INESCPORTO
INSTITUTO DE ENGENHARIA DE SISTEMAS
E COMPUTADORES DO PORTO
LABORATÓRIO ASSOCIADO

Campus da FEUP
Rua Dr. Roberto Frias, 378
4200 - 465 Porto
Portugal

T +351 222 094 000
F +351 222 094 050

www@inescporto.pt
www.inescporto.pt

2007 APR

Stochastic Star Communication Topology in Evolutionary Particle Swarms (EPSO)

Vladimiro Miranda, *IEEE Fellow*
Hrvoje Keko
Álvaro Jaramillo Duque

Evolutionary Algorithms

- Primal search methods rely on two mechanisms to find the optimum of a problem:
  - Movement mechanism
  - Evaluation
- A dumb method is Random Search – evaluation does not influence movement
- Meta-heuristic methods develop smart strategies for searching in the solution space
- In Evolutionary Algorithms, Evaluation leads to Selection: defining the starting points for the new move that will generate new candidates in the solution space, by applying the principles of Natural Selection – discarding the worst and keeping the best (fittest)

Particle Swarm Methods & other

- In Tabu Search, no selection is applied but the movement rule includes provisions to avoid cycling and is influenced by the evaluation values
- In PSO – Particle Swarm Optimization methods, no selection is applied – but the movement rule has dynamic characteristics that lead to progress towards the optimum.
- THESE ARE NOT EVOLUTIONARY METHODS!

Movement Rules in Evolutionary Algorithms

- The movement rule in EA is composed of two parts
  - Mutation
  - Recombination

- Mutation generates a new solution by (randomly) modifying a single individual
- Recombination generates a new individual by (randomly) mixing the characteristics of more than one previous solutions

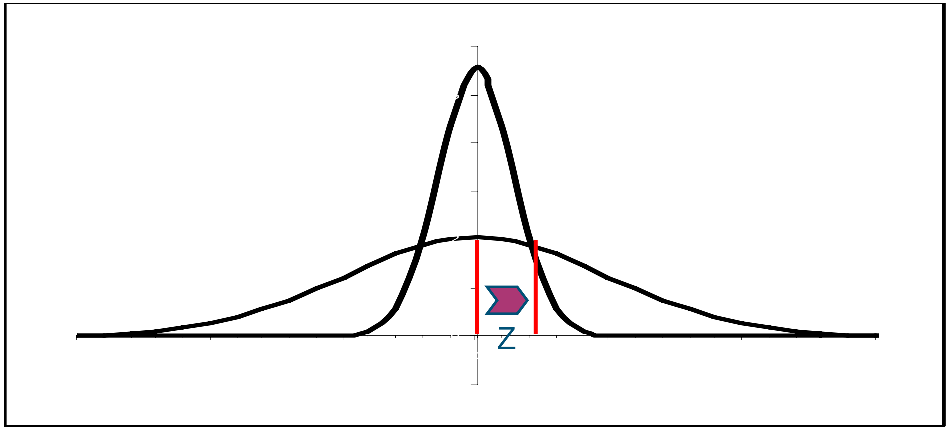
- Neither of these mechanisms contribute to a push towards the optimum!

Mutation

- In Evolution Strategies/Evolutionary Programming, where variables are real numbers, mutation is achieved by random deviations, usually subject to Gaussian mutations controlled by a mutation rate σ

$$\mathbf{Z} = \sigma\left(N_1(0,1),\ldots,N_n(0,1)\right)^t$$

$$\tilde{\mathbf{X}} := \mathbf{X}\ (\mathbf{1}+\mathbf{Z})$$



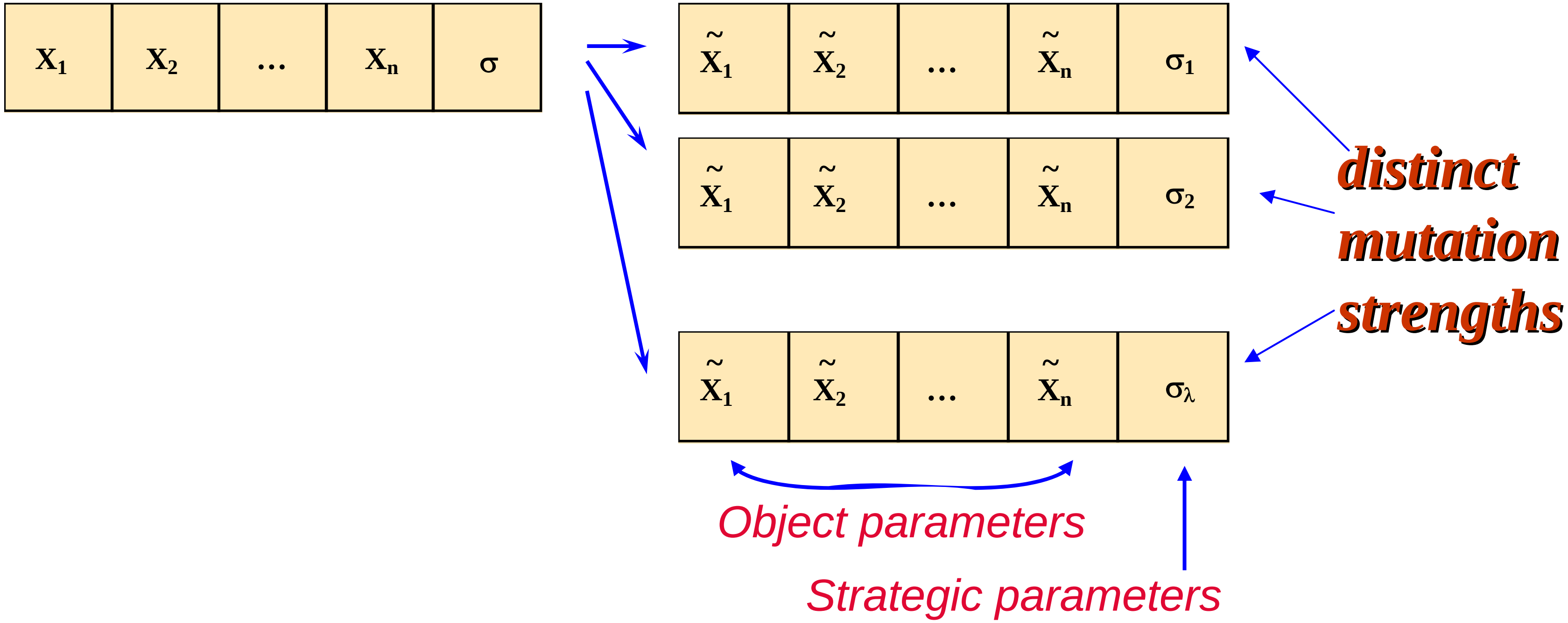
- Mutations may also be multiplicative

$$\tilde{\mathbf{X}} := \mathbf{X}e^{\tau \mathbf{N}(0,1)}$$

- or under a log-normal law

Evolution Strategies: the σSA (1,λ) ES

- In self-adapting Evolution Strategies, each descendent has a distinct mutation strength



- The selection process also selects the most favorable mutation strength – embedded into the fittest descendent

Recombination

- RECOMBINATION: a new individual is formed from the recombination of existing ρ parents.

