Bird flocking and power plants

Axel Groniewsky

May 25, 2011

Introduction

- In the field of energy engineering where a completely new power plant concept is rarely developed, more often an existing design is adjusted to fit specific site and project requirements fine tuning and therefore optimization methods have great importance.
- Classical optimization techniques have limited scope in practical applications.
- Multi-criteria and multi-objective optimization problems of power plants demand state of the art optimization methods.

PSO

- A heuristic optimization technique based on swarm intelligence that is inspired by the behavior of bird flocking or fish schooling.
- Developed in 1995 by James Kennedy (social-psychologist) and Russell Eberhart (electrical engineer).

The basic principle of PSO I

- Each particle moves around in the search space looking for the optimum
- Each particle has a position and a velocity.
- Each particle remembers the position it was in where it had its best result so far (its personal best)

The basic principle of PSO II

- The particles in the swarm *co-operate*. They exchange information either directly or indirectly about what they've discovered in the places they have visited
- The co-operation of the classical PSO is simple:
  - A particle has a *neighborhood* associated with it.
  - A particle knows the fitnesses of those in its neighborhood, and uses the *position* of the one with best fitness.
  - This position is simply used to adjust the particle's velocity

Computational Implementation of PSO

PROBLEM STATEMENT:

D-dimensional minimization problem

$$\mathrm{Min}\ f(X),\ \ X = [x^1, \ldots x^j, \ldots x^D]$$

- where X, as a member (particle) of the swarm is a solution to be optimized in a form of a D-dimensional vector.

Classical PSO

$$v_i^j = w \cdot v_i^j + c_1 \cdot rand1_i^j \cdot \left(pbest_i^j - x_i^j\right) + c_2 \cdot rand2_i^j \cdot \left(gbest^j - x_i^j\right)$$

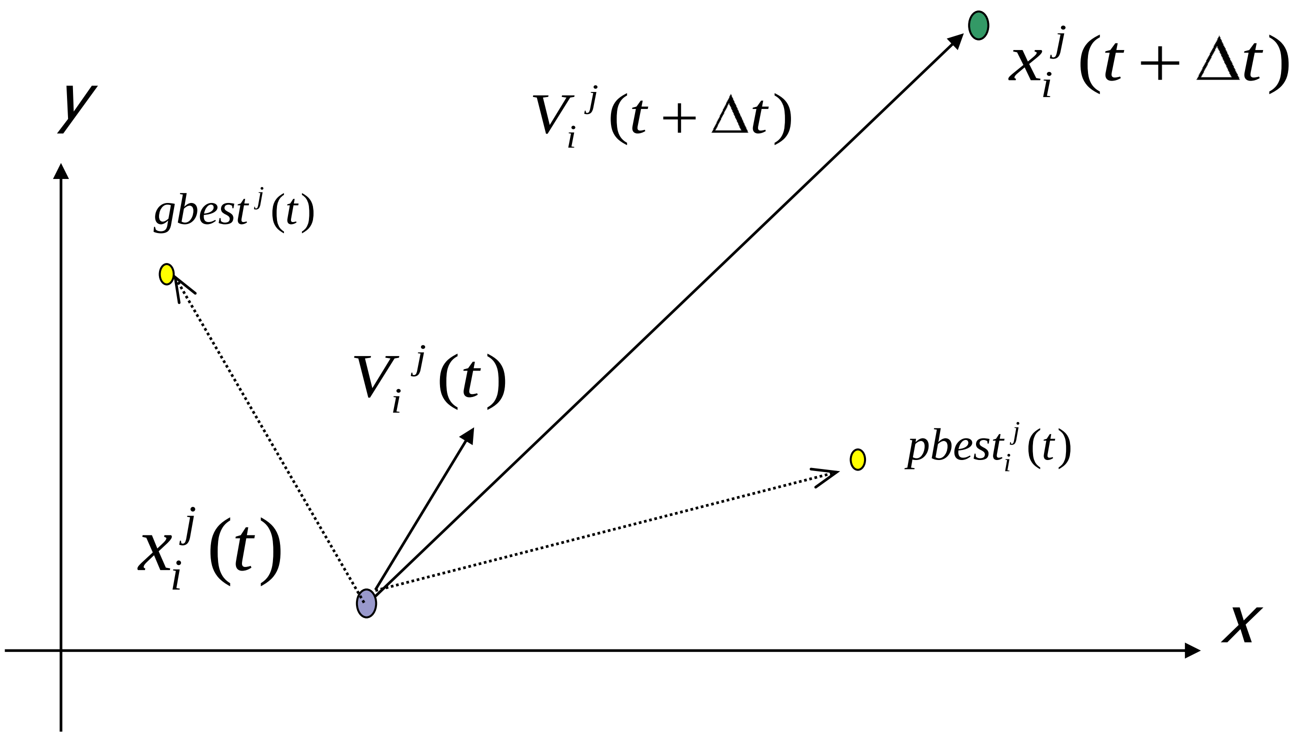
$$\mathrm{x}_\mathrm{i}^\mathrm{j} = \mathrm{x}_\mathrm{i}^\mathrm{j} + \mathrm{v}_\mathrm{i}^\mathrm{j} \cdot \Delta \mathrm{t}$$

