Optimization

Dario Pescini[1]

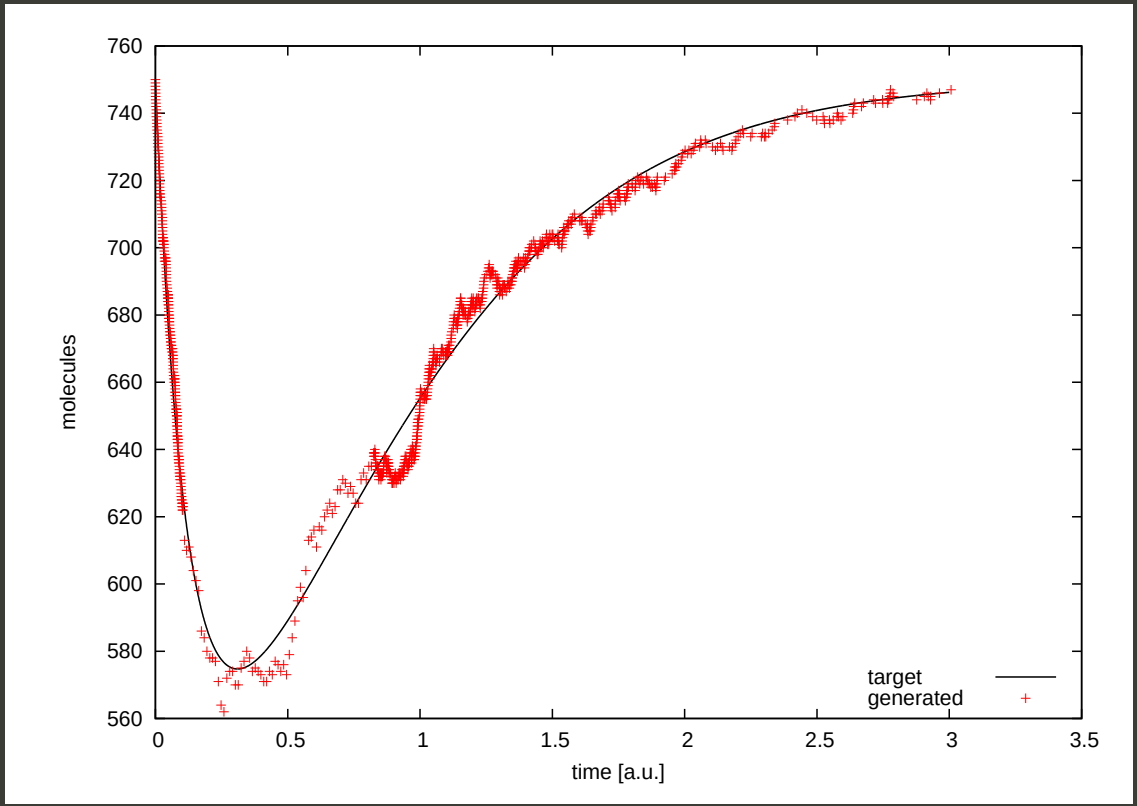
Università degli Studi di Milano-Bicocca

Dipartimento di Statistica e Metodi Quantitativi

dario.pescini@unimib.it

The Goal

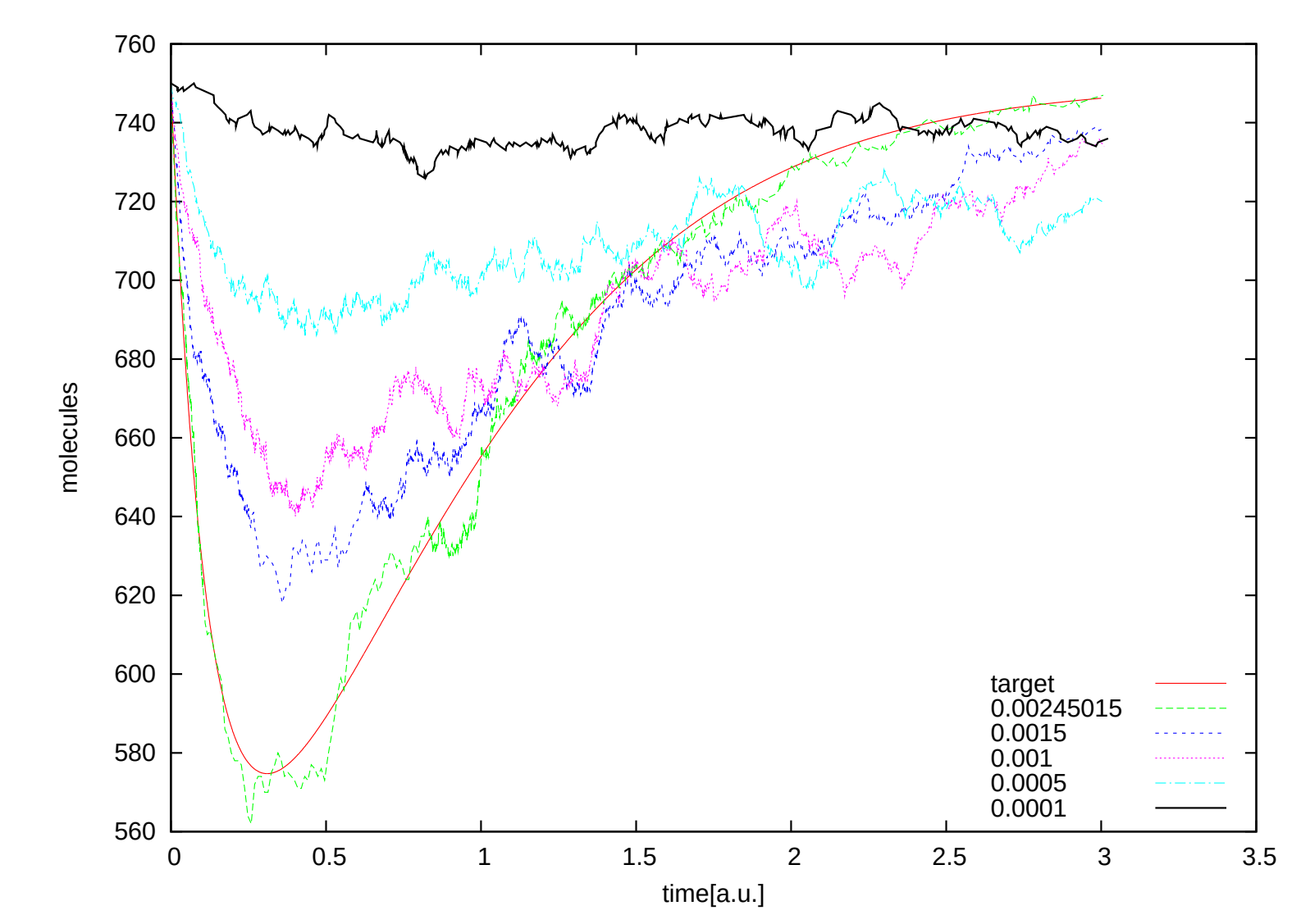
We would like to reproduce the target dynamics via stochastic simulations:



