Parameter estimation in Systems Biology

Marco S. Nobile – marco.nobile@unimib.it

University of Milano-Bicocca, Department of Informatics, Systems and Communication

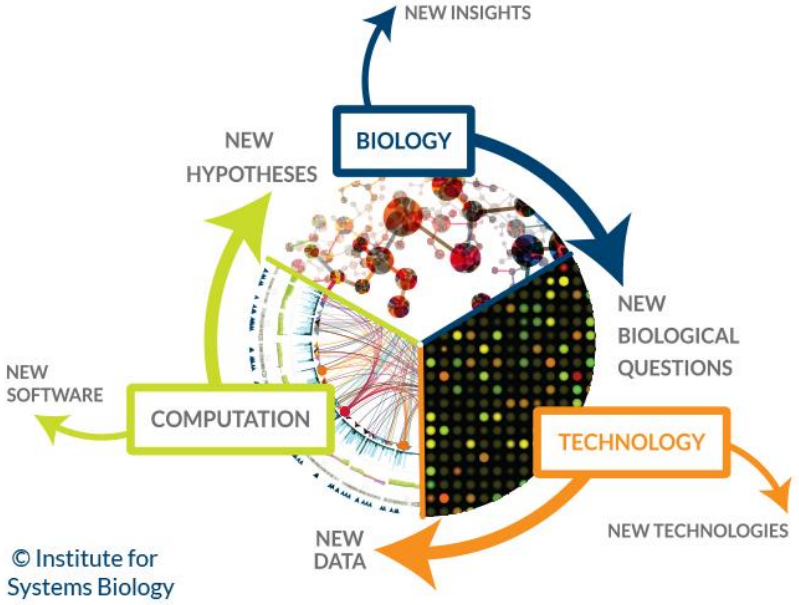
PhD program in Computer Science

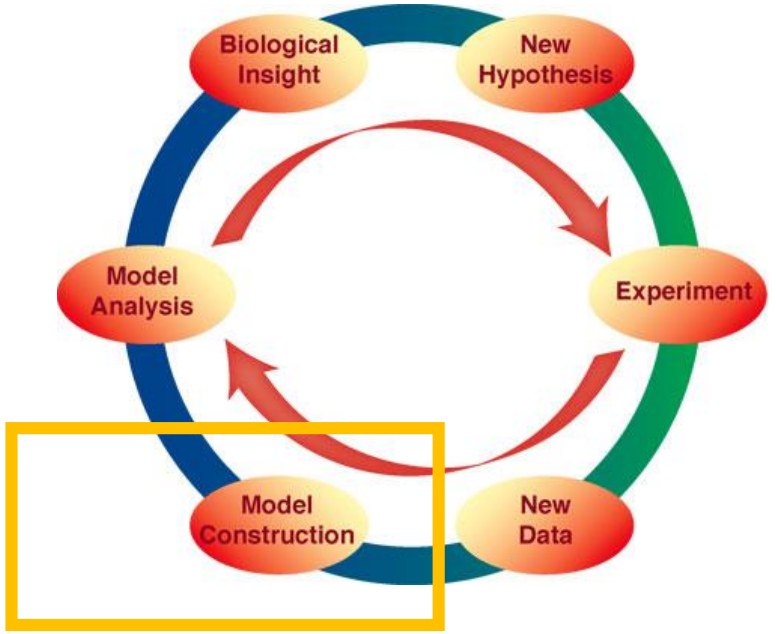
Outline

- Systems Biology
- The problem of Parameter Estimation (PE)
- Fitness functions for complex biological systems
- Benchmark functions *vs* PE
- The impact of initialization
- The impact of representation
- Acute Myeloid Leukemia

Systems Biology

- **Systems Biology** investigates **biological systems** by means of combined "**wet**" (=laboratory) and "**dry**" (=computational) **experiments** [Kitano, Nature 2001]
  - "Dry" investigation is generally performed using **rigorous mathematical models** describing the **fundamental mechanisms** of the biological system
  - Models are **analyzed** using **simulations**
  - Analyses lead to new **insights**, new **hypotheses** and, possibly, **novel laboratory experiments**





Reaction-based models

- One widespread modeling approach for cellular systems is by means of a set of M biochemical reactions involving N chemical species with the following form:

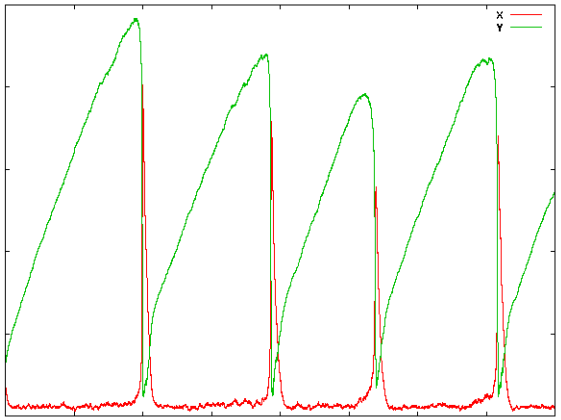
$$R_i: \sum_{j=1}^{N} a_{ij} \cdot s_j \xrightarrow{k_i} \sum_{j=1}^{N} b_{ij} \cdot s_j \,, \qquad i = 1, \dots, M$$

  - $a_{ij}, b_{ij} \in \mathbb{N}$ denote the **stoichiometric coefficients**, i.e., the number of molecules of the chemical species s_j involved in reaction R_i as **reactants** and **products**, respectively
  - $k_i \in \mathbb{R}^+$ denotes the **kinetic** (or stochastic) constant determining the **rate** of reaction R_i
- All this data is **mandatory** to perform a dynamic simulation
- The vector of kinetic parameters $\mathbf{k} = (k_1, \dots, k_M)$ are difficult (sometimes impossible) to measure in laboratory experiments

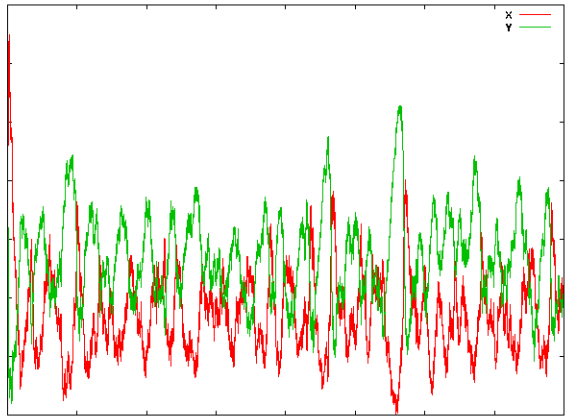
The importance of (accurate) model parameters

Stochastic simulation of the Brusselator model - Initial amounts:
A = 200 molecules (fixed), X = 200 molecules, B = 600 molecules (fixed), Y = 300 molecules

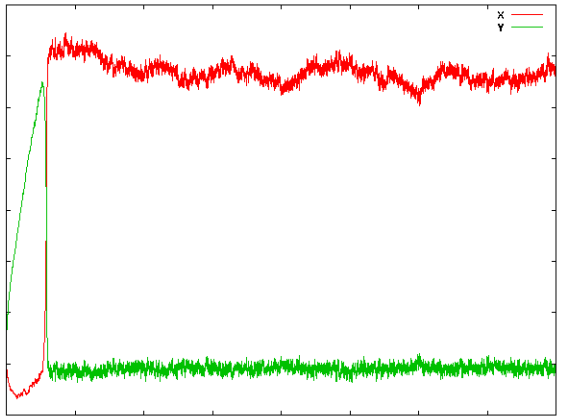
$$A \xrightarrow{k_1} X$$
$$2X + Y \xrightarrow{k_2} 3X$$
$$B + X \xrightarrow{k_3} Y + D$$
$$X \xrightarrow{k_4} E$$



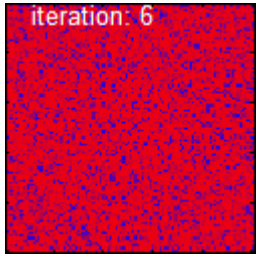
$k_1 = 1$
$k_2 = 5 \times 10^{-3}$
$k_3 = 2.5 \times 10^{-5}$
$k_4 = 1.5$



$k_1 = 1$
$k_2 = 5 \times 10^{-3}$
$k_3 = 2.5 \times 10^{-4}$
$k_4 = 1.5$

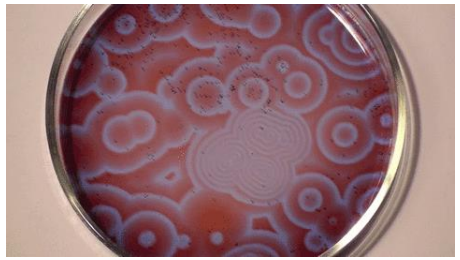


$k_1 = 1$
$k_2 = 5 \times 10^{-3}$
$k_3 = 2.5 \times 10^{-5}$
$k_4 = 1.5 \times 10^{-1}$



