of parameters: $k = 71$
- number of constraints: $q = 26$
- number of independent runs: 30

JADE parameters:

- population size: 20
- generations: 5000
- percent of best elements: $p = 10\%$

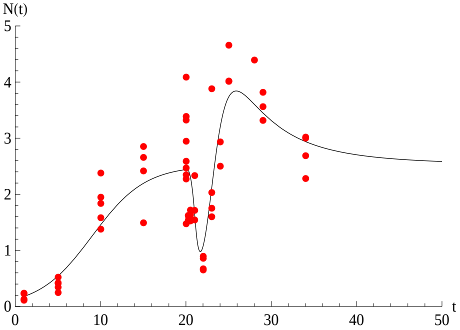
# Comparison between ranking rules

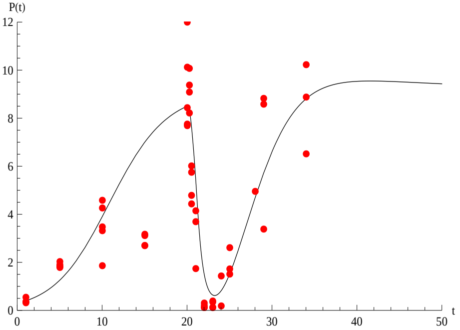
Quality of fit ($MSE$), constraints satisfaction degree ($S$), estimated feasibility probability ($FP(\theta)$ for $\theta = 0.99$).

| Rule | $MSE$ | $S$ | $FP(\theta)$ |
|---|---|---|---|
| A ($\theta = 1$) | $0.0338 \pm 0.0012$ | $1 \pm 0$ | 1 |
| A ($\theta = 0.99$) | $0.0270 \pm 0.0010$ | $0.9966 \pm 0.0033$ | 1 |
| B ($\theta = 0.99$) | $0.0268 \pm 0.0014$ | $0.9999 \pm 5 \cdot 10^{-6}$ | 1 |
| C | $0.0261 \pm 0.0009$ | $0.9878 \pm 0.0119$ | 0.45 |
| D | $0.0290 \pm 0.0017$ | $0.9999 \pm 3 \cdot 10^{-6}$ | 1 |
| E ($P_f = 0.45$) | $0.0250 \pm 0.0005$ | $0.9935 \pm 0.0011$ | 1 |
| Unconstrained | $0.0208 \pm 0.0022$ | $0.0468 \pm 0.0776$ | 0 |

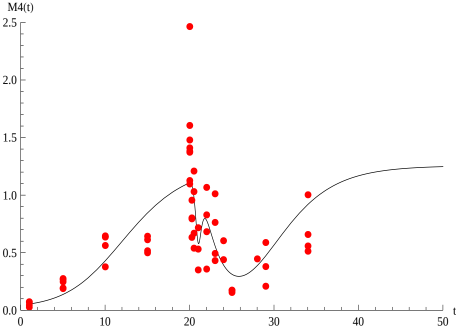
Remark: Rule E is better than the other ones (Mann-Whitney statistical test, p-value$< 10^{-5}$)