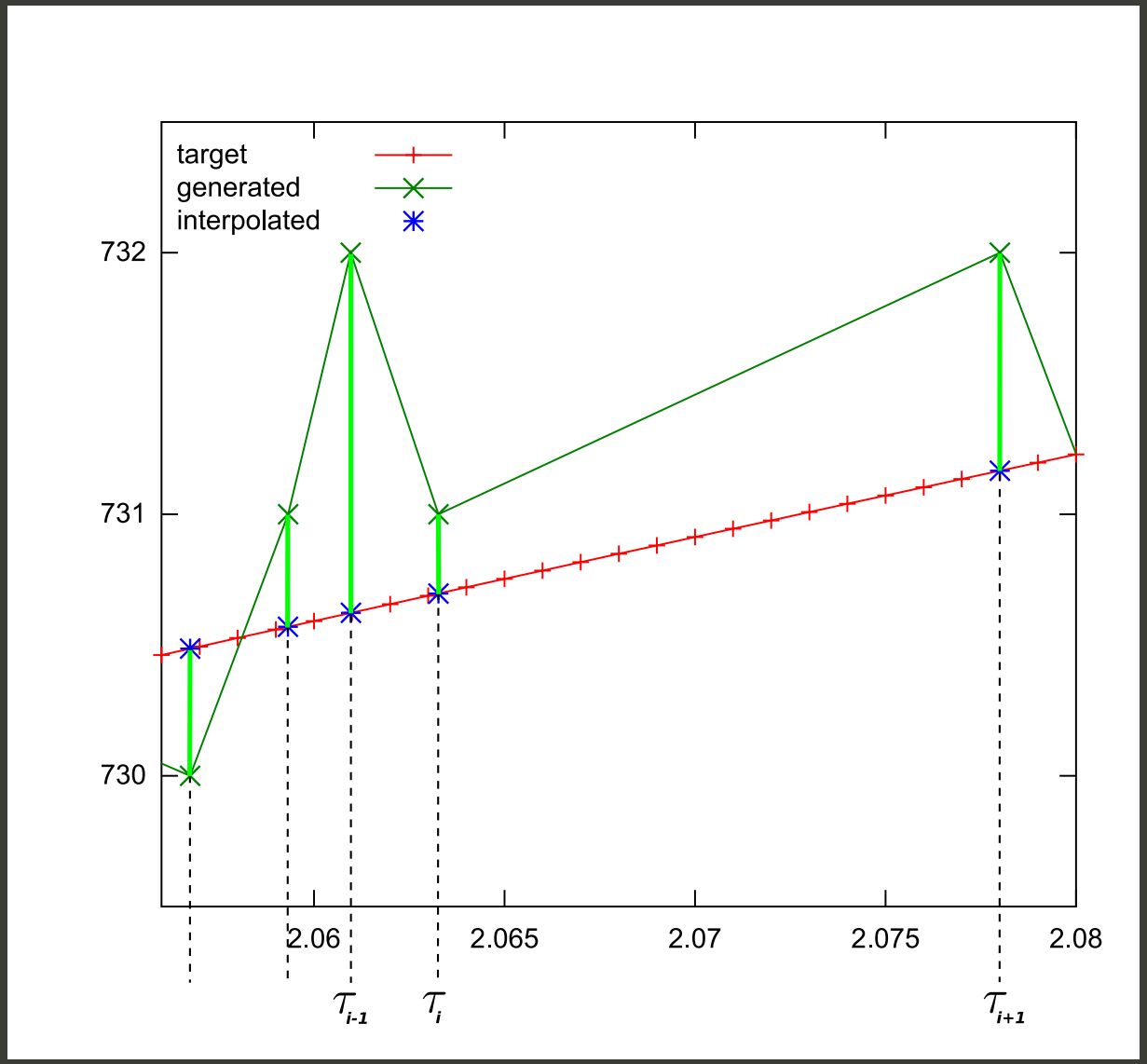
not necessarily using the same set of constants.