2D reaction-diffusion stochastic simulation of Brusselator

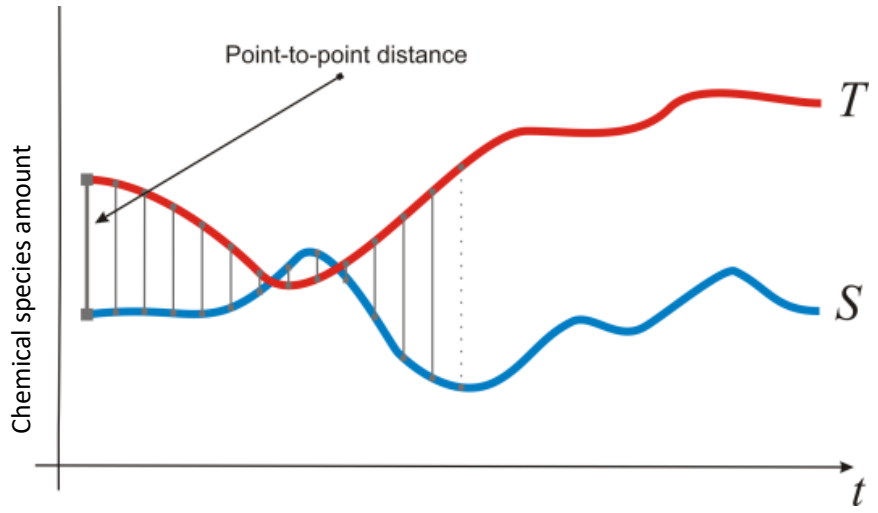
Belousov–Zhabotinsky reaction
(inspired the Brusselator model)



[Kondepudi and Prigogine, 1998]

Parameter Estimation

- **Parameter Estimation** (PE) problem: given some experimental target data (e.g., discrete-time time-series of molecular amounts), determine $\mathbf{k}$
  - PE can be re-stated as an optimization (**minimization**) problem: find a parameterization $\mathbf{k}$ such that the **distance** between the **target time-series** and the **simulated dynamics** is minimized
  - **NP-hard task**
  - Multi-modal, non-convex, non-linear, noisy…
  - **Global optimization** techniques must be exploited
    [Tangherloni *et al.*, Appl Soft Comp 2019]
    [Nobile *et al.*, IEEE CEC 2018]
    [Moles *et al.*, Genome Res 2003]
    [Dräger *et al.*, BMC Syst Biol 2009]



A few words on the fitness function

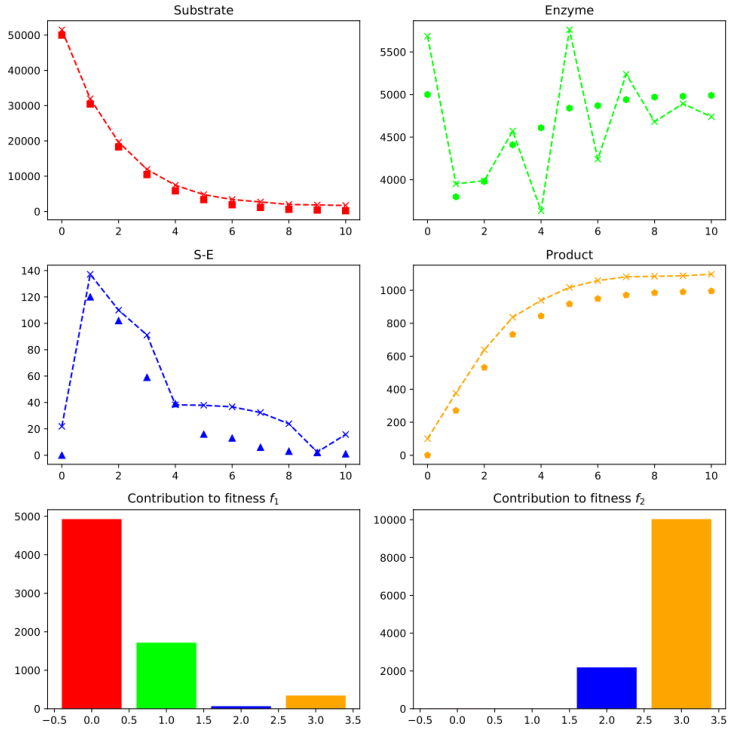
- The definition of the **fitness** is a delicate step
  - Fitness **based on pure distance** between time-series

$$f^1(k) = \frac{1}{TN}\sum_{t=1}^{T}\sum_{n=1}^{N}|X^n(t) - Y_k^n(t)|$$

  - Normalized fitness (**MAPE** – Mean Absolute Percentage Error)

$$f^2(k) = \frac{1}{TN}\sum_{t=1}^{T}\sum_{n=1}^{N}\left|\frac{X^n(t) - Y_k^n(t)}{X^n(t)}\right|$$

  - $X^n(t)$ denotes the **experimental** value of species n at time t
  - $Y_k^n(t)$ denotes the **simulated** value of species n at time t using parameterization k
  - The normalized fitness function f^2 **reduces the impact** to the final fitness value of species with **high number of molecules**
  - WARNING: if one experimental value has 0 molecules, the fitness diverges to infinity



The role of hyper-parameters

- [Dräger *et al*., BMC Syst Biol 2009] investigated several variants of **multiple optimization methods**
  - In particular, Differential Evolution (DE), Evolution Strategy (ES), Genetic Algoritms (GA), and Particle Swarm Optimization (PSO)
  - Their conclusion: **a properly set PSO** (wrt to the model) is optimal for the PE problem

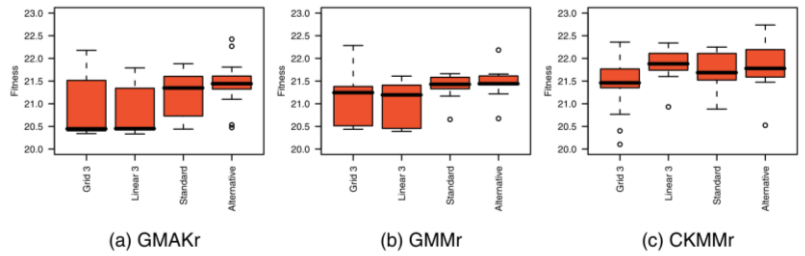


Figure 9
Influence of the settings on particle swarm optimization. Besides the standard star topology, a grid 3 and a linear 3 topology is tested on the three reversible models with $\varphi_1 = \varphi_2 = 2.05$. Furthermore, the alternative setting $\varphi_1 = 2.8$ and $\varphi_2 = 1.3$ is applied to the star topology. The grid topology performs best according to the median for the GMAKr and the CKMMr models, but is slightly worse than the linear topology on the GMMr model. All experiments are repeated 20 times.

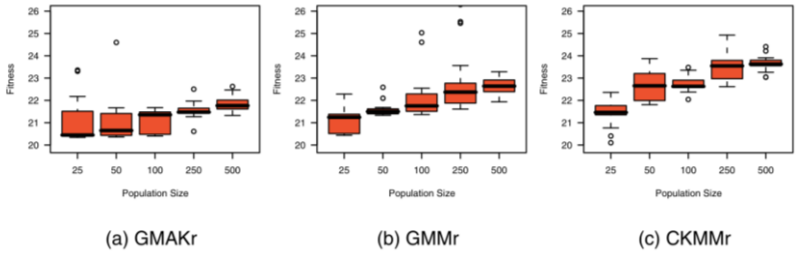


Figure 10
Influence of the population size on the performance of PSO with grid 3 topology. An increasing population size cannot improve the performance of PSO.

Table 3: Preliminary test results

GMAK, reversible					GMAK, irreversible				
Min.	Algorithm	Average	Std. Dev.	Algorithm	Min.	Algorithm	Average	Std. Dev.	Algorithm
20.334	PSO	21.190	0.576	PSO	24.587	PSO	25.967	1.171	Tribes
20.335	DE	21.228	0.756	DE	25.006	Tribes	29.502	9.610	DE
21.401	Tribes	21.725	0.275	Tribes	25.683	DE	33.169	10.143	PSO
23.097	binGA	26.106	2.204	binGA	25.981	binGA	35.670	3.125	cmaESplus
24.321	cmaESplus	27.598	2.091	cmaESplus	30.704	cmaESplus	50.663	4.138	HC MS 10
GMM, reversible					**GMM, irreversible**				
Min.	Algorithm	Average	Std. Dev.	Algorithm	Min.	Algorithm	Average	Std. Dev.	Algorithm
20.312	PSO	21.272	0.461	PSO	24.477	Tribes	24.654	0.282	Tribes
20.407	DE	21.711	1.153	DE	24.499	DE	37.696	19.374	DE
21.590	Tribes	21.887	0.243	Tribes	24.553	PSO	41.529	21.768	binGA
22.913	binGA	26.742	2.711	binGA	25.266	binGA	31.136	9.052	cmaESplus
23.890	cmaESplus	26.624	1.851	cmaESplus	25.812	stdES	31.338	0.546	HC MS 1
CKMM, reversible					**CKMM, irreversible**				
Min.	Algorithm	Average	Std. Dev.	Algorithm	Min.	Algorithm	Average	Std. Dev.	Algorithm
20.882	PSO	21.773	0.352	PSO	21.632	PSO	23.968	0.931	DE
21.821	DE	22.633	0.562	DE	22.651	DE	24.624	0.315	Tribes
22.258	Tribes	23.079	0.464	Tribes	24.191	Tribes	25.761	0.331	binGA
22.829	cmaES	24.341	1.026	cmaES	25.152	binGA	26.434	0.339	cmaESplus
23.687	binGA	24.736	0.557	cmaESplus	25.738	cmaESplus	26.539	0.210	HC MS 10

For all deterministic models (reversible models on the left, irreversible models on the right) the five absolute best algorithms and the five average best algorithms with standard deviations are listed.