  Biology has $\rho = 2$

  SCHEMES:

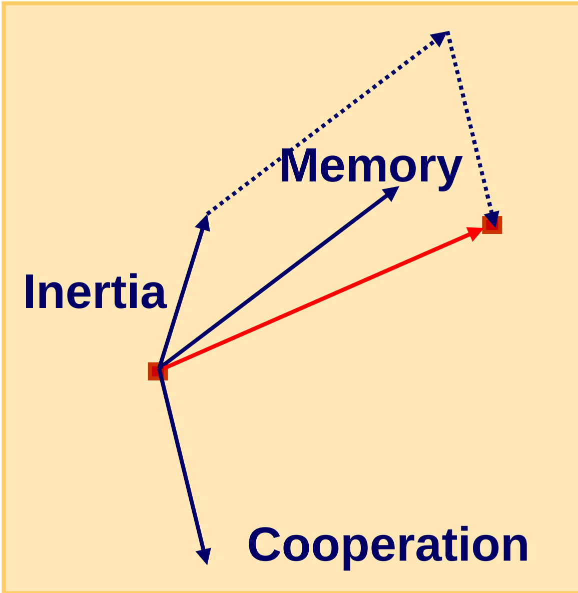
- Uniform crossover: for each variable, randomly select one of the ρ parents to donate its value
- Intermediary recombination: any variable receives a percentage contribution from all the ρ parents - many schemes possible
- Point crossover: define crossover points, and then take parent contributions in turns

Particle Swarm Optimization (Classic PSO)

- A set of particles (solutions) in the search space
- movement of a particle:

$$X_i^{new} = X_i + V_i^{new}$$

- inertia:
  - moving in the same direction
- memory:
  - attraction by particle past best
- cooperation: attraction for global best
- basic model



$$V_i^{new} = V_i + Rnd_1 . w_i^{(1)} (b_i - X_i) + Rnd_2 . w_i^{(2)} (b_g - X_i)$$

Particle Swarm Optimization (Classic PSO)

- Here is an attempt to have evolving weights

$$\mathbf{V}_i^{new} = Dec(t).w_{i0}\mathbf{V}_i + Rnd_1.\mathbf{w}_i^{(1)}\left(\mathbf{b}_i - \mathbf{X}_i\right) + Rnd_2.\mathbf{w}_i^{(2)}\left(\mathbf{b}_g - \mathbf{X}_i\right)$$

- Another model is the constriction factor of M. Clerc

$$V_{ik}^{new} = K_k\left[V_{ik} + Rnd_1 Wm_{ik}(b_{ik} - X_{ik}) + Rnd_2 Wc_{ik}(b_{Gk} - X_{ik})\right]$$

$$K_k = \frac{2}{\left|2 - W_k - \sqrt{W_k^2 - 4W_k}\right|} \qquad W_k = Wm_k + Wc_k \qquad W_k \geq 4$$

- Evolving weights - the need to adapt the progress of the algorithm to the different search phases and the different global and local landscapes of problems

Re-interpretation of the movement rule

- What is the PSO movement rule, really?
  - It is a form of intermediary recombination!
- The following parents are used to produce a new individual:
  - A particle
  - Its direct ancestor
  - Its best ancestor (kept in suspended animation)
  - The best ancestor found by the swarm (also kept in suspended animation, for reproduction purposes)
- The sharing proportion is defined by the weights:

$$X^{new} = (1 + w_I - w_M - w_C)X + w_I X^{old} + w_M b_i + w_C b_g$$

- This rule is biased towards the optimum, it is not neutral!

RECOMBINATION via THE MOVEMENT RULE

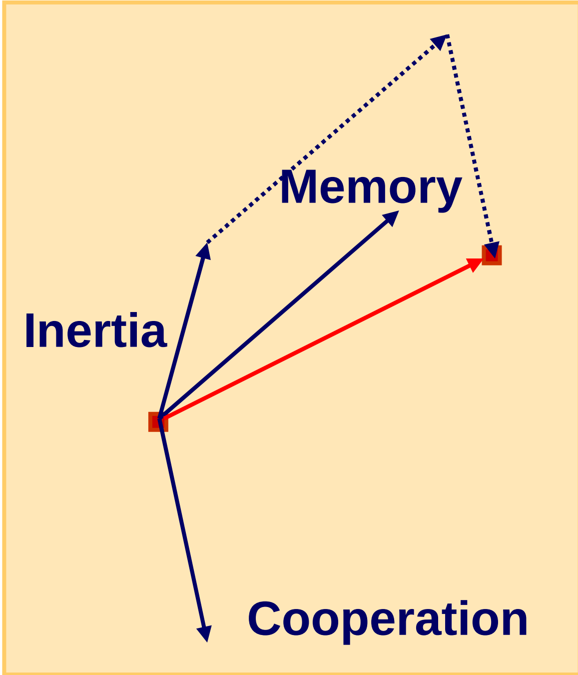
- movement of a particle:

$$ {}^{*}x_i^k = x_i^k + {}^{*}v_i^k $$

- inertia:
  moving in the same direction

- memory:
  attraction by particle past best

- cooperation:
  attraction for global best



$$ {}^{*}v_i^k = w_{i,inertia}^k v_i^k + w_{i,mem}^k (x_i^{k,mem} - x_i^k) + w_{i,coop}^k (x_i^{best*} - x_i^k) $$

EPSO as a self-adaptive recombination process

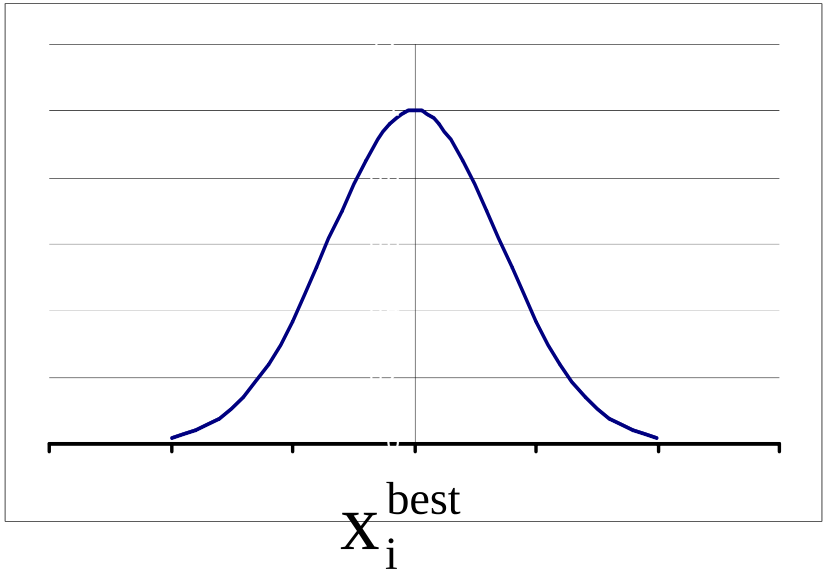
- Each weight (strategic parameter) suffers mutation