The Problem

Quantify the “distance” among the dynamics and find the closest one



Fitness: Standard Distance

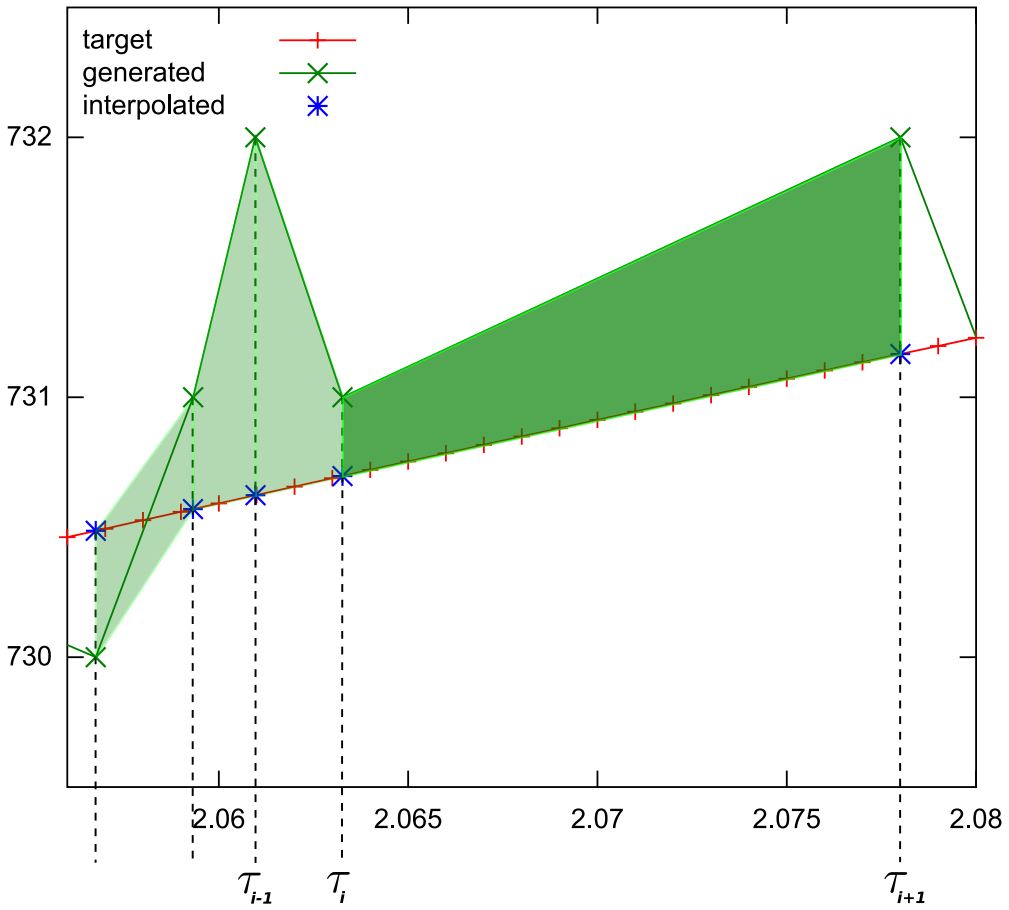


Fitness: Tricks

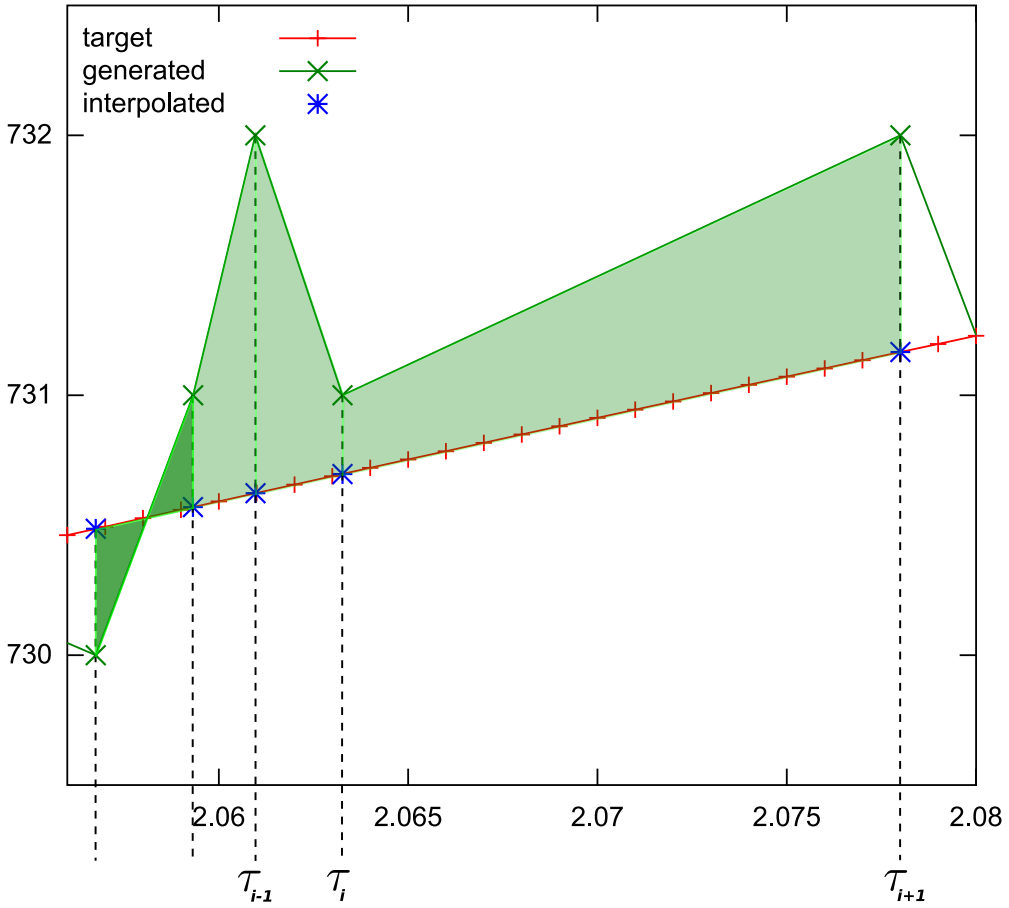
Facts related to intrinsic noise that should not be neglected:

- each outcome j is quantitatively different $\{X(\tau_i)\}_j$
  - not evenly sampled $\{\tau_i\}_j$
  - not same number of points
  - $\{\neq X(\tau_i)\}_j$
- in the thermodynamic limit $\{X(\tau_i)\} \to [X](t_i)$
  - exploit ensemble behavior (may flatten oscillations, not in phase outcome)
  - $\langle\, F\,(X(\tau_i))\,\rangle \stackrel{?}{=} F\,(\,\langle X(\tau_i)\rangle\,)$
- same parameters, possibly (almost always), generate different values of the fitness function ("weak convergence")

Fitness: “Area” Distance



Fitness: “Area” Distance (refinement)



Combinatorial Optimization Problems

- To solve a problem of combinatorial optimization means to find the "best solution" or "optimum" within a given set of alternatives solutions
- It is mandatory to measure quantitatively the "quality" of each solution so that a comparison is possible.

Hill Climbing

Hill climbing (or local search) is a technique to solve combinatorial optimization problems.

It consists of a stepwise improvement of the fitness function exploiting the concept of neighborhood.