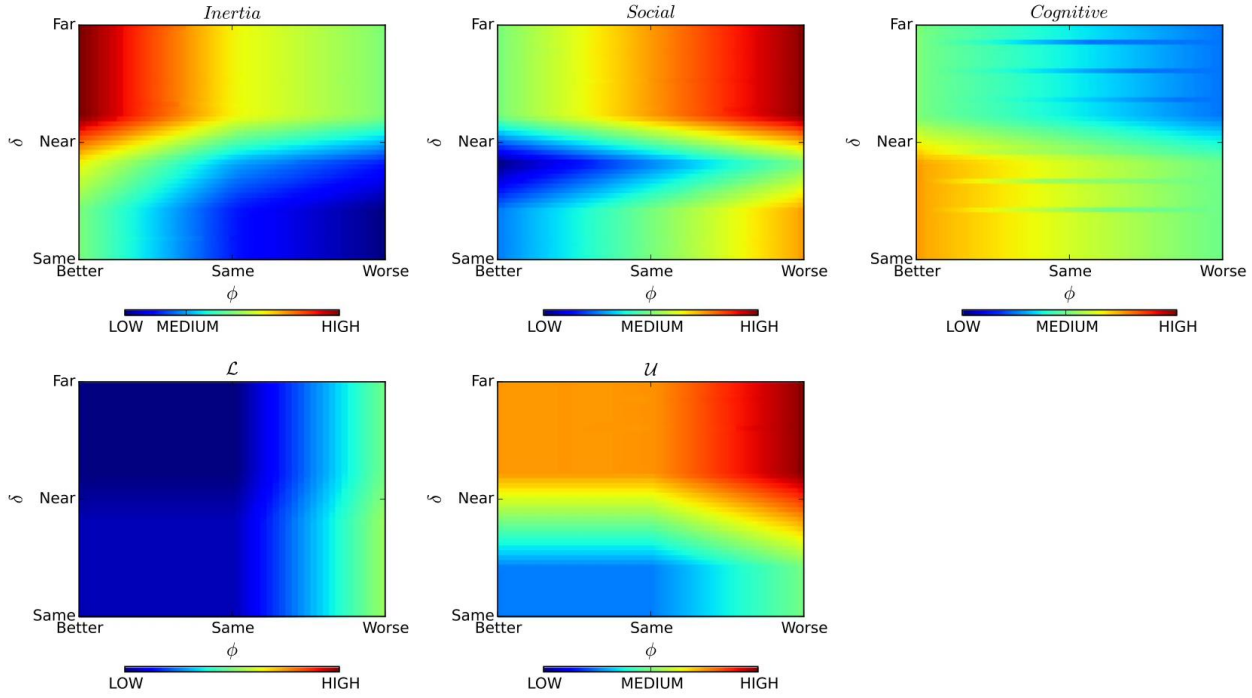
Fuzzy Self-Tuning PSO

- **Adaptive PSO** based on fuzzy reasoning [Nobile *et al.*, Swarm Evol Comp 2018]

Fuzzy rules used by FST-PSO.

Rule no.	*Rule definition*
1	if (ϕ is *Worse* or δ is *Same*) then (*Inertia* is *Low*)
2	if (ϕ is *Same* or δ is *Near*) then (*Inertia* is *Medium*)
3	if (ϕ is *Better* or δ is *Far*) then (*Inertia* is *High*)
4	if (ϕ is *Better* or δ is *Near*) then (*Social* is *Low*)
5	if (ϕ is *Same* or δ is *Same*) then (*Social* is *Medium*)
6	if (ϕ is *Worse* or δ is *Far*) then (*Social* is *High*)
7	if (δ is *Far*) then (*Cognitive* is *Low*)
8	if (ϕ is *Worse* or ϕ is *Same* or δ is *Same* or δ is *Near*) then (*Cognitive* is *Medium*)
9	if (ϕ is *Better*) then (*Cognitive* is *High*)
10	if (ϕ is *Same* or ϕ is *Better* or δ is *Far*) then ($\mathcal{L}$ is *Low*)
11	if (δ is *Same* or δ is *Near*) then ($\mathcal{L}$ is *Medium*)
12	if (ϕ is *Worse*) then ($\mathcal{L}$ is *High*)
13	if (δ is *Same*) then ($\mathcal{U}$ is *Low*)
14	if (ϕ is *Same* or ϕ is *Better* or δ is *Near*) then ($\mathcal{U}$ is *Medium*)
15	if (ϕ is *Worse* or δ is *Far*) then ($\mathcal{U}$ is *High*)

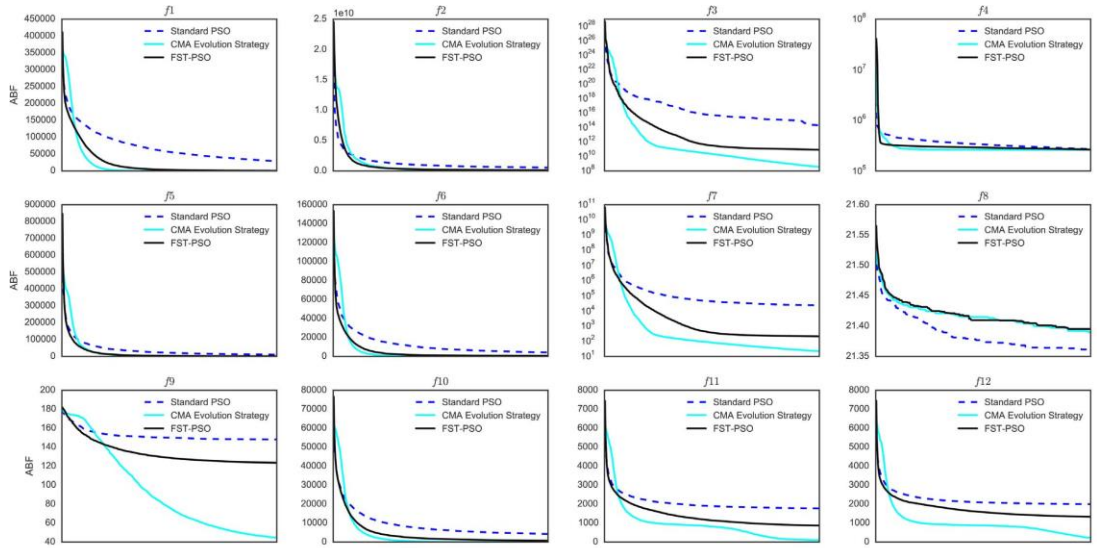
Effect of rules according to δ and ϕ (calculated with Sugeno inference engine)

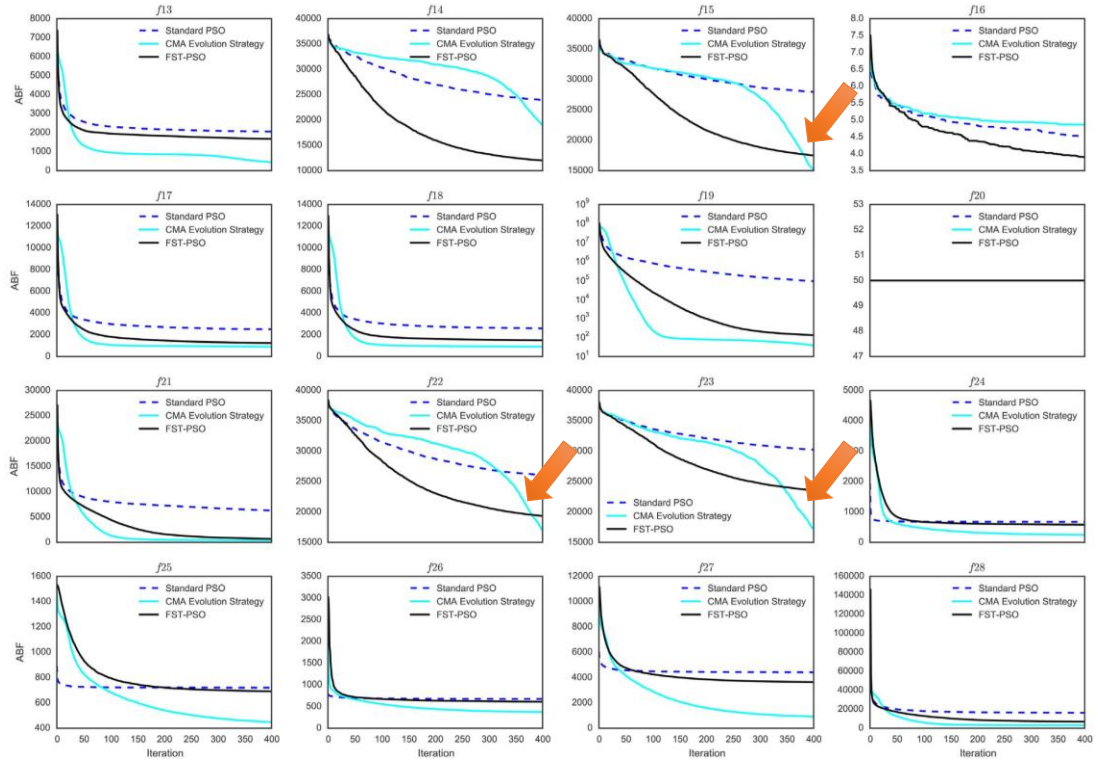


Also: damping boundary conditions [Xu *et al.*, IEEE Trans Antennas Prop 2007]; global best topology; supported initializations: uniform, log-uniform, normal, log-normal

FST-PSO *vs* CMA-ES on CEC 2013 test suite

- We tested FST-PSO and compared it against CMA-ES, the most efficient algorithm with benchmark functions in our tests. Specifically, we used the **CEC 2013 test suite** [Liang *et al*. 2013]
  - **28 benchmark functions** (rotated/shifted/multimodal/composition…), 100-dimensional search space
  - **FST-PSO performs well** in many cases
  - **Surprising recovery by CMA-ES** after ≈ 300 iterations, thanks to adaptation and «the 1/5 rule»