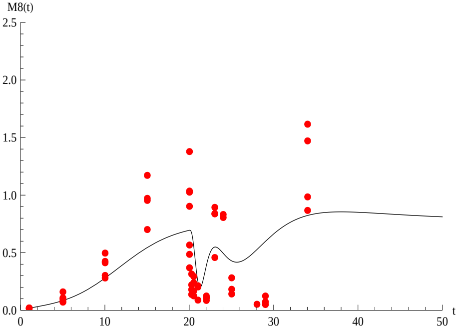
# Simulated dynamics



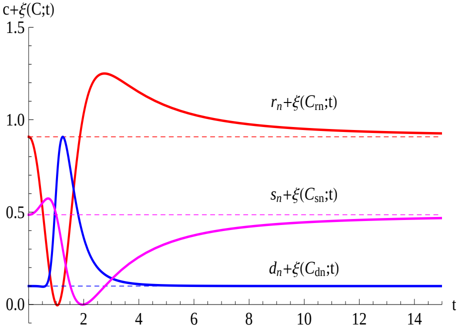



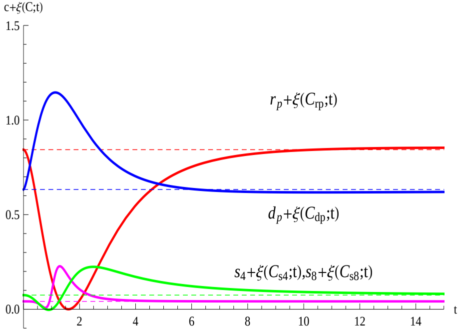
# Simulated dynamics



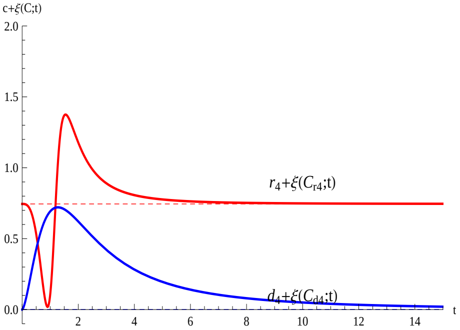



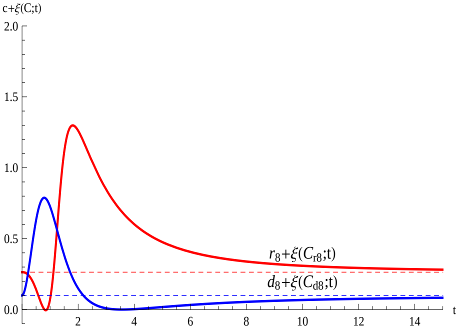
# Perturbed rates



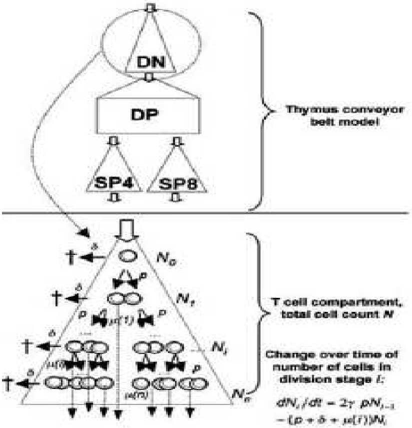



# Perturbed rates



# A multi-stage model




[Thomas-Vaslin et al., 2008]

- the proliferation process consists of several stages
- differentiation arises at each stage but with different rates
- used to model the impact of ganciclovir on thymus dynamics (when continuously administrated for 7 days)
- perturbation modelled by triggering off the proliferation process

# A multi-stage model

$$
\begin{aligned}
\dot{N}_0(t) &= \sigma_N - (r_N + d_N)N_0(t) \\
\dot{N}_i(t) &= 2\gamma(t) r_N N_{i-1}(t) - (r_N + d_N + \mu_N(i))N_i(t), \quad i = \overline{1, n_N}
\end{aligned}
$$

$$
\begin{aligned}
\dot{P}_0(t) &= \sum_{i=1}^{n_N} \mu_N(i) N_i(t) + 2\gamma(t) r_N N_{n_N}(t) - (r_P + d_P)P_0(t) \\
\dot{P}_i(t) &= 2\gamma(t) r_P P_{i-1}(t) - (r_P + d_P + \mu_P(i))P_i(t), \qquad i = \overline{1, n_P - 1} \\
\dot{P}_{n_P}(t) &= \sum_{i=1}^{n_P - 1} \mu_P(i) P_i(t) + 2\gamma(t) r_P P_{n_P - 1}(t) - \mu_{LP} P_{n_P}(t)
\end{aligned}
$$