$$*w_{i,j}^{k} = w_{i,j}^{k}(1 + \tau N[0, \sigma^2])$$

- ... an Evolutionary Process !!!
- PLUS - the global best has a "foggy" definition

$$x_i^{best*} = x_i^{best}(1 + \tau' N[0,1])$$

Exploration strength



EPSO as a self-adaptive Evolution Strategy

The particles “do not move”: they reproduce

- REPLICATION - each particle is replicated r times *(cloning)*
- MUTATION - each clone has its weights w mutated
- RECOMBINATION - each mutated particle generates 1 offspring according to the particle movement rule
- EVALUATION - each offspring has its fitness evaluated
- SELECTION - by stochastic tournament (or elitism) the best particle in each group of r survives to form a new generation

(the best particles carry with them, to the following generation, their mutated weights)

Self-adaptation of the recombination operator

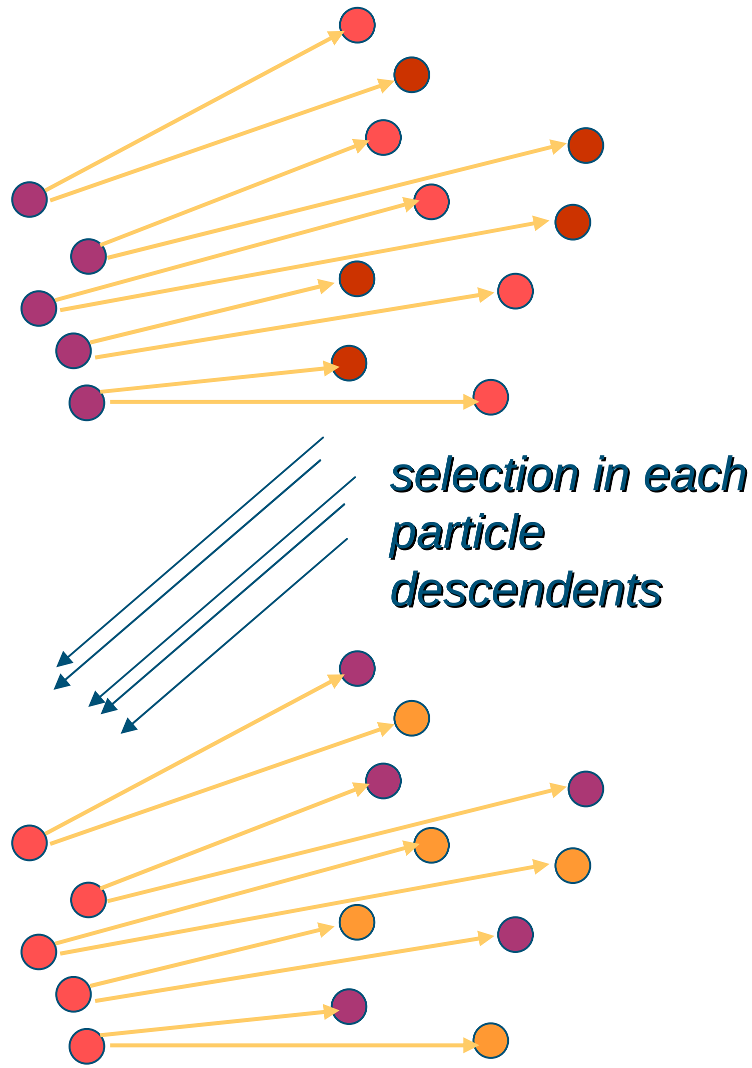
- Previous self-adaptive Evolutionary Algorithms have been designed to make the mutation operator evolve – a mutation rate is subject to selection and stays attached to a solution and its descendents.

- In EPSO, it is the recombination operator that is made to evolve, in a special form of intermediary recombination (the movement rule), by self-adapting the proportions of contributions of parents in forming an offspring

- The weights are strategic parameters and like object parameters are subject to selection and are passed to the descendents

EPSO in action

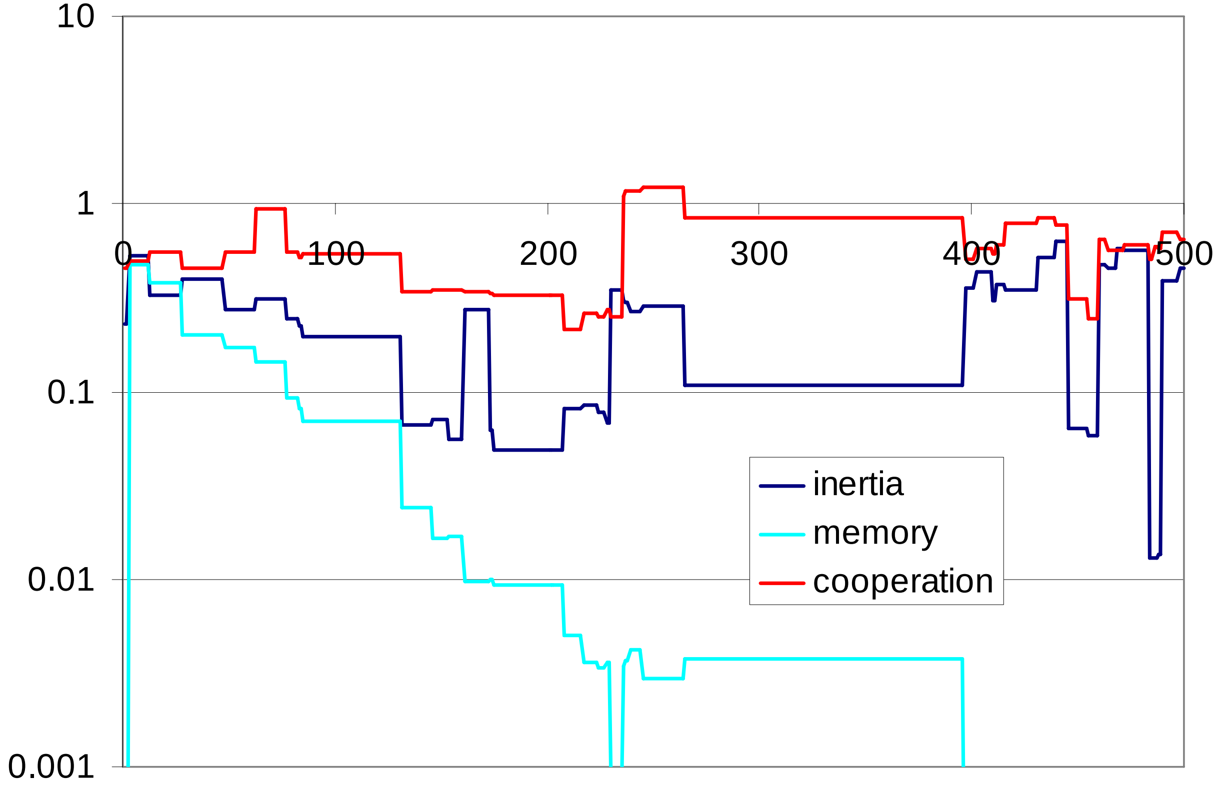
- Selection acts separately on the descendents of each particle
- It is a parallel process where the interaction among particles is assured by the recombination rule
- Recombination proportion is evolving under selection pressure