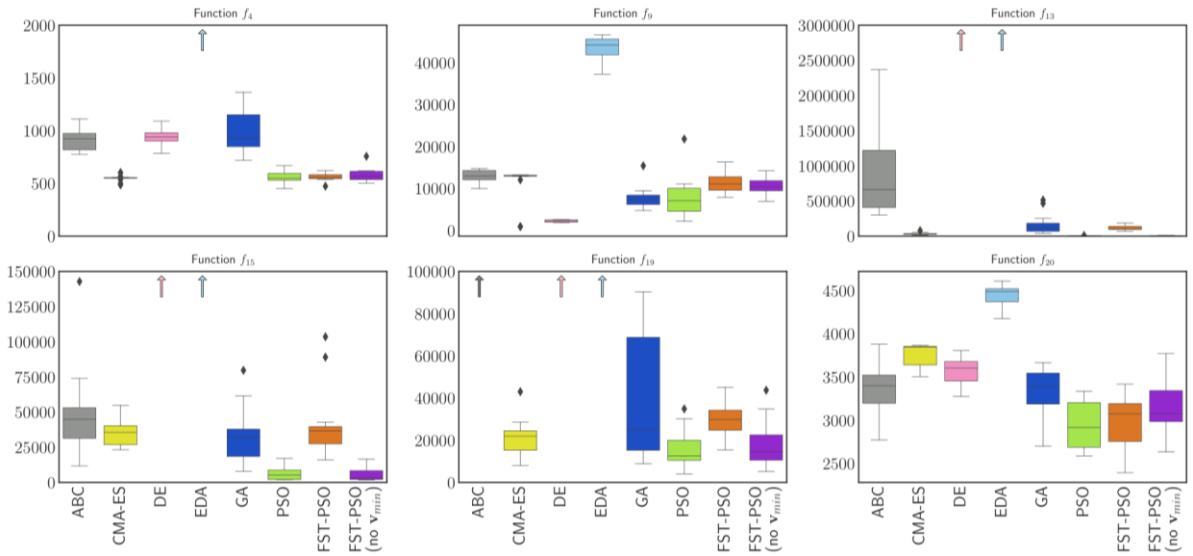




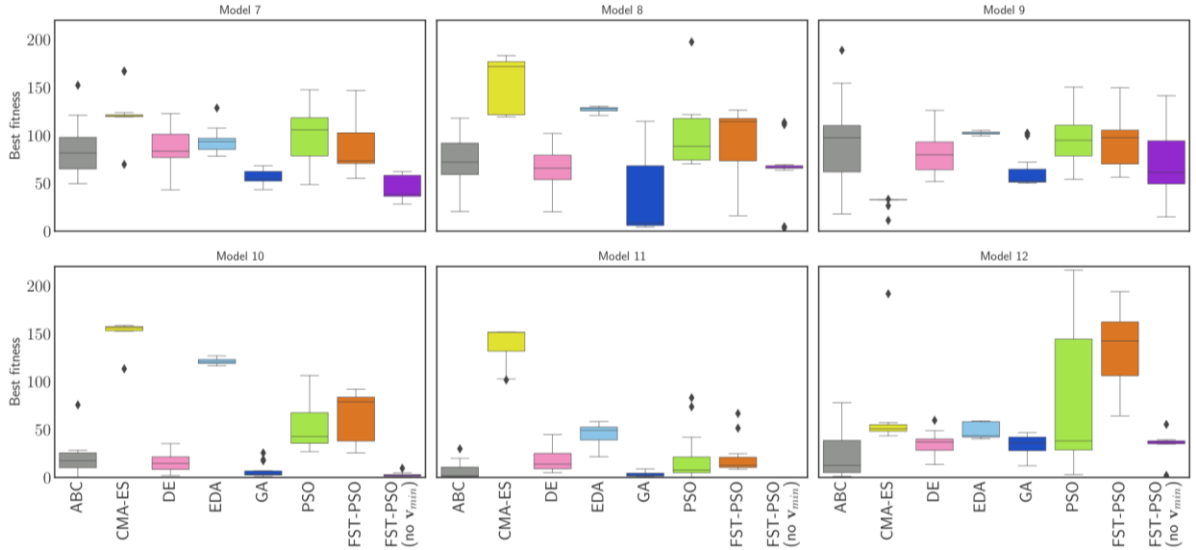
A more general (surprising) result

- We compared the performances of some state-of-the-art BICI methods (ABC, CMA-ES, DE, EDA, GA, PSO, and FST-PSO) on multiple **benchmark functions** (25 and 50 missing parameters)
- We also compared the performances of the same algorithms on multiple **biochemical models**
- Rankings are **very different** [Tangherloni *et al.*, Appl Soft Comp 2019]

Benchmark functions 50 parameters



Biochemical models 50 parameters



Results on biochemical PE problem

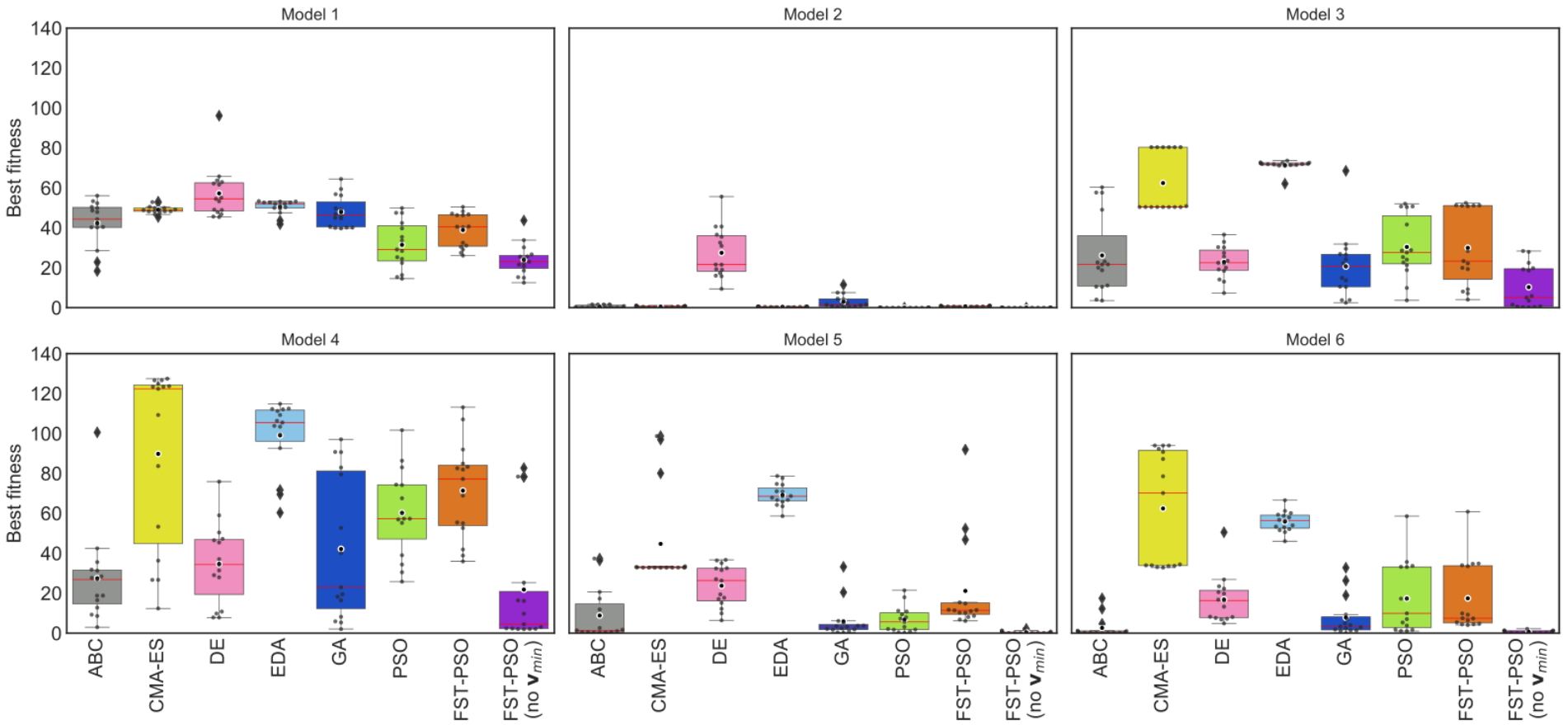


Fig. 1. The box-plots show the performances of the tested CI techniques applied to the PE of synthetic models characterized by 25 reactions and 25 chemical species. The red line denotes the median values, the black dots denote the average values, diamonds represent the outliers.