For each solution i belonging to the solution space S, it computes (if possible) a subset of solutions $N \subseteq S$ "neighbor" of i, then:

- it chooses the solution $j \in N$ with the "best" fitness in N
- it computes the neighborhood of j for the next iteration
- it halts when it finds neighborhood of a solution k in which there is no better fitness than the one of k

Hill Climbing: pseudo-code

In the case of a minimization problem, the pseudo-code is:

```
Procedure LOCAL SEARCH
begin
  initialization (i_start)
  i := i_start
  repeat
    generate j from S_i
    if f(j) ≤ f(i) then i := j
  until f(j) ≥ f(i), ∀j ∈ S_j
end
```

Hill Climbing: Comments

- By definition, hill climbing halts returning a local optimum
- The “quality” of the result is influenced by the chosenstarting solution
- The algorithm returns a global optimum only if the neighborhood is extended to the whole solution space (unfeasible in most of application cases)

Hill Climbing: Comments

Pros:

- Ease of use concerning the implementation and the application
- Flexibility in the problem representation and in the choice of the initial conditions

Cons:

- The result is a local optimum

To avoid the drawbacks it is possible to:

- Repeat the procedure for a huge number of distinct initial conditions
- Define a complex neighborhood structure (if possible)
- Accept (probabilistically) no better fitness solutions (Simulated Annealing)

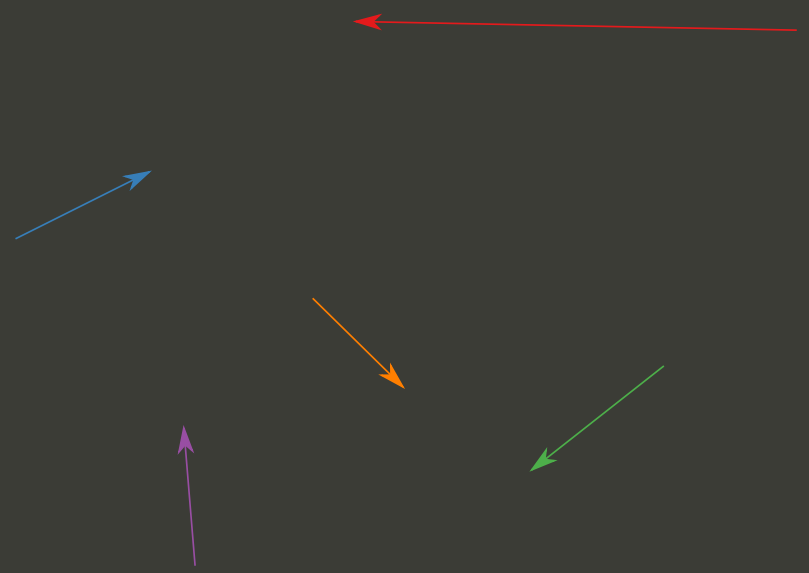
Particle Swarm Optimizers

- The particle swarm algorithm has been shown to optimize hard mathematical problems in continuous or binary space (Kennedy and Eberhart, 1995; Kennedy and Eberhart, 1997)

- Particles, defined as multidimensional points in parameter space
- adjust their trajectories toward so far best positions
  - of their own
  - found by any member of a topological neighborhood

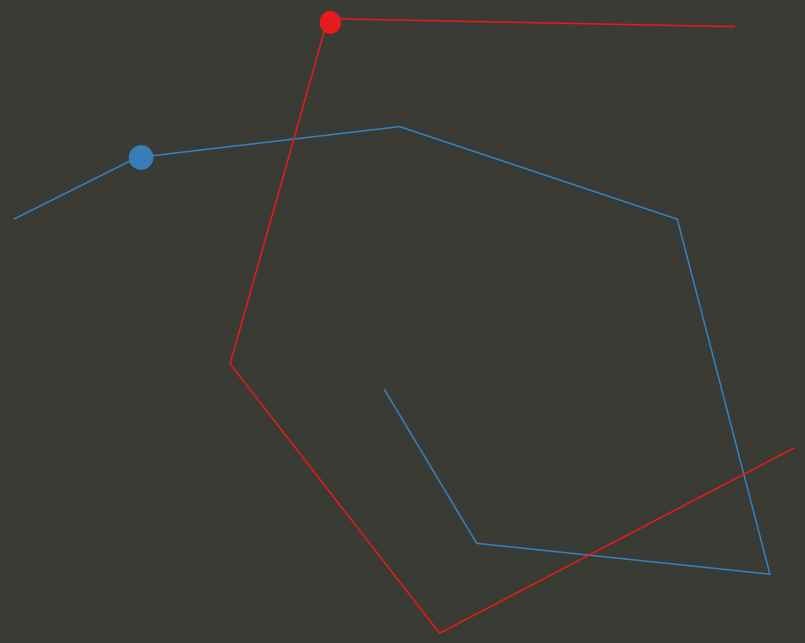
PSO

- A population of particles is initialized with random positions $\overrightarrow{x_{id}}$ and "velocities" $\overrightarrow{v_{id}}$. At each time-step, each particle is tested in an evaluation function (the fitness)



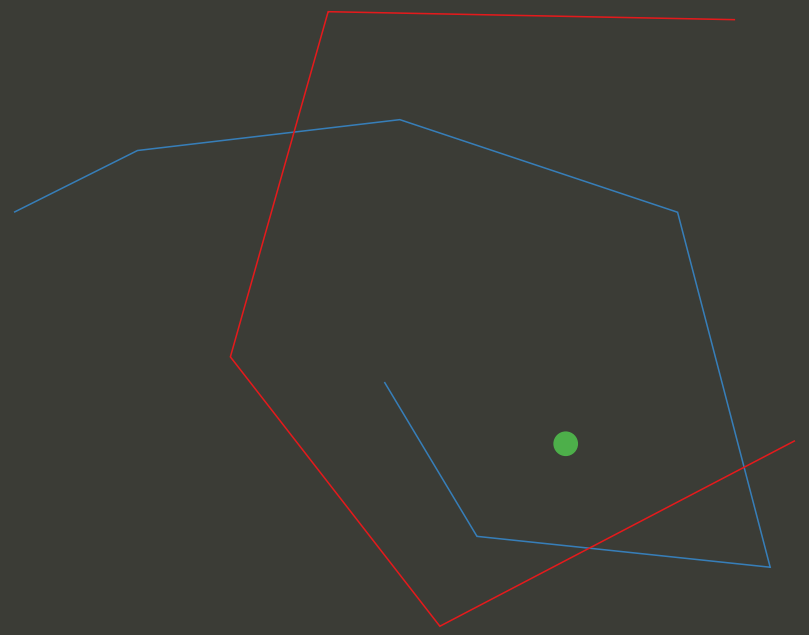
PSO

- A population of particles is initialized with random positions $\overrightarrow{x_{id}}$ and "velocities" $\overrightarrow{v_{id}}$. At each time-step, each particle is tested in an evaluation function (the fitness)
- The best position found so far by a particle (individual best) $\overrightarrow{b_{id}}$ is stored.