## Parameters to be estimated

- Proliferation rates: $r_N$, $r_P$, $r_4$ and $r_8$
- Death rates: $d_N$, $d_P$, $d_4$ and $d_8$
- Transfer rates: $\mu_N(i) = (\alpha_N \cdot i)^n$, $\mu_P(i) = (\alpha_P \cdot i)^n$, $\mu_{LP}$, $e_4(i) = (\alpha_{e4} \cdot i)^n$ and $e_8(i) = (\alpha_{e8} \cdot i)^n$
- Number of stages: $n_N$, $n_P$

# A multi-stage model

$$
\begin{aligned}
\dot{M}_{40}(t) &= \alpha_4 \mu_{LP} P_{n_P}(t) - (r_4 + d_4) M_{40}(t) \\
\dot{M}_{4i}(t) &= 2\gamma(t) r_4 M_{4,i-1}(t) - (r_4 + d_4 + e_4(i)) M_{4i}(t), \qquad i = \overline{1, n_4 - 1} \\
\dot{M}_{4n_4}(t) &= 2\gamma(t) r_4 M_{4,n_4-1}(t) - (d_4 + e_4(n_4)) M_{4n_4}(t)
\end{aligned}
$$

$$
\begin{aligned}
\dot{M}_{80}(t) &= \alpha_8 \mu_{LP} P_{n_P}(t) - (r_8 + d_8) M_{80}(t) \\
\dot{M}_{8i}(t) &= 2\gamma(t) r_8 M_{8,i-1}(t) - (r_8 + d_8 + e(i)) M_{8i}(t), \qquad i = \overline{1, n_8 - 1} \\
\dot{M}_{8n_8}(t) &= 2\gamma(t) r_8 M_{8,n_8-1}(t) - (d_8 + e_8(n_8)) M_{8n_8}(t)
\end{aligned}
$$

## Parameters to be estimated

- Proliferation rates: $r_N$, $r_P$, $r_4$ and $r_8$
- Death rates: $d_N$, $d_P$, $d_4$ and $d_8$.
- Transfer rates: $\mu_N(i) = (\alpha_N \cdot i)^n$, $\mu_P(i) = (\alpha_P \cdot i)^n$, $\mu_{LP}$, $e_4(i) = (\alpha_{e4} \cdot i)^n$ and $e_8(i) = (\alpha_{e8} \cdot i)^n$
- Number of stages: $n_4$, $n_8$

# A multi-stage model

## On/off proliferation control

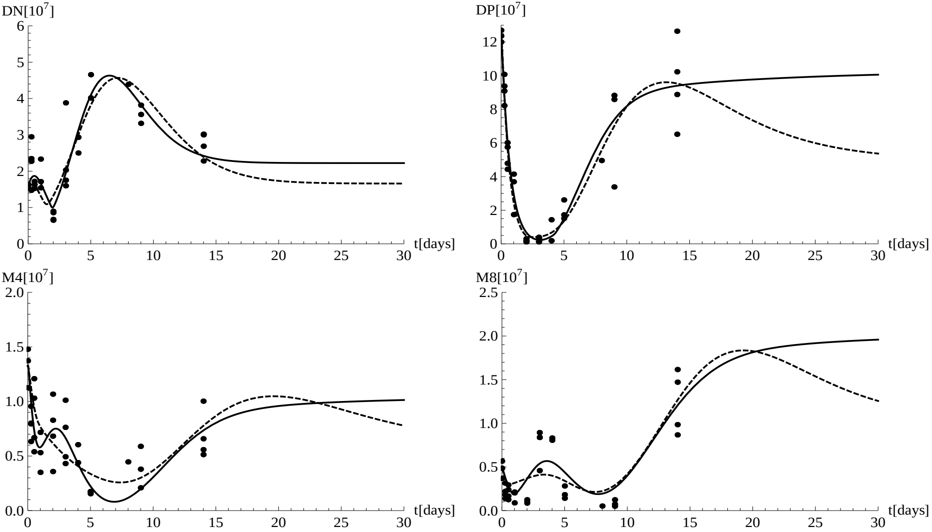
$$\gamma(t) = \begin{cases} 0 & \text{if } t < \tau_0 \\ 1 & \text{if } t \geq \tau_0 \end{cases}$$