[Nobile *et al.,* IEEE WCCI 2018]

Results on biochemical PE problem

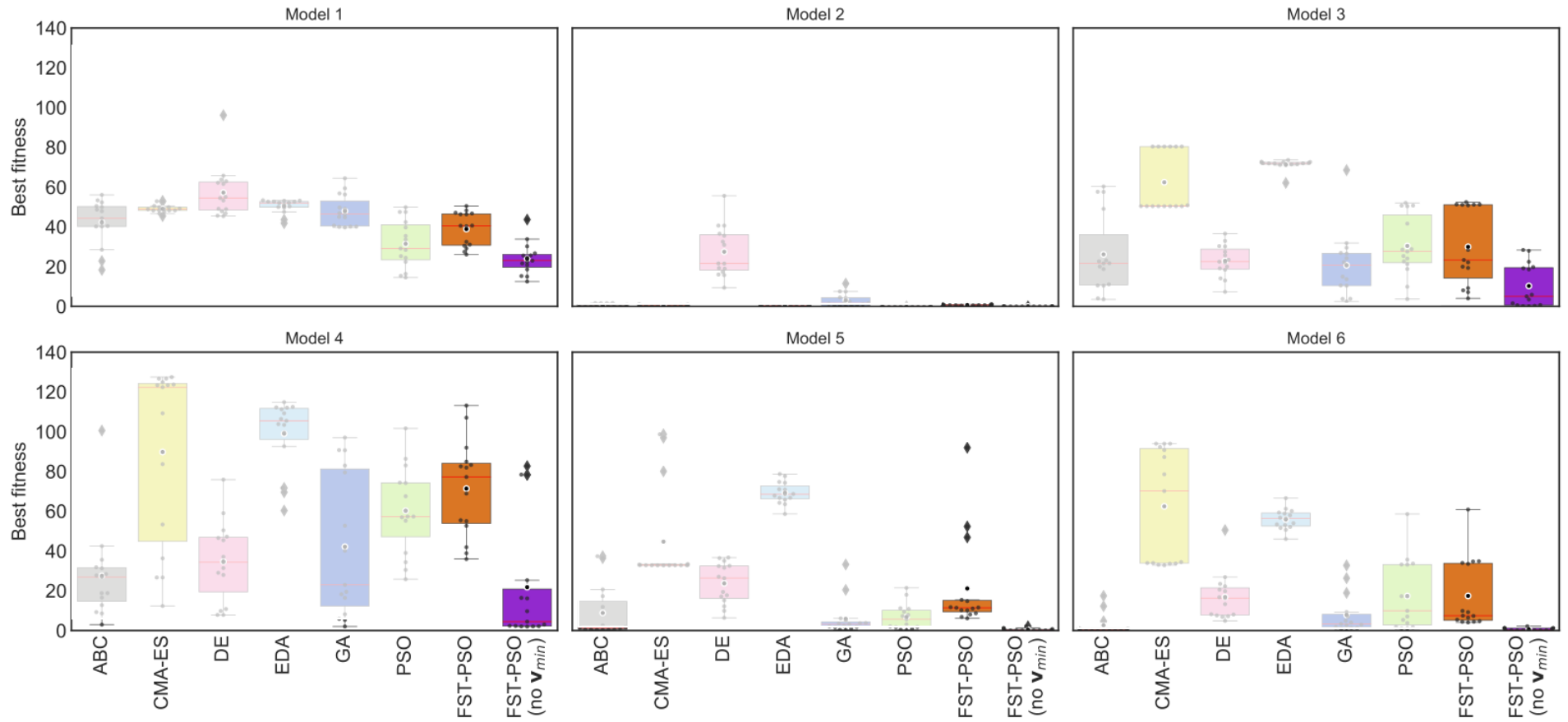
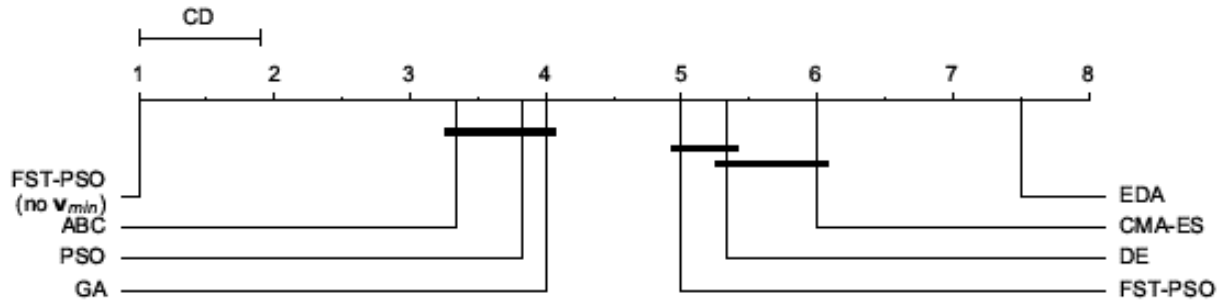


Fig. 1. The box-plots show the performances of the tested CI techniques applied to the PE of synthetic models characterized by 25 reactions and 25 chemical species. The red line denotes the median values, the black dots denote the average values, diamonds represent the outliers.

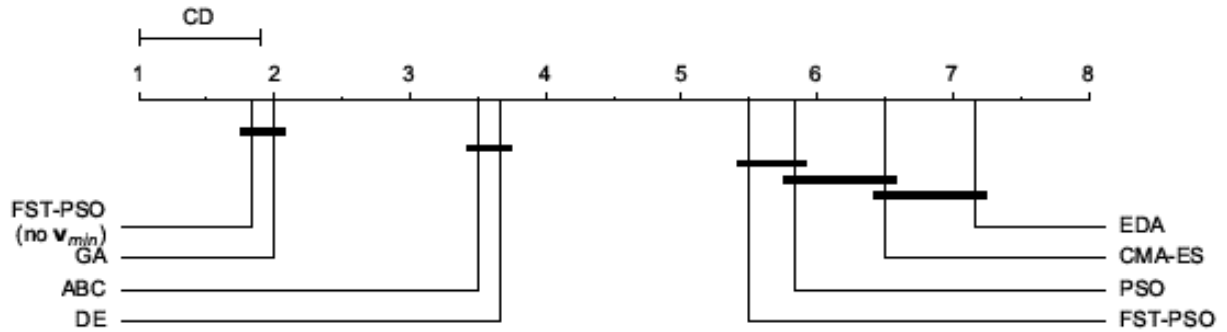
[Nobile *et al.,* IEEE WCCI 2018]

Bonferroni-Dunn's Critical Differences diagram

- **FST-PSO$_{no\ v_{min}}$ always ranked in the highest position**
  - Performances **tied to GA** at higher dimensions
  - ABC suitable alternative
  - EDA and (surprisingly!) **CMA-ES always ranked in the lowest positions**



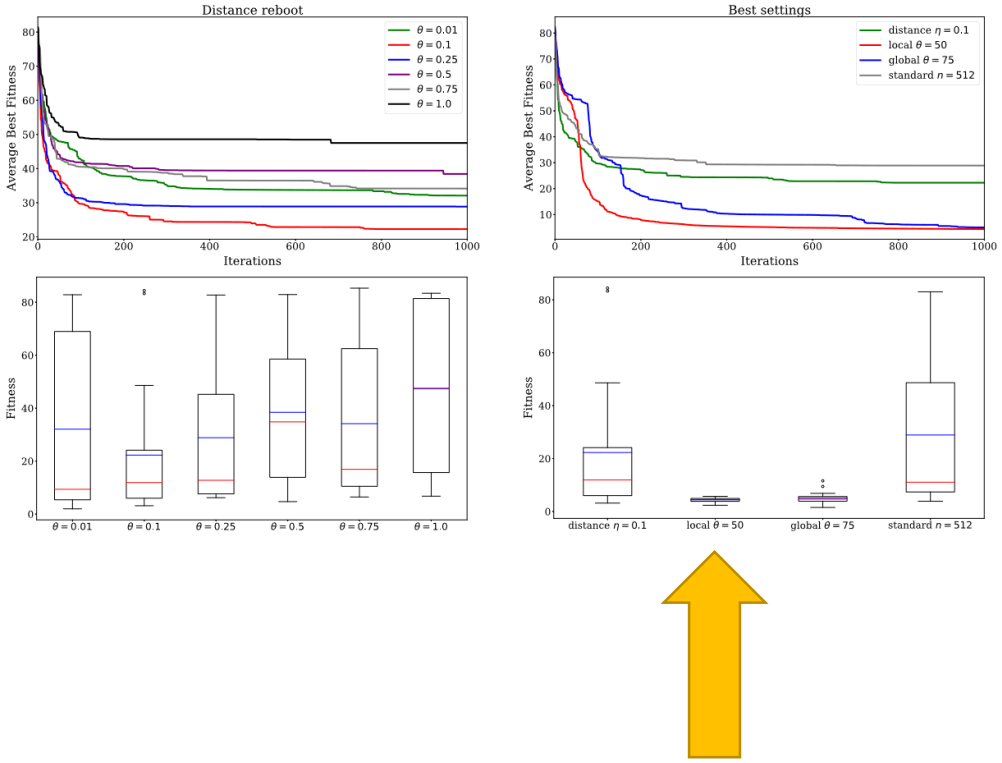
25 species / reactions



50 species / reactions

Reboots

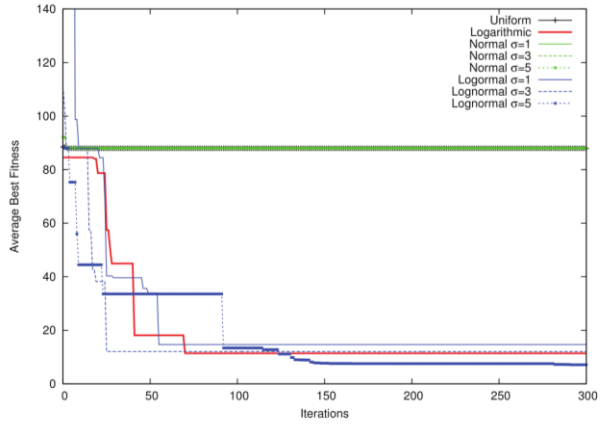
- After some iterations of the PSO, the particles can lose their momentum and begin **to «stagnate»**
  - The particles (or the whole swarm) can be **«rebooted»** in new positions
- We tested **three reboot methodologies** using a model of heat shock response in yeast (HSR) [Petre *et al.*, Nat Comput 2011]
  - **Distance** (reboot when the distance from the global best is below a threshold η)
  - **Local** (reboot if the personal best does not change for a fixed amount of iterations θ)
  - **Global** (reboot if the global best does not change for a fixed amount of iterations θ)