Evolving weights

Weights evolve and adapt

Evolution of weights of successive global best



- The following slides are showing one *real life* example where EPSO competed against other algorithms and proves to be a winner

Competition against other algorithms

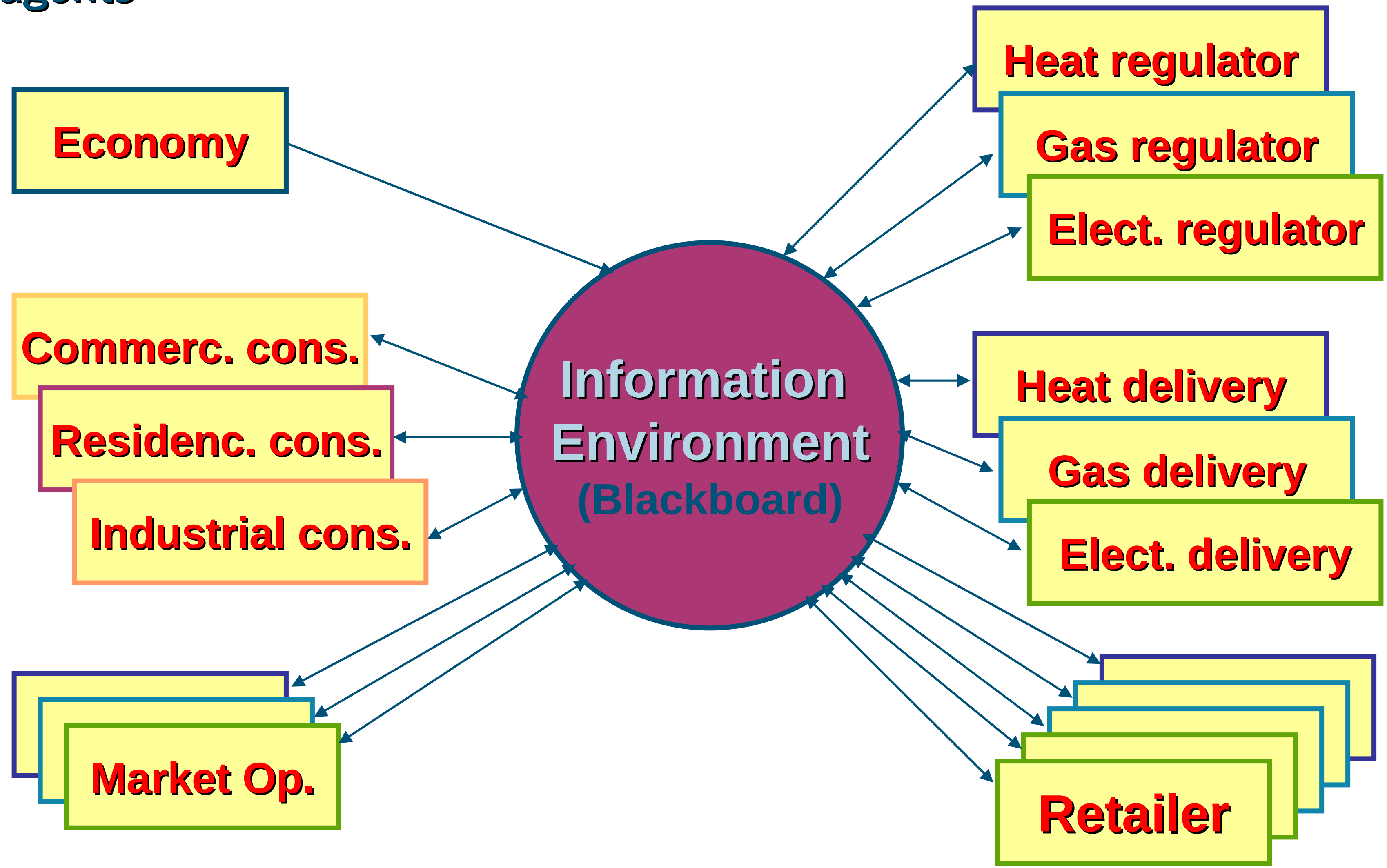
- An intelligent agent platform simulating a multi-energy retail market
- A retailer agent with the capacity to perform internal simulations to optimize its market strategy

- If we equip distinct retailers with different algorithms, and then run for some time a complex market simulation, will there be a winner?