- $\mathrm{x}_\mathrm{i}^\mathrm{j}$: position of the ith particle in the *jth dimension*
- $\mathrm{v}_\mathrm{i}^\mathrm{j}$: velocity of the ith particle in the *jth dimension*
- $\mathrm{pbest}_\mathrm{i}^\mathrm{j}$: best position of the ith particle in the *jth dimension*
- $\mathrm{gbest}_\mathrm{i}^\mathrm{j}$: overall best position of the swarm in the *jth dimension*
- c_1 , c_2 : cognitive and social learning rates
- $rand1_i^j$, $rand2_i^j$: randomly generated numbers
- $\Delta\mathrm{t}$:time steps between two iterations
- w : inertia weight

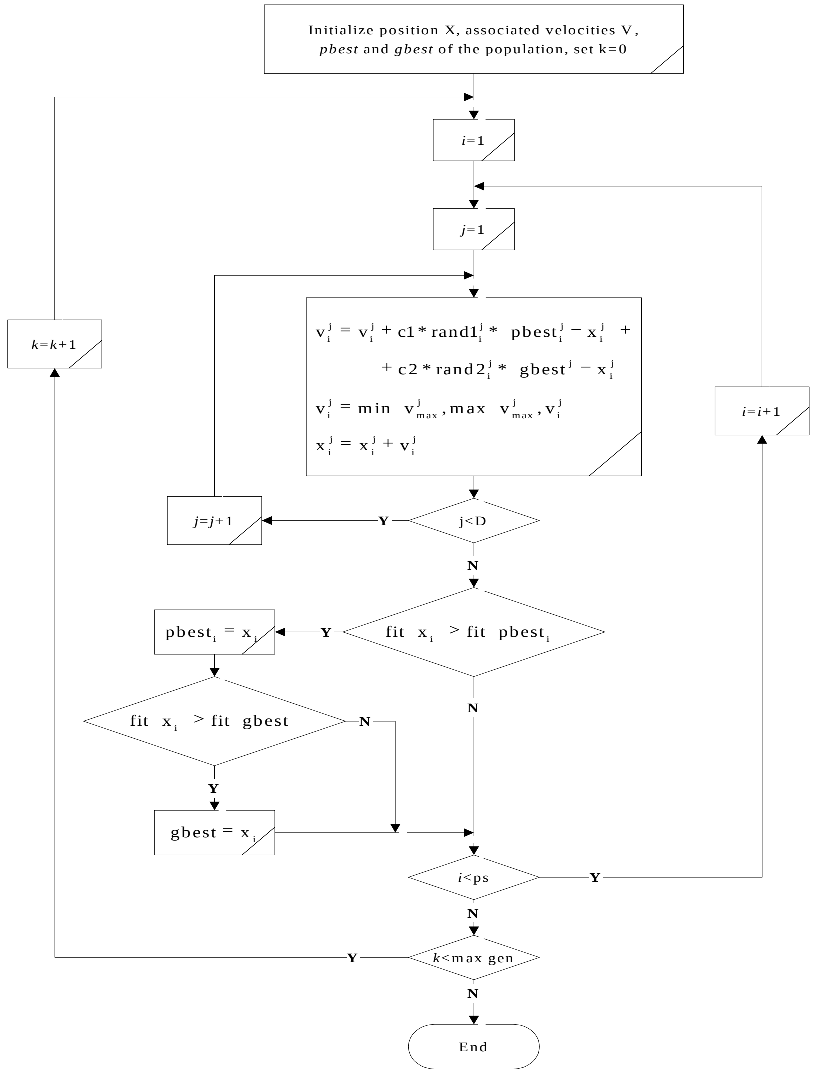
Classical PSO

$$v_i^j = w \cdot v_i^j + c_1 \cdot rand1_i^j \cdot \left(pbest_i^j - x_i^j\right) + c_2 \cdot rand2_i^j \cdot \left(gbest^j - x_i^j\right)$$

$$\mathrm{x}_\mathrm{i}^\mathrm{j} = \mathrm{x}_\mathrm{i}^\mathrm{j} + \mathrm{v}_\mathrm{i}^\mathrm{j} \cdot \Delta \mathrm{t}$$