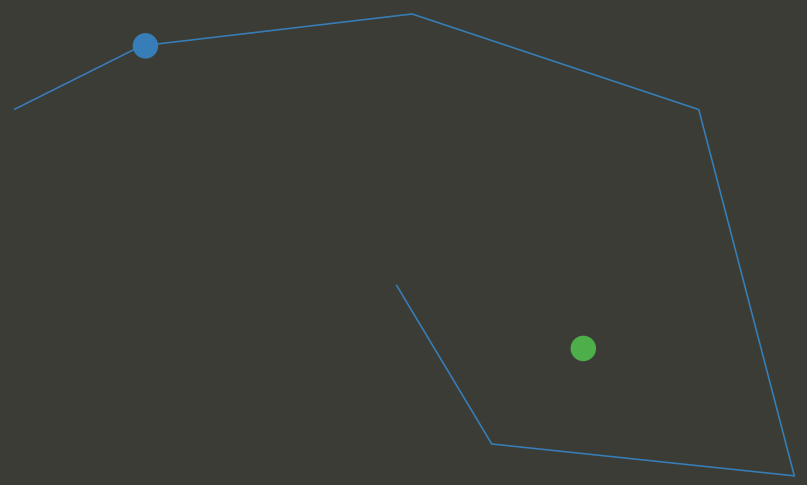
PSO

- A population of particles is initialized with random positions $\overrightarrow{x_{id}}$ and "velocities" $\overrightarrow{v_{id}}$. At each time-step, each particle is tested in an evaluation function (the fitness)
- The best position found so far by a particle (individual best) $\overrightarrow{b_{id}}$ is stored.
- As each particle is evaluated, the best position found so far by the swarm (global best) $\overrightarrow{b_g}$ is stored.



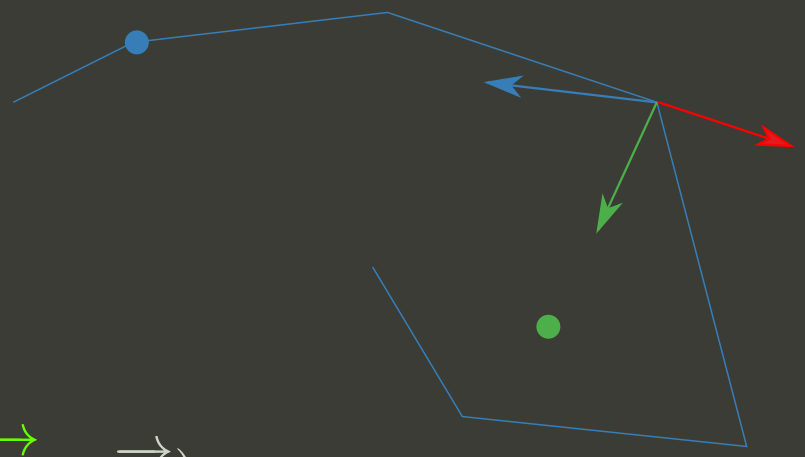
PSO

- A population of particles is initialized with random positions $\overrightarrow{x_{id}}$ and "velocities" $\overrightarrow{v_{id}}$. At each time-step, each particle is tested in an evaluation function (the fitness)
- The best position found so far by a particle (individual best) $\overrightarrow{b_{id}}$ is stored.
- As each particle is evaluated, the best position found so far by the swarm (global best) $\overrightarrow{b_g}$ is stored.



PSO

$$\overrightarrow{x_{id}}(t) = \overrightarrow{x_{id}}(t-1) + \overrightarrow{v_{id}}$$



$$\overrightarrow{v_{id}} = w\,\overrightarrow{v_{id}} + c_1\,r_1(\overrightarrow{b_{id}} - \overrightarrow{x_{id}}) + c_2\,r_2(\overrightarrow{b_g} - \overrightarrow{x_{id}})$$

- w inertia weight
- c_1 weight of the individual inheritance
- c_2 weight of the social inheritance
- r_1, r_2 random numbers in $[0, 1]$

PSO: pseudo-code

```
Procedure PSO
begin
  for i = 1 to population_size
    for d = 1 to dimension
      if eval(i) < best_i then p_id = x_id
      v_id = w v_id + c_1 r_1 (b_id - x_id) + c_2 r_2 (b_gd - x_id)
      if v_id > V_max then v_id = V_max
      else if v_id < -V_max then v_id = -V_max
      x_id = x_id + v_id
end
```

Genetic Algorithms: Darwinian Theory of Evolution (1)

According to Darwin's theory, natural selection induces the evolution of the species

In this theory, he points out some fundamental aspects that allow the evolution:

- Reproduction
  Two individuals can generate an offspring.
- Environmental Adaptation
  It is the ability of an individual to adapt to the living environment, increasing its chances to survive and to reproduce.

Genetic Algorithms: Darwinian Theory of Evolution (2)

- Inheritance
  Some features of the parents are passed to the offspring
- Variation Some features of the offspring can differ from those of the parents, possibly increasing the adaptation
- Competition Due to the limited resources there exists a competition among the individuals to survive

Genetic Algorithms

The starting point is a population of individuals (subset of the solution space) and a fitness function to assess the quality of each individual.

Iteratively, the “best” individuals are selected at each generation, and variation techniques are applied to them.

The aim is to get a better adapted set of individuals to the next generation.

Genetic Algorithms

Genetic algorithms accomplish two fundamental steps:

- Selection: the process of selecting the fittest individuals

  Selection mimics the Darwinian principles of survival and competition.

- Variation: mainly realized by means of two operators
  - Sexual reproduction (Crossover)
  - Mutation

  Variation operators mimic the Darwinian principles of reproduction, inheritance and variation

Selection Operators

Selection operators are not necessarily applied to the whole population. They are applied only to those individuals with higher survival chances.