- A test for 24 month simulation, allowing an internal 2 month ahead simulation

A model for the retail market

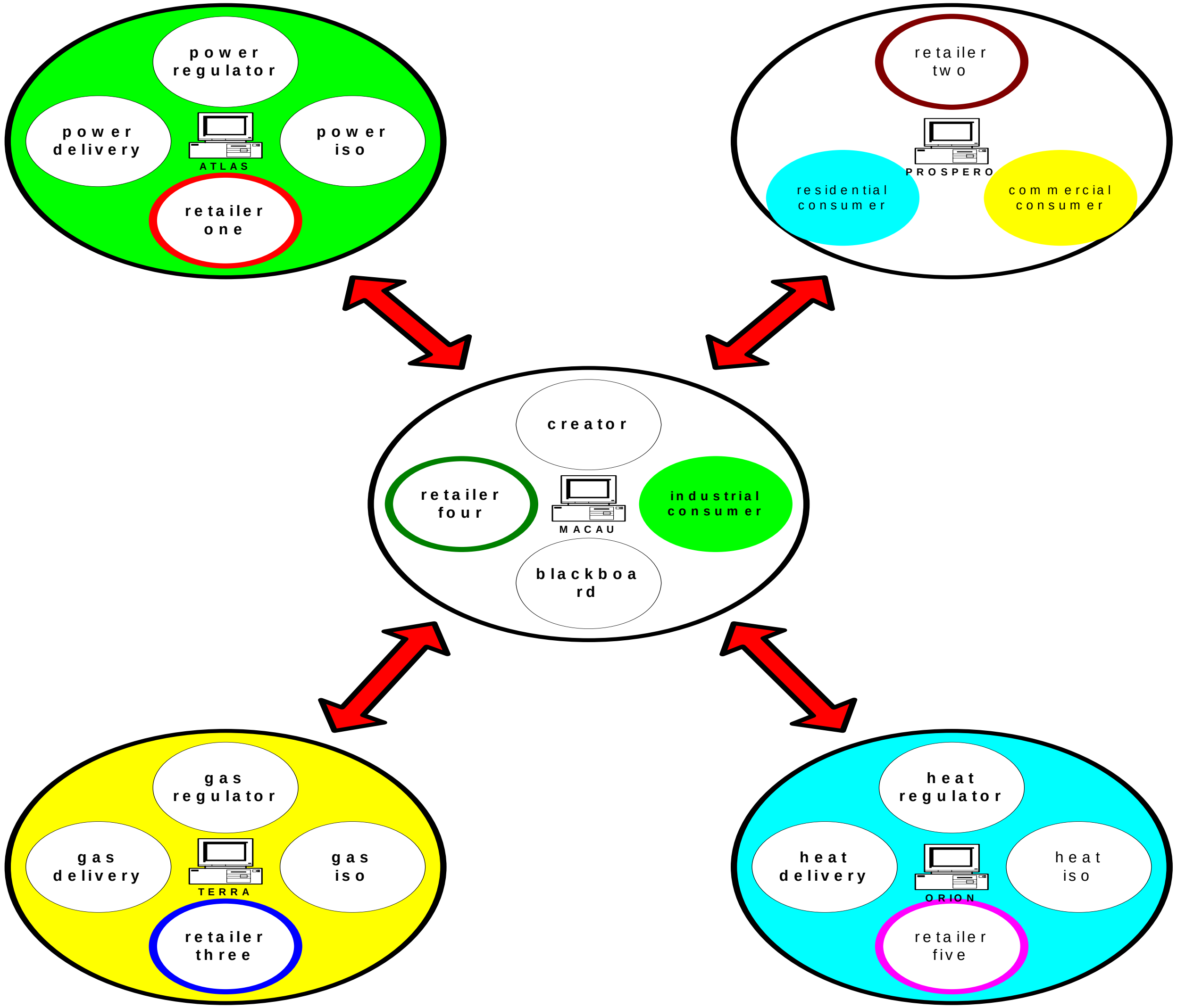
19 agents



Parallel processing with JADE platform

JADE

FIPA compliant



5 PCs - LAN

A simulation within the simulation

- Each retailer agent optimizes its strategy by simulating market evolution before making a move (such as changing prices, deciding investment, etc.)
- This simulation aims at optimizing a vital function of the agent - could be maximizing profit, or keeping market share, for instance.

Competing algorithms

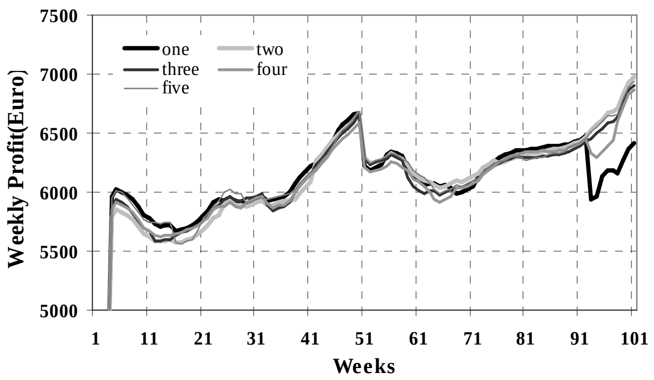
- Basic EPSO
- Basic PSO
- SSGA (steady state): a Genetic Algorithm with elitism (the best 20% always surviving)
- MPGA (multiple population): a Genetic Algorithm with two populations exchanging 2 individuals per generation
- DCGA (deterministic crowding): a Genetic Algorithm with a special rule for selection

Experiments

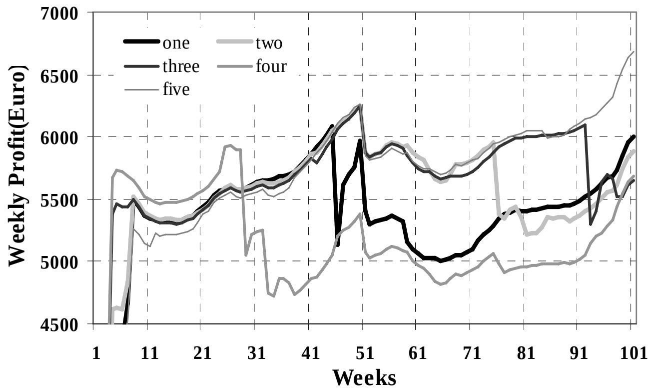
- Fixed population for all algorithms – 20 individuals
- Same fitness (maximizing profit for the retailer)
- Same stopping criterion: no improvement in the fitness function in 10 consecutive generations

Experiment...

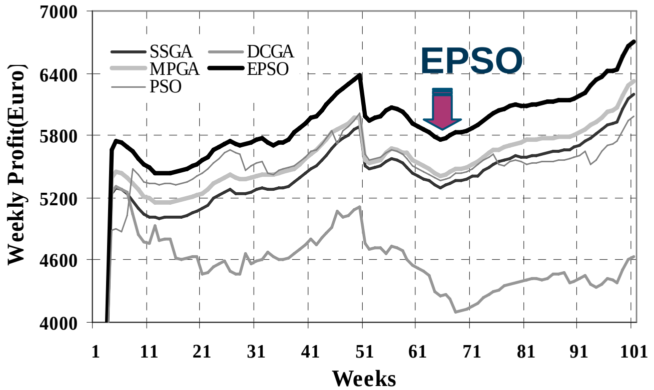
5 runs of EPSO



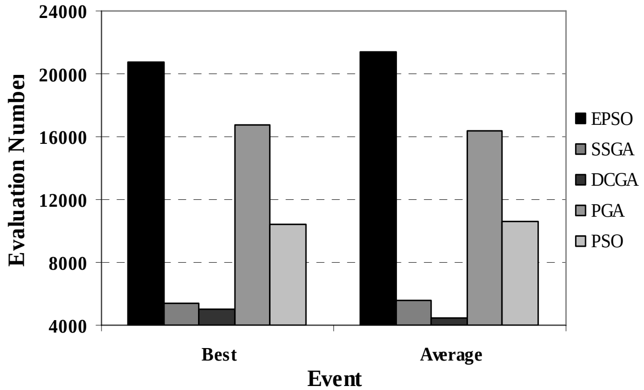
5 runs of PSO



Average of profits in 5 runs



Effort in no. evaluations

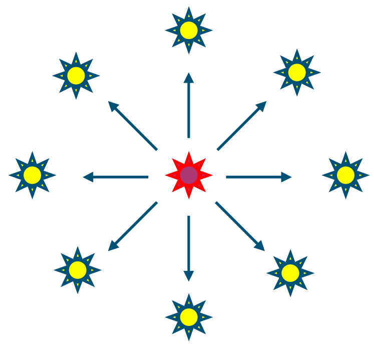


Back to…

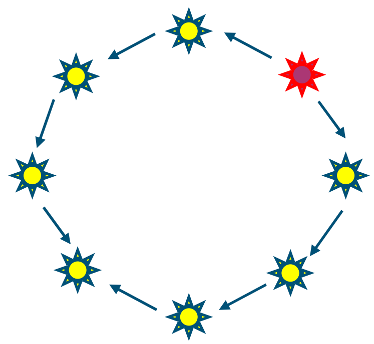
- after we have seen an real life example – there is another thing in EPSO that differs from other PSOs