Flowchart of the conventional PSO

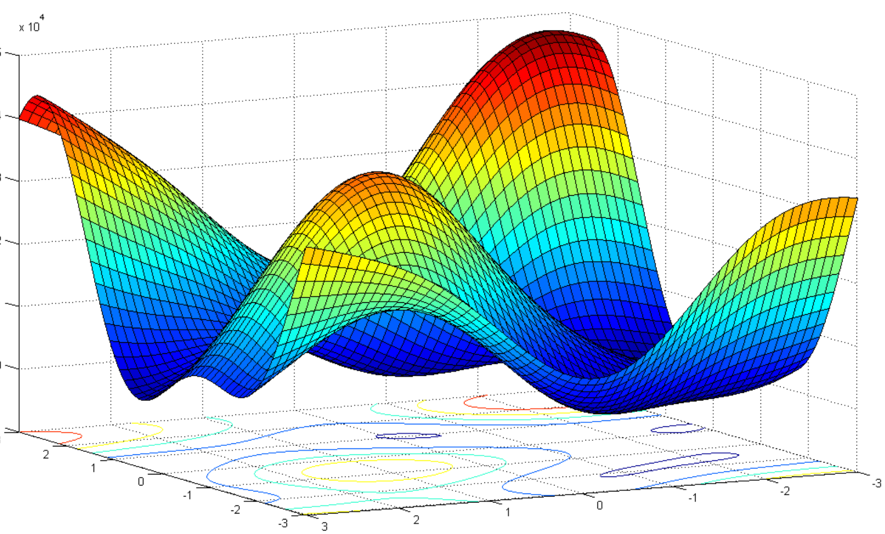


Simulation

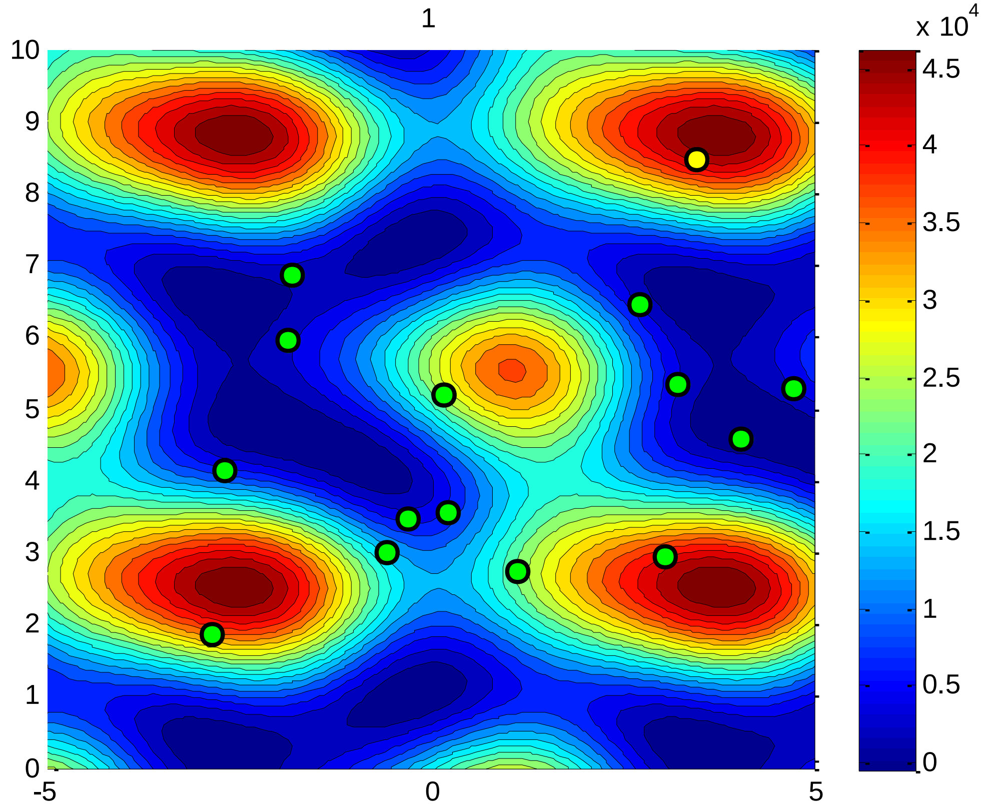
Schwefel’s problem 2.13

$$S=\sum_{i=1}^{D}\left(\sum_{j=1}^{D}\left(a_{ij}\cdot\sin(\alpha_j)-b_{ij}\cdot\cos(\alpha_j)\right)-\sum