## Continuous inhibition function

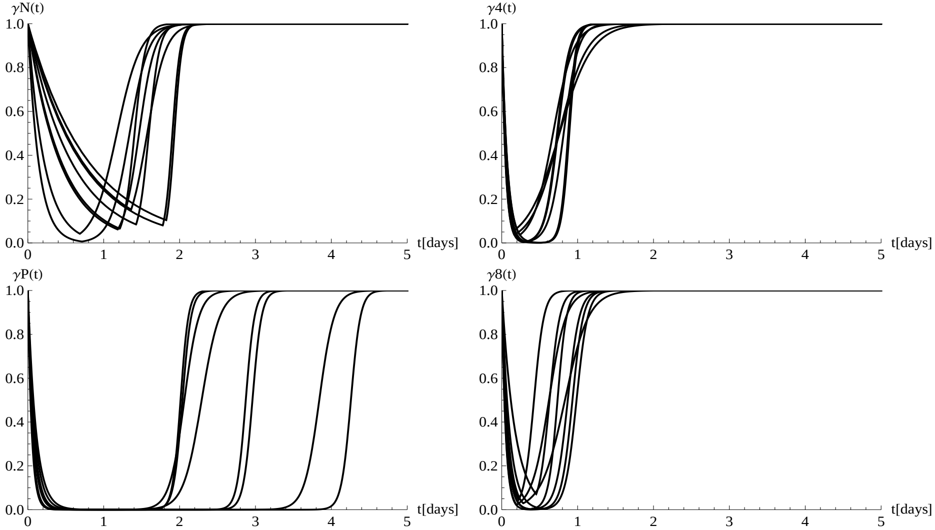
$$\gamma(t) = \begin{cases} \exp(-\delta_0 t) & \text{if } t < \tau_0 \\ 1/(1 + \exp(-\delta_1 (t - \tau_1))) & \text{if } t \geq \tau_0 \end{cases}$$

- $\tau_0$ is estimated
- exponential decrease of the proliferation rate and logistic growth of the proliferation rate
- different parameters for different populations