Communication structures among particles

- Classical communication structure: the star, where all individuals share at the same time the knowledge about the location of the new global best
- Too much communication is against exploration of the search space – may induce premature convergence



- Alternative structure: the ring, where each particle only commuunicates with two neighbours - information about a new global best takes time until it reaches all individuals
- Too little communication risks approaching the process to a set of parallel independent individual searches



Stochastic communication

- EPSO: stochastic star

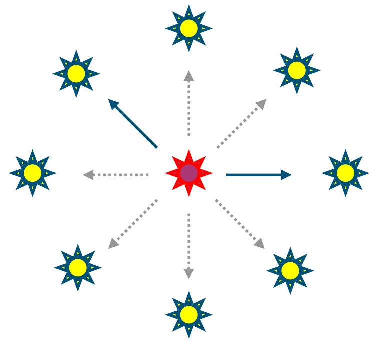
  There is a communication probability threshold p below which communication is allowed and above which information about the global best does not pass to an individual.

- Probability threshold p is applied to each dimension of an individual – it may receive information in some dimensions and have it blocked in other!

- Experiments led to adopting a value of

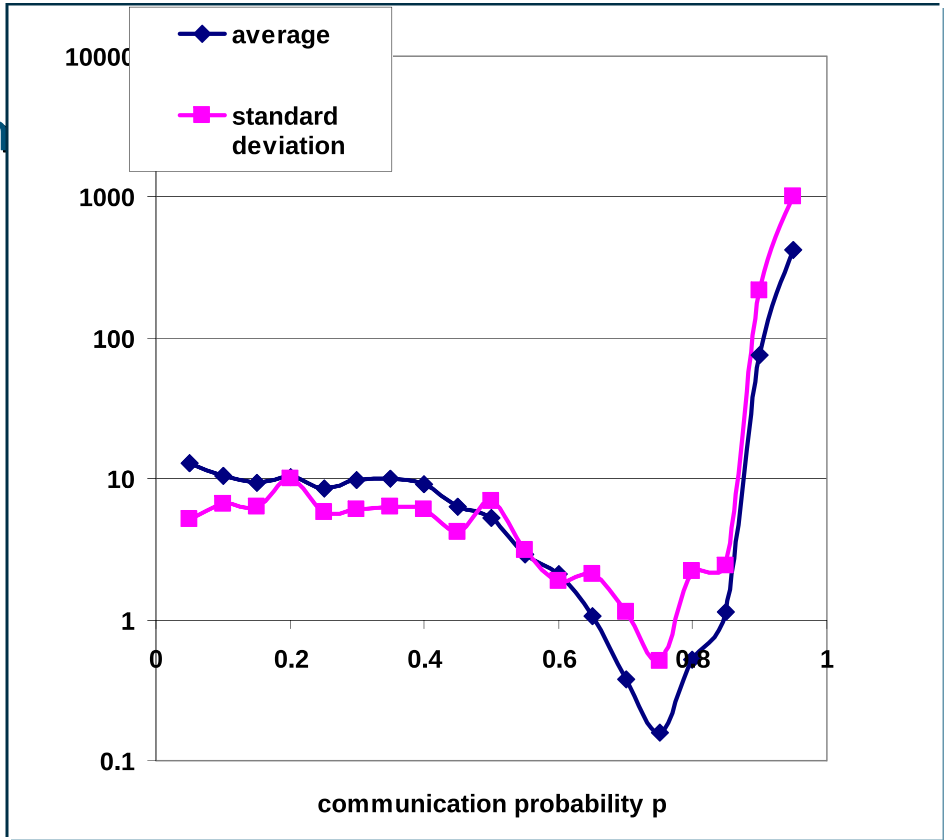
    - p = 0,2

(as a "rule of thumb")



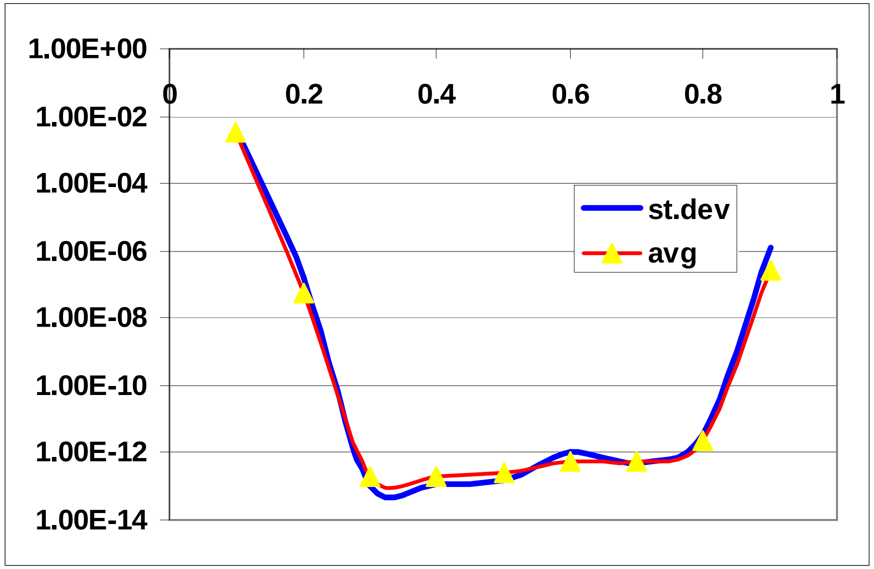
Rosenbrock function and stochastic star

- Rosenbrock function
- 20 runs, 50.000 fitness function evaluations
- average error and standard deviation of 20 runs is shown
- p=1 isn't shown on the graph, it leads the algorithm into premature convergence
- note: Y-axis scale is logarithmic