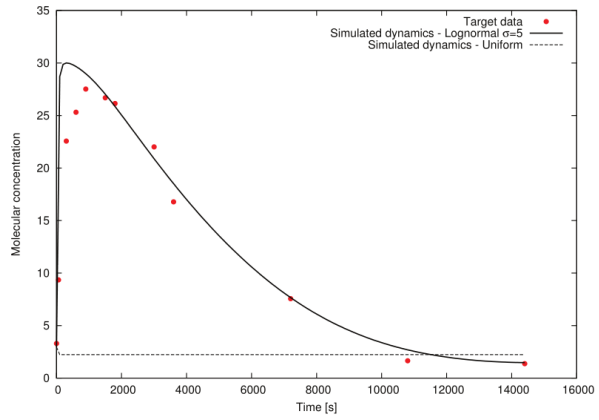


- According to our tests, «local» with $\theta = 50$ is the best strategy for PE
  - [Spolaor *et al.*, IEEE CIBCB 2017]

On population's initialization/update

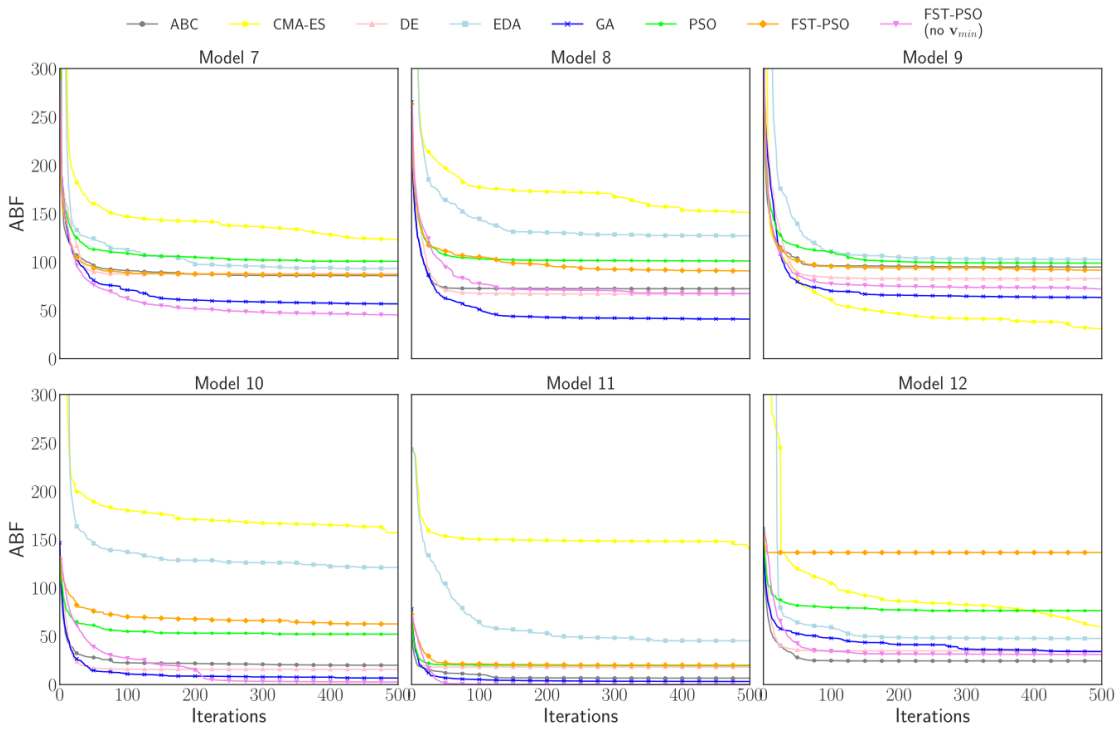
- The traditional way to create individuals in the search space is to sample the coordinates from a **uniform distribution**
  - **0-knowledge initialization**
  - Might affect the performances in some contexts, e.g., the PE
  - We compared the peformances of uniform, logarithmic, normal, and log-normal distributions on the HSR model
  - **Logarithmic and log-uniform initializations** yield better performances [Cazzaniga *et al*., IEEE CIBCB 2015]
  - Can be explained by the **log-uniform distribution** of kinetic parameters [Yue *et al*., Int J Chem Kinet 2008]





A change of semantics [1/2]

- We investigated the **bad performances** of state-of-the-art algorithms in the context of PE
  - We focused on the meta-heuristics based on **evolving probabilistic models**
  - Specifically, we tested **CMA-ES** and **EDA**
  - We speculated that, due to the peculiar **log-uniform** distribution of missing parameters, the **normal distributions fail** in generating useful candidate solutions



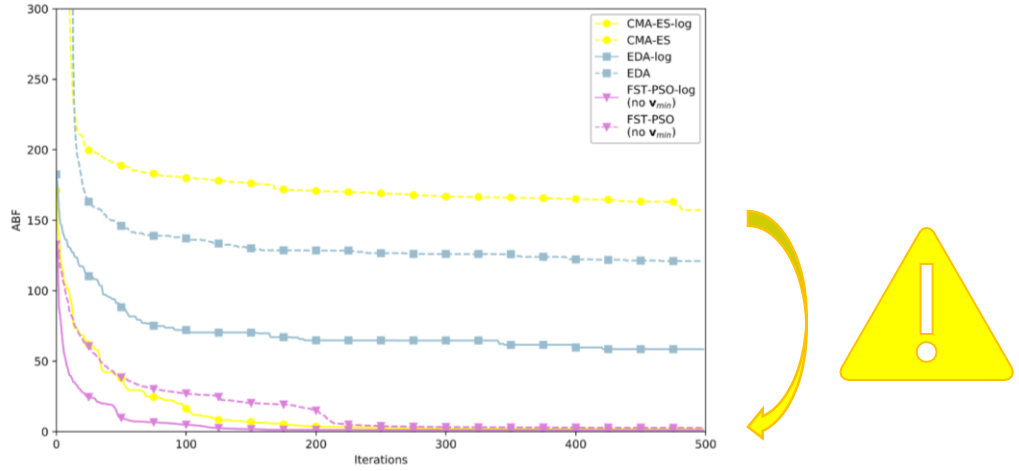
A change of semantics [2/2]

- To verify our hypothesis we tried to **change the semantics** of parameters using the transformation

$$\phi_m = \log_{10}(\beta_m^{max}) - \left(\log_{10}\beta_m^{min} + \log_{10}\beta_m^{max}\right)k_m$$

$$k'_m = 10^{\phi_m}$$

- According to our results, CMA-ES becomes competitive, but **all optimization algorithms can benefit** from this transformation [Tangherloni *et al.*, Appl Soft Comp 2019]

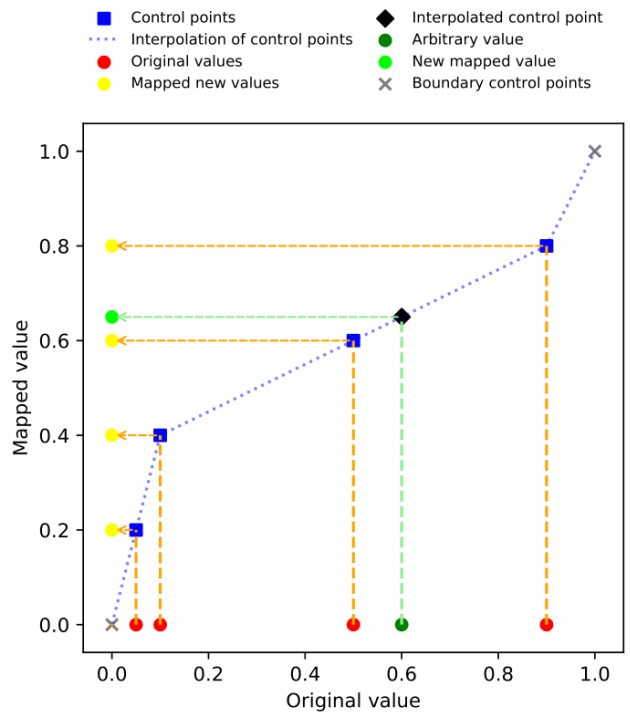


Dilation functions

- The **semantic transformation** used in the previous example basically **«dilates»** the lowest orders of magnitude

- We extended this idea with **«dilation functions»** (DF) able to «expand» an «compress» the search space

- We implemented DFs with **«control points»** (CP)
  - CPs determine a new mapping of parameters $d: [0,1] \rightarrow [0,1]$
  - Interpolation provides values for *any* parameters to be mapped
  - By means of shift/scale we can use DF to re-map the parameters of *any* optimization problem