# A multi-stage model

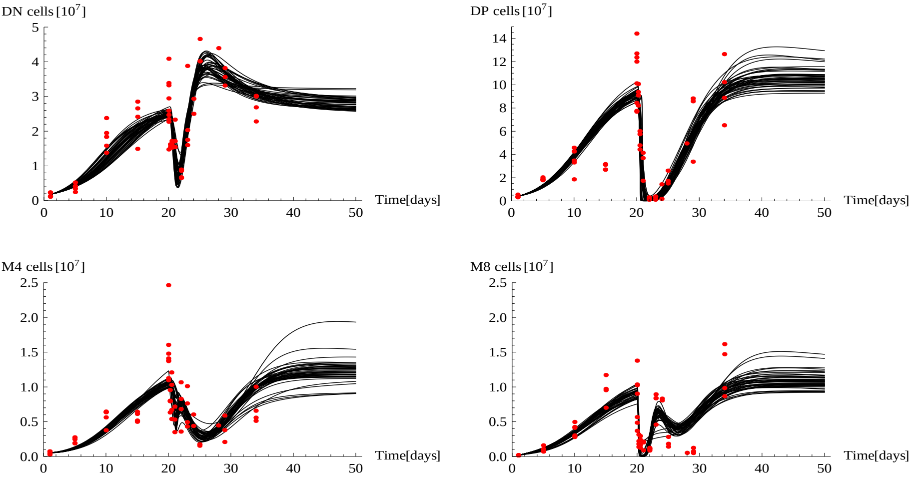


# A multi-stage model



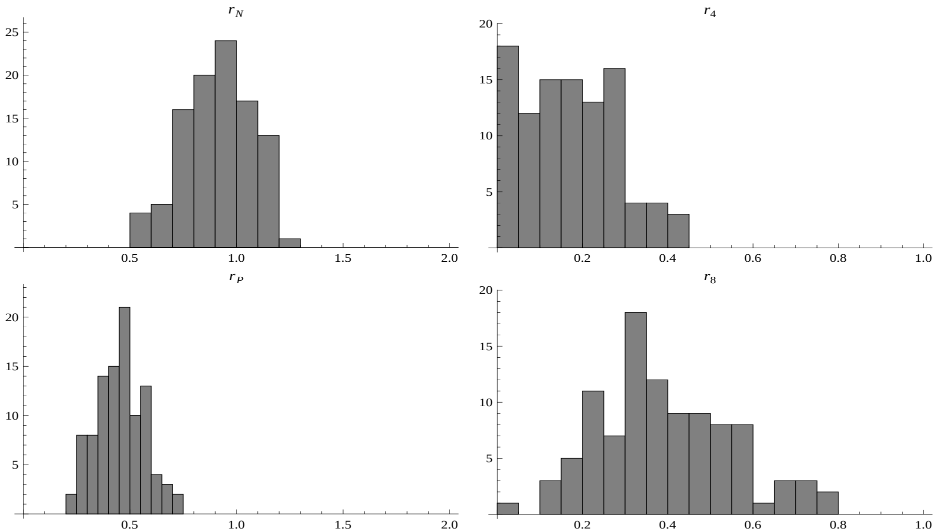
# Outline

# Exploring the output of the evolutionary search




Evolved solutions of similar quality ($MSE = 0.029 \pm 0.002$)

# Exploring the output of the evolutionary search




The evolved parameter values can be used to obtain information on the distribution of parameters

# Summary

- in the case of ”semi-transparent” or ”opaque” models, estimating the parameters can lead to difficult constrained optimization problems
- for some real problems it is not easy the check if a constraint involving the parameters is satisfied or not; estimating a constraint satisfaction degree could be useful
- combining the MSE value with the constraint satisfaction degree could be beneficial
- an evolutionary search for parameters can provide several possible solutions $\Longrightarrow$ information about the distribution of parameters

# Invitation to Timisoara - Romania

- ... at the High Performance Research Center (http://hpc.uvt.ro)
    - hybrid CPU+GPU cluster (450 cores, 7 Nvidia Tesla based blades, 40Gbps Infiniband, 750 GB RAM, 30TB storage)
    - IBM BlueGene/P (4096 cores, 11.7 Tflops, 4TB RAM, 28TB storage)
- ... open acces offered through FP7-REGPOT project HOST (http://host.hpc.uvt.ro)
- ... current research topics
    - Cloud computing technologies for HPC service exposure
    - Scheduling algorithms and techniques
    - Parallel computing in remote data processing
    - Large scale numerical computations
    - HPC-based intelligent services

# Invitation to Timisoara - Romania

... at SYNASC 2013 - 15th Symposium on Symbolic and Numeric Algorithms for Scientific Computation - 23-26 September 2013 (http://synasc13.info.uvt.ro)

Program chair: Nikolaj Bjorner, Microsoft Research

Tracks:

- Symbolic Computation
- Numeric Computing
- Logic and Programming
- Distributed Computing
- Artificial Intelligence
- Advances in the Theory of Computing

Invited speakers

- Ivona Brandic, Vienna University of Technology
- Gabriel Ciobanu, Romanian Academy, Iasi
- Leonardo de Moura, Microsoft Research, USA
- Grigore Rosu, University of Illinois at Urbana-Champaign
- Dan Simovici, University of Massachusetts, Boston

# Invitation to Timisoara - Romania

... at SYNASC 2013 - 15th Symposium on Symbolic and Numeric Algorithms for Scientific Computation - 23-26 September 2013 (http://synasc13.info.uvt.ro)

Workshops:

- ACSys: Agents for Complex Systems
- NCA: Natural Computing and Applications
- HPCSP: High Performance Computing for Scientific Problems
- MICAS: Management of Resources in Sky and Cloud Computing
- IAFP: Iterative Approximation of Fixed Points

Submission deadline for workshops: 15-30 July 2013 (depending on the workshop)