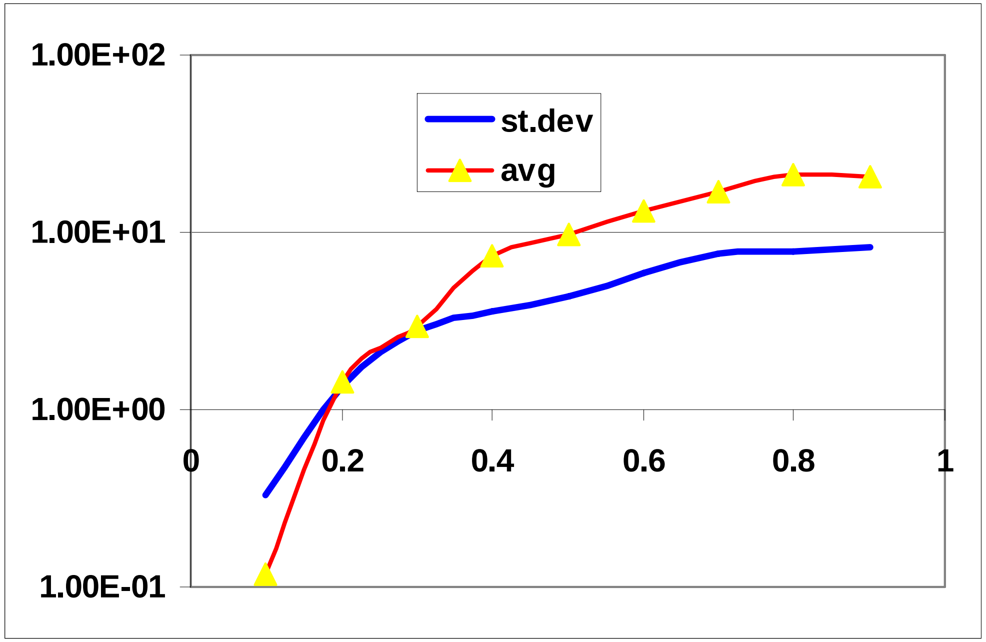
CEC 2005 f2 Schwefel's problem

- function from battery of tests for non-constrained real-valued optimization according to CEC2005 conference
- stopping criteria is a fixed number of fitness function evaluations
- very low values of p lead to bad results, as well as very high
- relatively insensitive to changing p in central area
- curve of standard deviation of solutions between runs follows shape of achieved error values



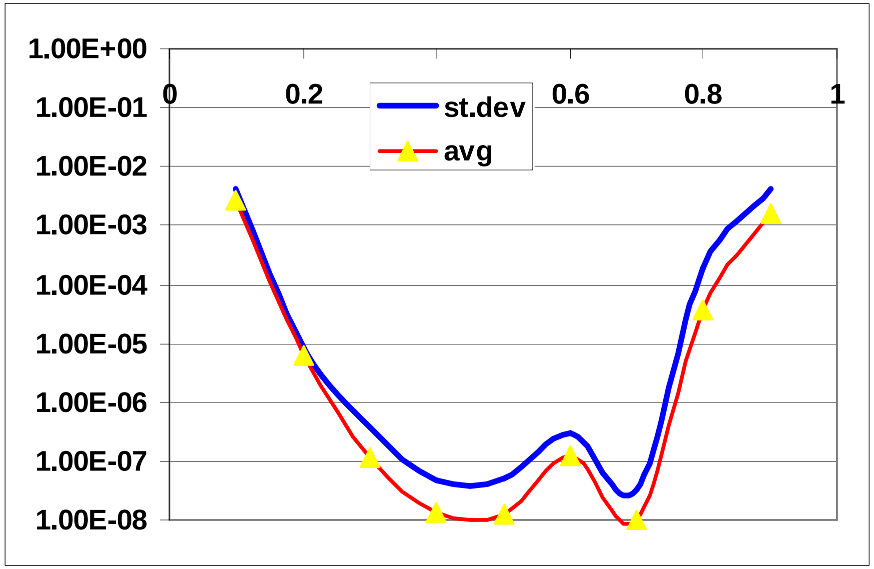
CEC 2005 f9 Rastrigin's function

- for this function lower values of communication probability are significantly better
- even though this function *reacts* differently, standard deviation again follows shape of achieved error values



Sphere function

- simple problem, so with higher number of fitness function evaluation algorithm always converges
- number of fitness function evalutions is low
  - 5000, 30 dimensions
- once again - the standard deviation follows the behavior of error values



Another from real life: Reactive Power Planning

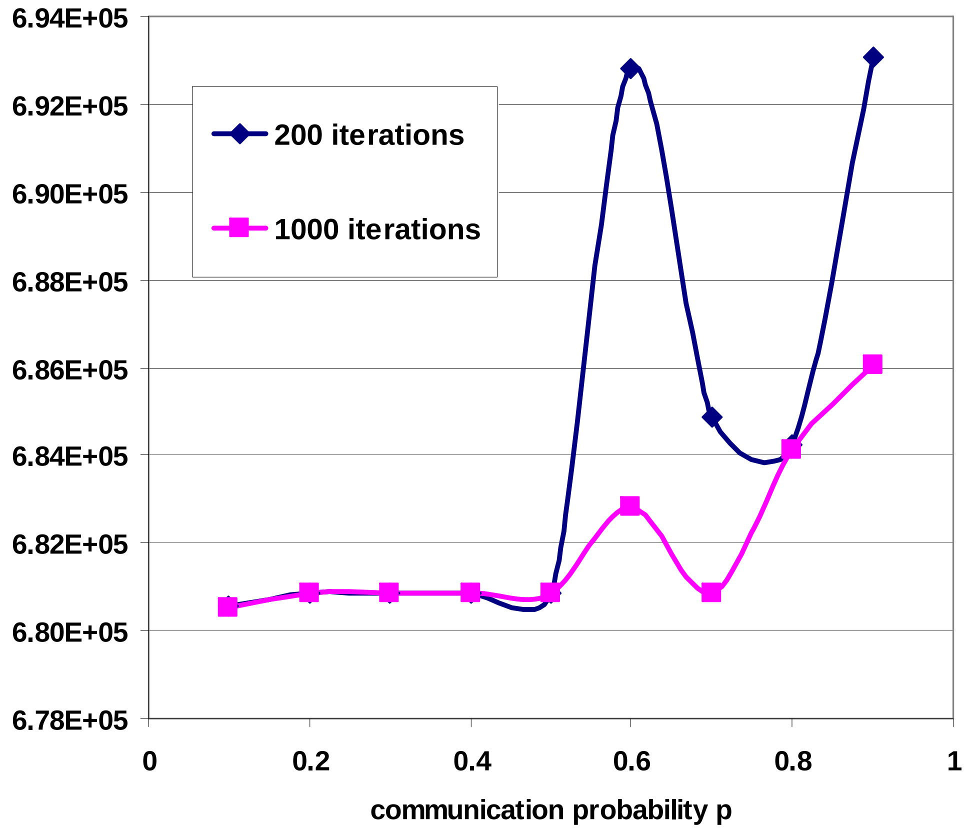
- a difficult nonlinear, multiobjective problem from power systems
- minimize total investment costs - while keeping network conditions appropriate
  - utilizing penalties to reflect planner's decisions
  - weighted sum of investment and penalties
- discrete and continuous control variables
  - not all operating limits are self-constrained in variables
  - need to utilize power flow calculations in a "heavyweight" fitness function
- large search spaces and demanding calculations
  - sizes of typical electric networks indicate number of search variables

Reactive Power Planning

- our implementation of EPSO takes advantage of power flow calculation library
- a multitude of input variables need to be included in planning process
- the application can handle various network load levels within single optimization process
  - during the night the operating conditions are different!!
- as a planning algorithm, EPSO has proven to be successful and robust
  - how does the stochastic star probability influence its performance on this problem?