Impact of a DF to the Ackley benchmark function

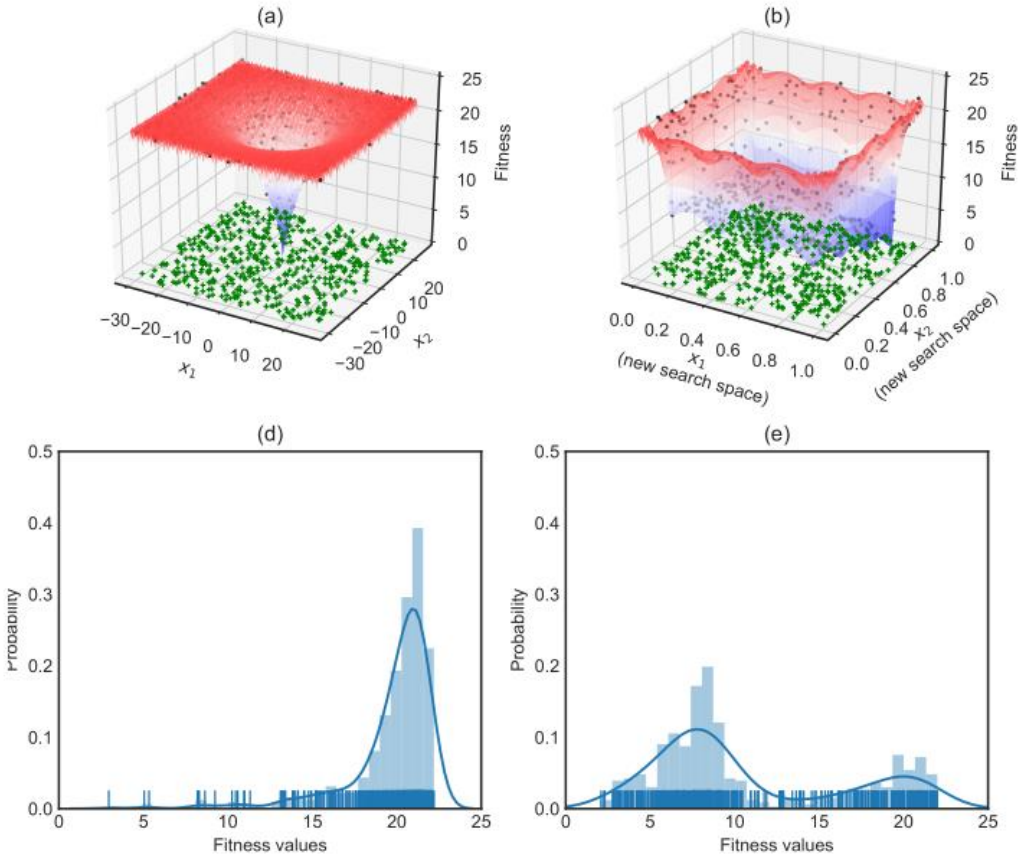
Original function

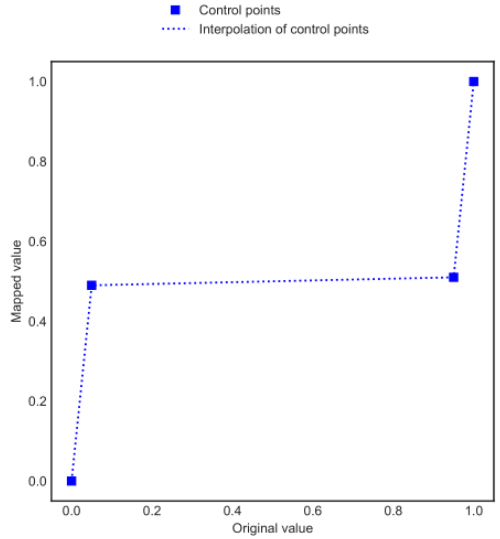
Dilated function

Solutions remapped on the original function

Ackley function

$$f(\mathbf{x}) = -a \exp\left(-b\sqrt{\frac{1}{d}\sum_{i=1}^{d} x_i^2}\right) - \exp\left(\frac{1}{d}\sum_{i=1}^{d}\cos(cx_i)\right) + a + \exp(1)$$

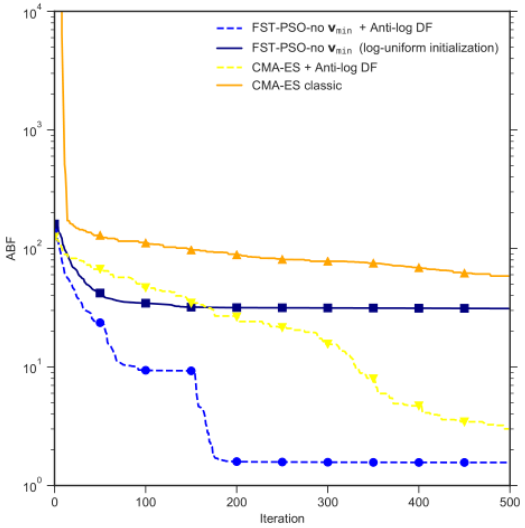




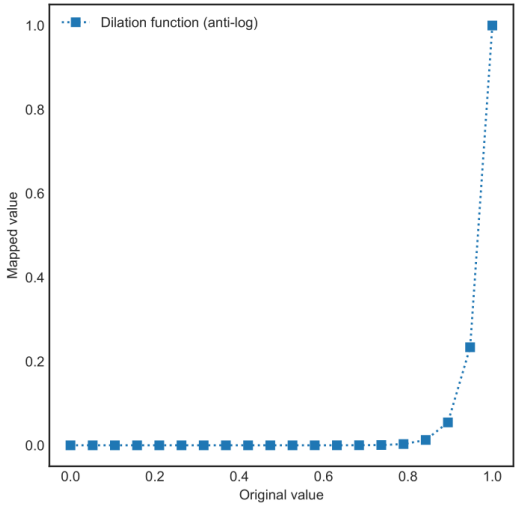
[Nobile *et al.*, IEEE CEC 2019]

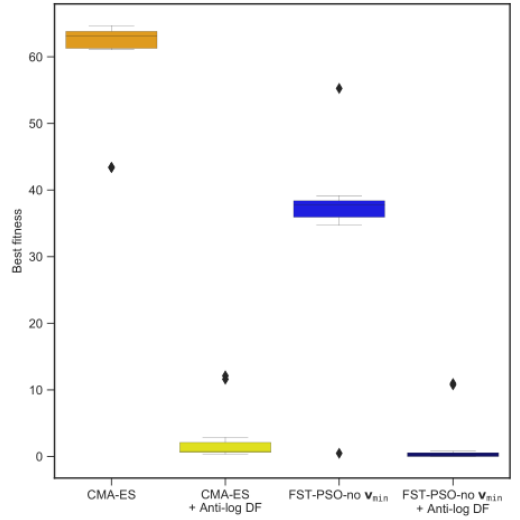
DFs, PE, and FST-PSO

- The Ackley function is well characterized, so that we could craft a DF tailored on that benchmark problem
  - This is clearly **not doable in real-world scenarios BUT…**
  - …we can still leverage some **general information** about the **fitness landscape**
  - E.g., in the case of PE, we know that parameters have **log-uniform** distribution



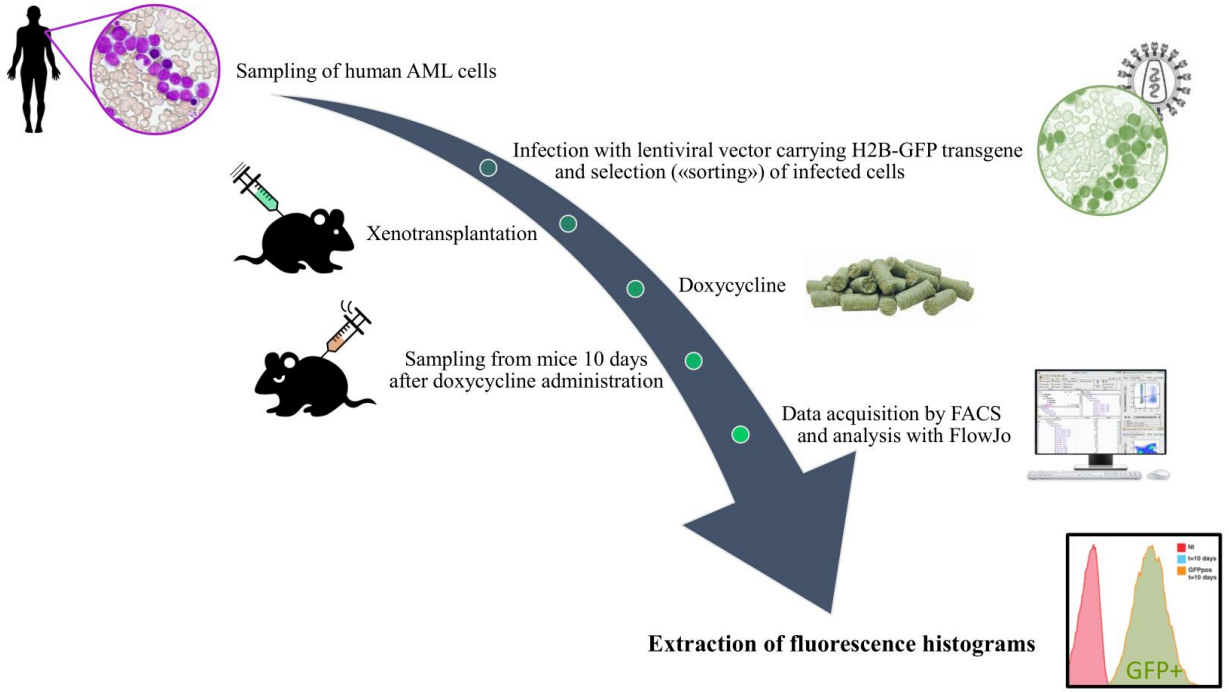
- We created and tested an «anti-log» DF that «expands» the **lowest orders of magnitude**
  - **Much better performances** than common PE
  - In particular, with CMA-ES
  - [Nobile *et al.*, IEEE CEC 2019]