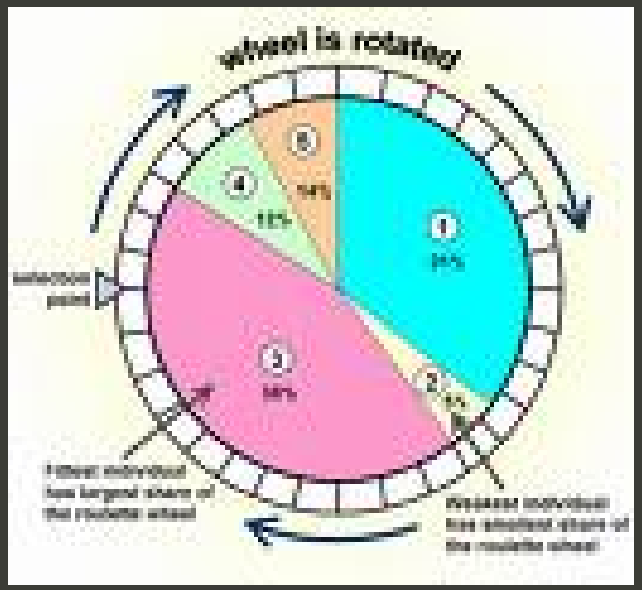
The most widely adopted selection operators are:

- roulette wheel
- ranking
- tournament

Selection Operators: Roulette Wheel

The probability of one individual of being selected is directly proportional to its fitness.

$$p_i = \frac{f_i}{\sum_{j=0}^{N-1} f_j}$$



Selection Operators: Ranking

The individuals are sorted according to their fitnesses values.
The probability of one individual of being selected is directly proportional to its ranking.

Note:
The most common functions to assign probabilities to individuals are either linear or exponential.

Selection Operators: Tournament

Features:

- With this operator it is possible not to evaluate the fitness values of each individual in the population
- Selection takes place randomly picking up k individuals, putting them in a race and choosing the one with the best fitness value
- k is the tournament size. It rules the selection pressure, weighting how valuable is to have better fitnesses w.r.t. the other individuals.

Selection Operators: Notes

These operators share some common features:

- they allow the individuals with higher fitness to be selected with higher probability (high fitness corresponds to greater chances to survive)
- each individual within the population has a probability greater than 0 to be selected (every individual can survive)
- a random process is involved in the selection (better adapted and “lucky” individuals survive)

Variation Operators: Crossover

Crossover: it is the technique that allows to exchange and/or mix features among the population individuals (usually 2). The intent is to generate an offspring that differs from parents but combines their features.

The crossover is a conservation operator meaning that the new individual mixes already present features only.

Variation Operators: Crossover

The crossover acts in the following way:

- 2 individuals (parents) are chosen by means of a selection technique
- it chooses the cutting point, exchange position among the parents
- the obtained substrings are combined in order to obtain 2 new individuals

0110|11001 $\longrightarrow$ **0110**|01010

1100|01010 $\longrightarrow$ 1100|**11001**

Variation Operators: Mutation

Mutation: it is the technique that allows to modify part of the structure of an individual to generate a new one for the next generation

The mutation is an innovation operator meaning that the created offspring might contain features that were not present in the population, thus introducing new genetic material.

Variation Operators: Mutation

The mutation acts in the following way:

- an individual is chosen by means of a selection technique
- it chooses the mutation position, usually one feature.
- the chosen feature is substituted with another among the allowed ones.

$$0100\mathbf{1}1011 \longrightarrow 0100\mathbf{0}1011$$

Genetic Algorithms: pseudo-code

Procedure **Genetic Algorithm**:

1. Generate the initial population of N individuals
2. Repeat until the halting criterion is satisfied
   - a Compute the individuals fitnesses
   - b Repeat until the new population contains N individuals
     - b.1 Choose a genetic operator
     - b.2 if the operator is crossover select 2 individuals else (mutation) select one individual
     - b.3 Apply the operator
     - b.4 Insert the offspring in the population
   - c Substitute the old population with the new one.
3. Return the individual(s) with the best fitness

Genetic Algorithms: Notes

The GA parameters choice are problem dependent. Artcraft work.

- The structure of an individual is said to be the genotype
- The fitness of an individual is said to be phenotype

In the GA the selection acts on the phenotype while the variation acts on the genotype.

Basic Idea

Mimicking Darwinian evolution to find the set of optimal solutions.

- reproduction
  - ereditariety
  - variation
- adaptation
- competition

Our PSO Implementation

$$v_{id} = w\, v_{id} + c_1 r_1 (b_{id} - x_{id}) + c_2 r_2 (b_{gd} - x_{id})$$

PSO1

- decreasing w inertia weight plus a Gaussian noise
- c_1, c_2 are independents

PSO2

- decreasing w inertia weight plus a Gaussian noise
- c_1, c_2 co-evolve, modified with a Gaussian noise

Our GA Implementation

- randomly generate initial population
- $\forall$ generation:
  - $\forall$ individual: evaluate the fitness function
  - apply elitism: prevents best fitness individuals to be modified
  - apply tournament strategy to select the remaining offspring
    - randomly pick up k individuals and choose the best two
    - apply crossover to generate the new individual
  - apply mutation according to a small probability value