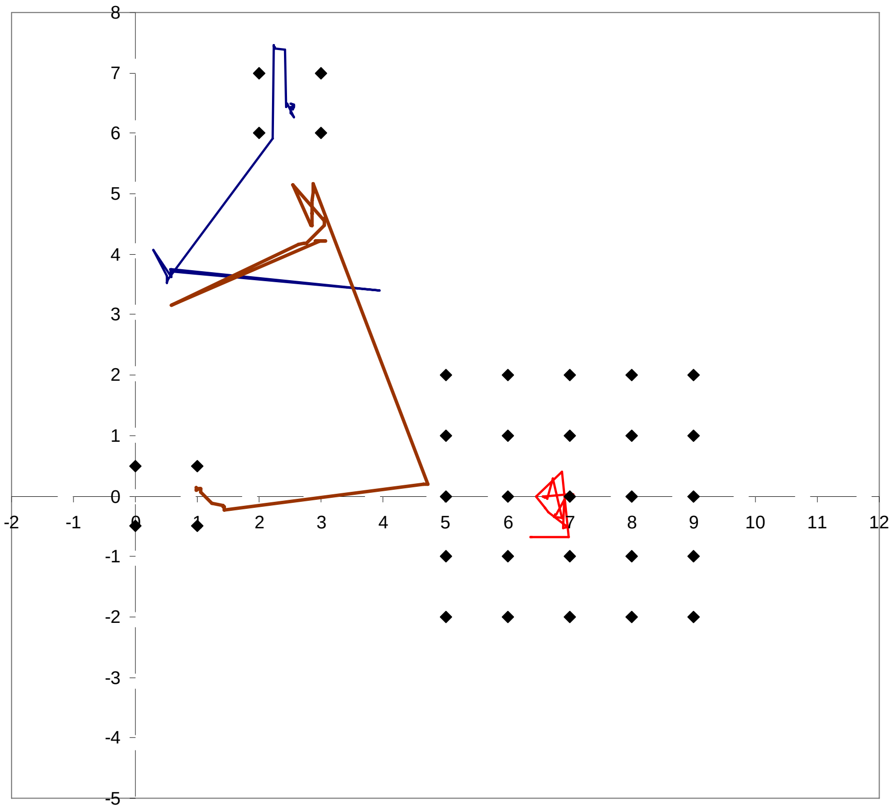
EPSO and Reactive Power Planning (2)

- 30 node network
- total cost of best solution shown, after 200 and 1000 iterations
- this problem (and this setup of input variables!) relatively insensitive to changing the *p*
- appropriate *p* and fast convergence especially interesting for using EPSO in close-to-real time applications

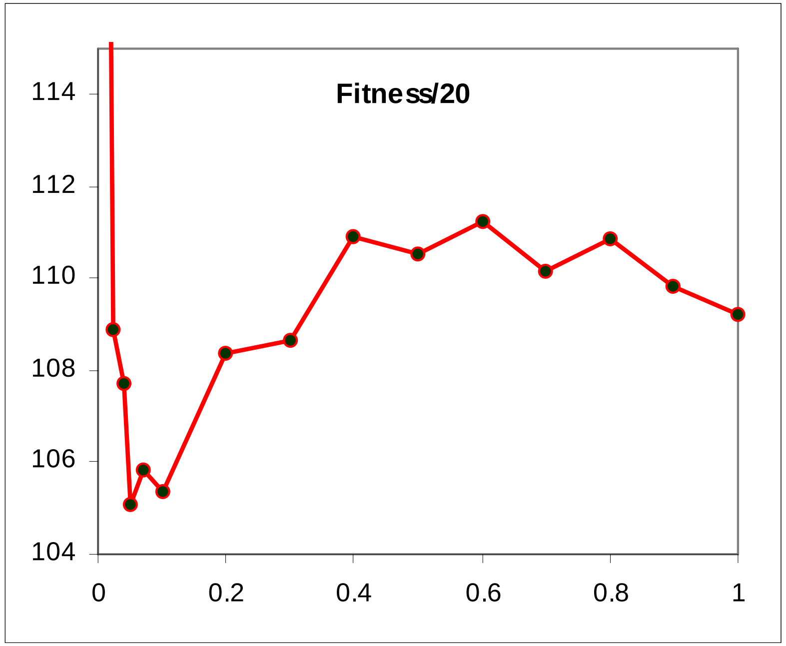


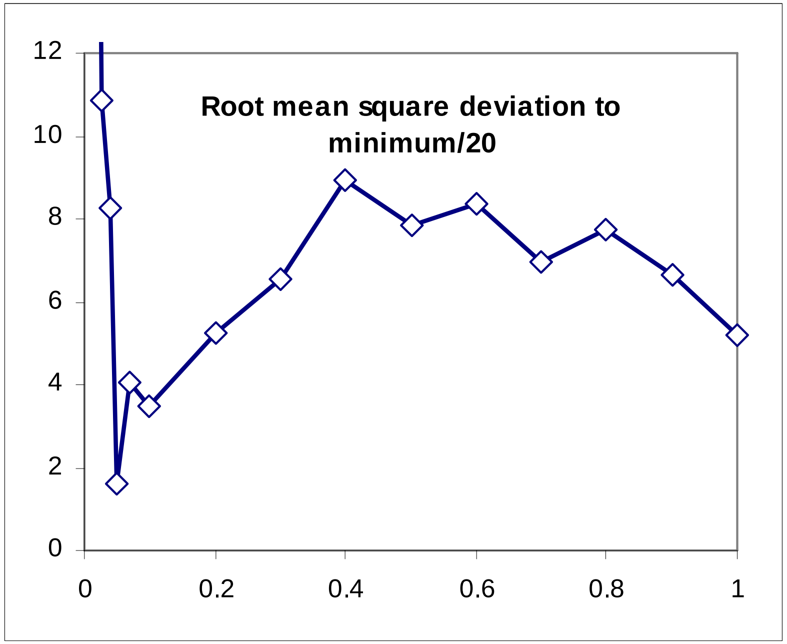
EPSO and Clustering Problem

- another problem from *mathematical world* – but clustering problems are also common in applications
- in power systems – grouping customers based on their demand curves into customer groups
- as an illustration: finding central points of three clusters shaped like squares
  - convergence process illustrated



EPSO in Clustering Problem





- similar to previous stochastic star tests, 20 runs of EPSO algorithm, average and deviation
- once again - best fitnesses and standard deviation behave similarly
- optimal value of *p* low – for this problem weak communication is better

Finally…

- as it is already proven – it is important *not* to have strict, fixed communication topology
- stochastic star topology of communication improved EPSO performance, both in test problems and real life problems
  - it is also simple for implementation!
- however: optimal value of "p" isn't unique, depends on the problem!
- this indicates we should step towards (truly) adaptive communication topologies
  - this is a promising direction of research; how to sample search space, what does the distance between particles mean…

INESCPORTO
INSTITUTO DE ENGENHARIA DE SISTEMAS
E COMPUTADORES DO PORTO
LABORATÓRIO ASSOCIADO

www.inescporto.pt

Merci!