Applied FST-PSO: investigating Acute Myeloid Leukemia

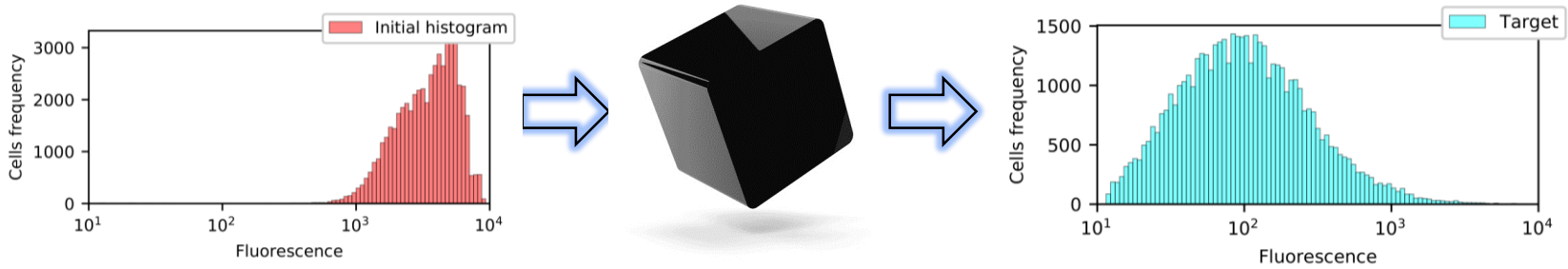
- Cooperation with European Institute of Oncology (IEO, Milan, Italy)
  - Investigation of **relapse** in **Acute Myeloid Leukemia** (AML)
  - Human AML cells **xenotransplanted in mice** (NOD.Cg-*Prkdc*^{scid}*Il2rg*^{tm1Wjl}/SzJ strain)
  - AML cells are marked with **green fluorescent proteins** by means of lentiviruses
  - **Flow cytometry** used to collect **fluorescence histograms** (at $t = 0$, $t = 10$, $t = 21$ days)



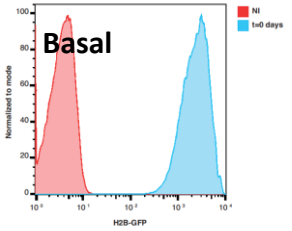
Experimental protocol figure taken from
[Nobile *et al.*, IEEE CIBCB 2019]

Investigating Acute Myeloid Leukemia

- Given the experimental fluorescence distributions at $t = 0$ and $t = 10$ days what can we say about **AML proliferation**?



- Some assumptions
  - GFP binds to H2B histon: **fluorescence level halves** at each **cell division**
  - Mice are **immunocompromised**: no immune response from lymphocites, NK cells, etc.
  - **Basal fluorescence level**: if fluorescence goes below a threshold it becomes «GFP-negative»



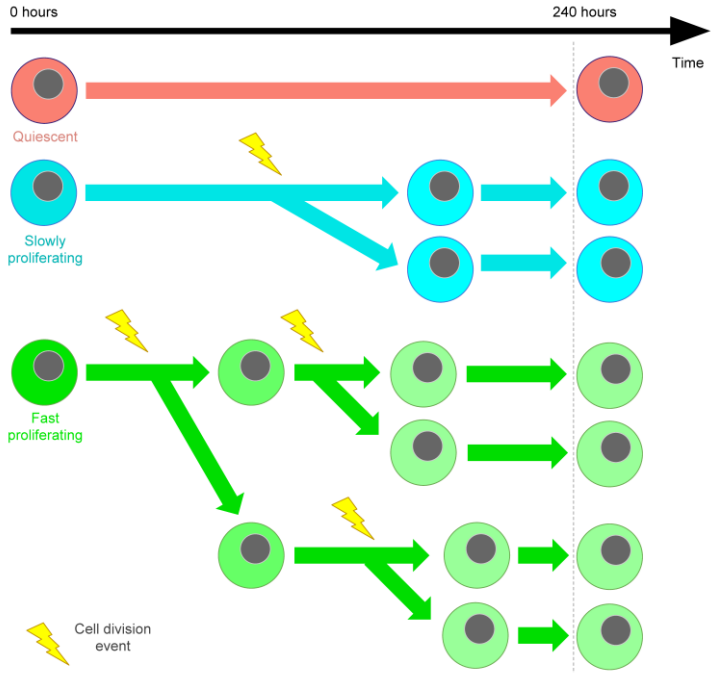
- We created a **novel modeling/simulation framework** to study cell proliferation: **ProCell**

ProCell

- **Stochastic** modeling and simulation of **cell proliferation**

- A simulation requires the following information
  - **Number of sub-populations** and their **ratio**
  - **Mean and standard deviation** of the cell division interval for each sub-population

- By specifying the **GFP-positivity threshold** and the **simulation time** (in hours) ProCell can perform a stochastic simulation
  - A **stack of cells** is maintained: new cells, generated by (random) division events, are dynamically added to the stack
  - **GFP-negative** cells are **removed** from the stack
  - The algorithm returns the stack of fluorescent cells at $t = t_{\max}$



FST-PSO and ProCell

- The output histograms produced by ProCell depend of the **parameterization** of the model
  - i.e., sub-populations, proportions, mean/sd of division intervals
  - These values are **impossible to measure** with experimental methods
  - A robust **PE methodology** is necessary

- ProCell was integrated with FST-PSO to develop a **fully automatic PE system**

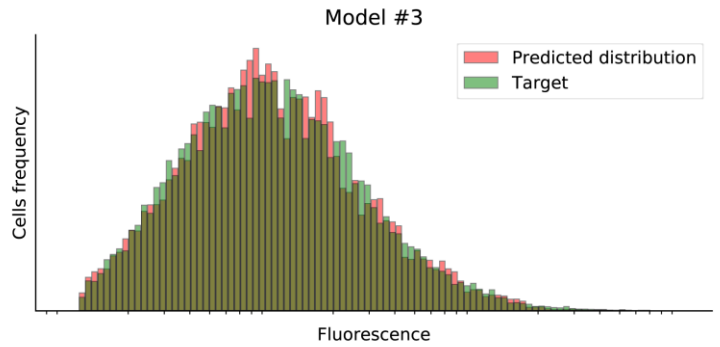
- Objective function: **Hellinger distance** between the normalized rebinned simulated histogram $\widehat{H}_j^{\boldsymbol{\theta}}$ created with parameterization $\boldsymbol{\theta}$ and the target experimental histograms $\hat{S}_j$:

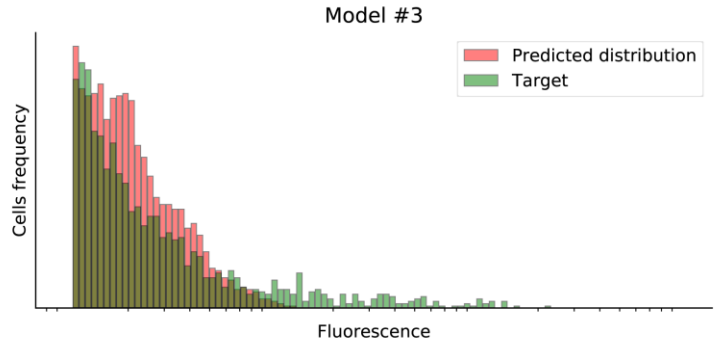
$$f(\boldsymbol{\theta}) = \frac{1}{\sqrt{2}} \sqrt{\sum_{j=1}^{N} \left(\sqrt{\widehat{H}_j^{\boldsymbol{\theta}}} - \sqrt{\hat{S}_j} \right)^2}$$

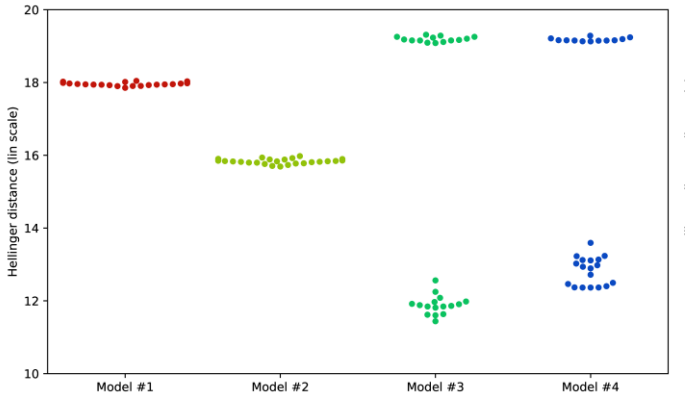
FST-PSO and ProCell

- We tested four competing models
  - Model #1: simple proliferation
  - Model #2: proliferating cells + quiescent cells
  - Model #3: slowly proliferating + fast proliferating + quiescent cells
  - Model #4: slowly proliferating + fast proliferating

- Our results suggest that **Model #3 can explain the observed experimental fluorescence data**
  - Lowest Hellinger distance among the tested models
  - The validation at time $t = 21$ days also shows a good fit with the experimental fluorescence data
  - [Nobile *et al.*, Bioinformatics 2019]
  - [Nobile *et al.*, IEEE CIBCB 2019]
  - [Nobile *et al.*, PLOS Comp Biol 2019 (in preparation)]







Conclusion

- Systems Biology allows the modeling and computational analysis of complex biological systems
- Accurate parameterization is mandatory to perform faithful simulations and lead to new insights of the biological system under investigation
- The PE can be efficiently performed using population-based global optimization techniques
- The performances of optimization algorithms for the PE problem can be improved by using some general (e.g., reboots) and domain-specific (e.g., advanced fitness functions, log-uniform initialization) strategies
- The naïve representation of candidate solutions is not efficient, especially in the case of algorithms based on generative probabilistic models
- Semantic transformations based on dilation functions can be used to «distort» the fitness landscape and improve the performances