GA operators { selection, variation }

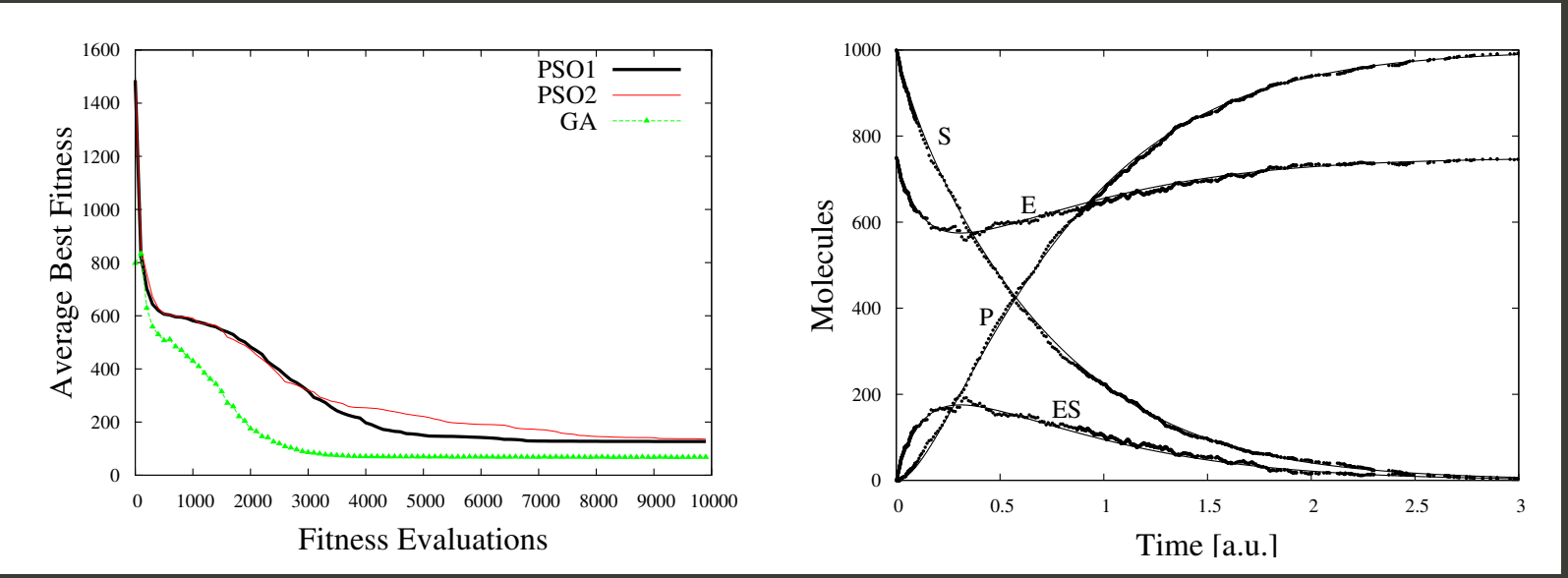
Its Operators

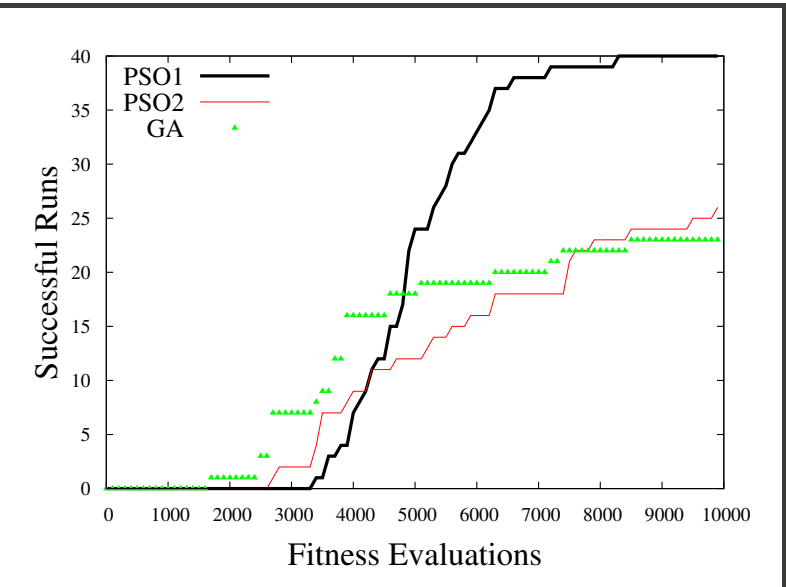
- One point average crossover: returns one offspring that contains at each position the average values of the parents chromosomes
- Elitism: reduced to one element to preserve the fittest
- Gaussian mutation: perturbs an allele with a number drawn from a normal distribution with mean equal to the current value and σ fixed
- Range mutation: increments or decrements an allele of a prefixed quantity (that is a fraction of the size of the admissible range)
- Reinitialization: changes the value of an allele with an uniformly distributed random number in the admissible range

Hints

- Elitism ≈ 0.01 of the population
- tournament $\longrightarrow$ selection pressure
- crossover
  - $P_c = 0.95$
  - gene average
- mutation
  - $P_m \in [0.05, 0.5]$ unusually high
  - uniform or Gaussian

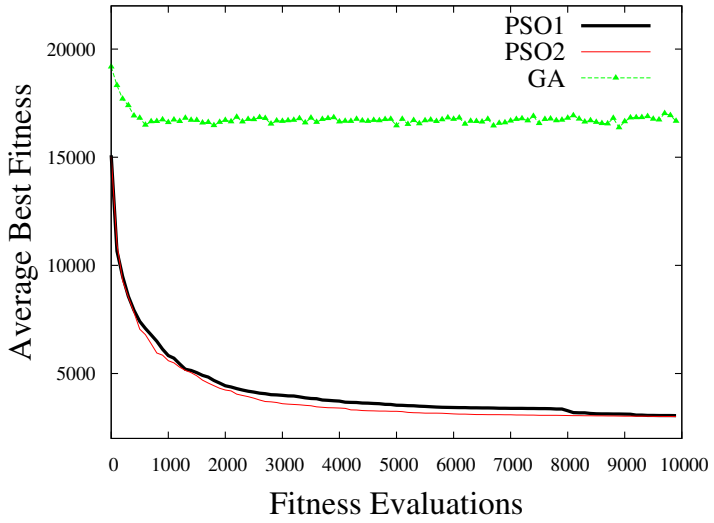
Comparison GA & PSO: Michaelis Menten

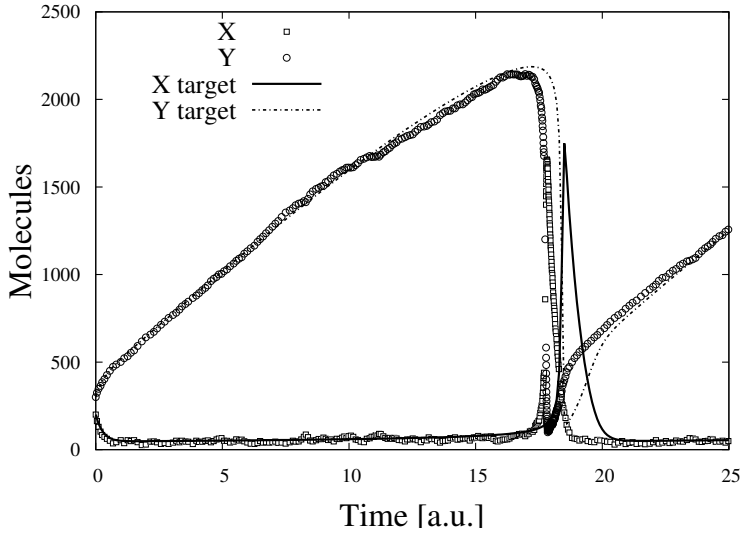


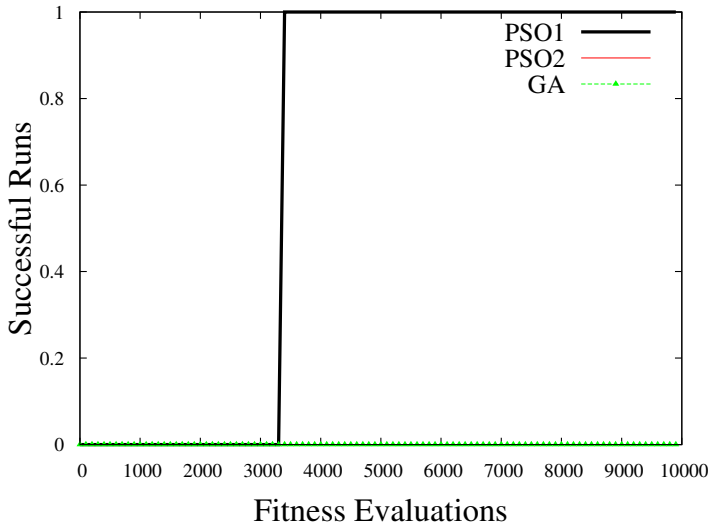


- PSO1 best performer
- similar average fitnesses
- dynamics faithfully reconstructed

Comparison GA & PSO: Oscillating Brusselator

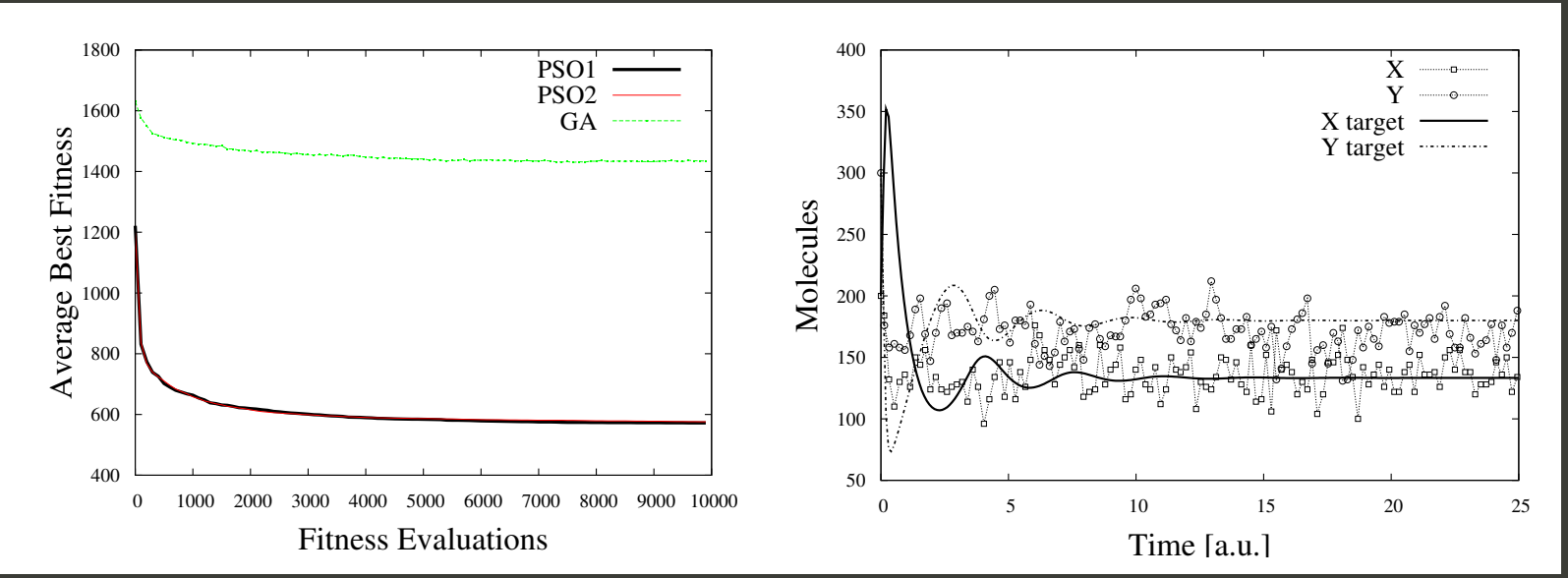


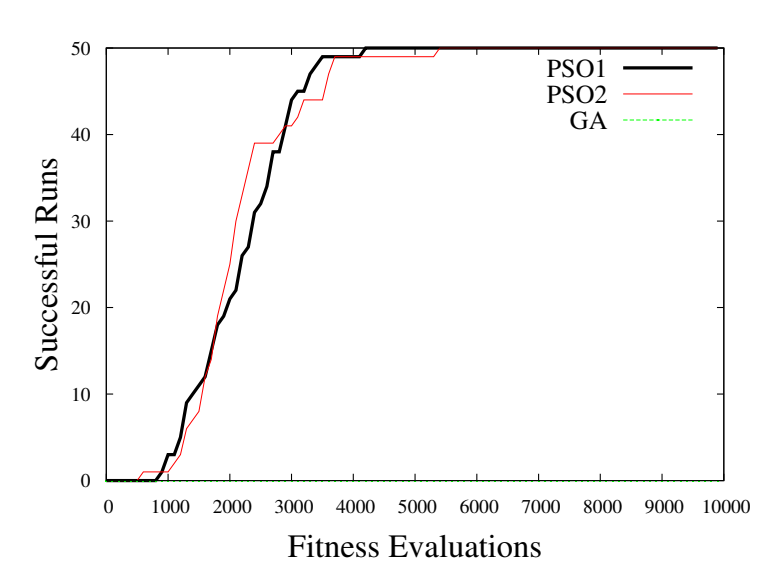




- PSO1/2 best performers
- GA worst average fitnesses
- dynamics faithfully reconstructed

Comparison GA & PSO: Dumped Brusselator





- PSO1/2 best performers
- GA worst average fitnesses
- dynamics unfaithfully reconstructed