_{j=1}^{D}\left(a_{ij}\cdot\sin(x_j)-b_{ij}\cdot\cos(x_j)\right)\right)^2$$

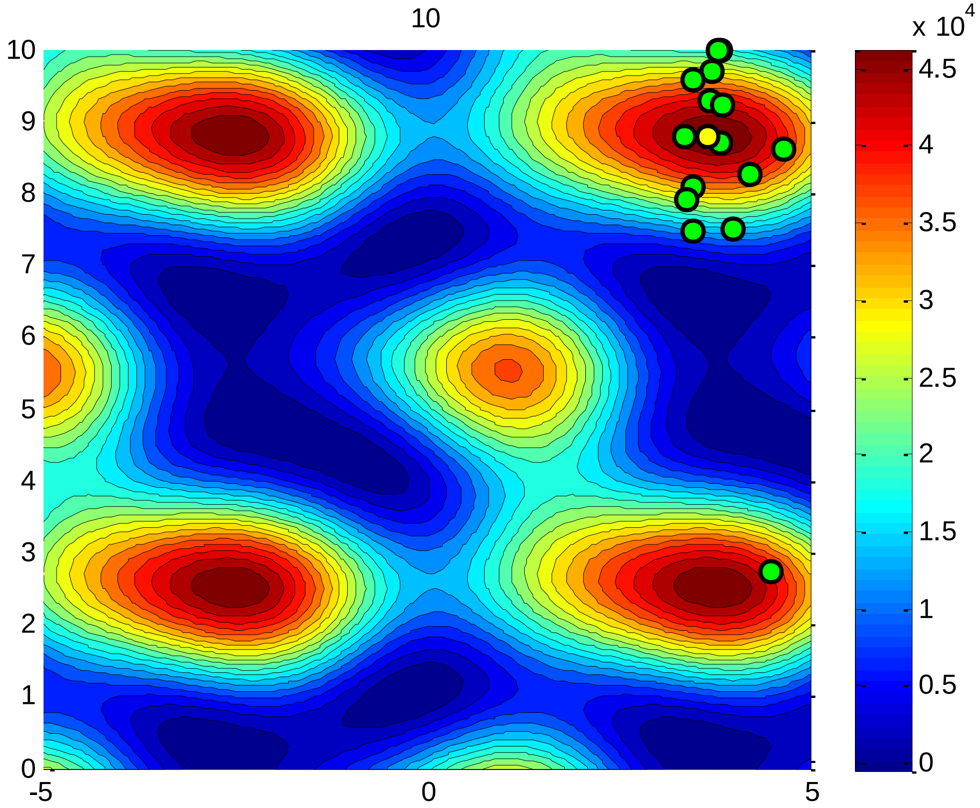
- a_{ij}, b_{ij} : randomly generated numbers in the interval of [-100, 100]
- α_j :randomly generated number in the interval of $-\pi, \pi$



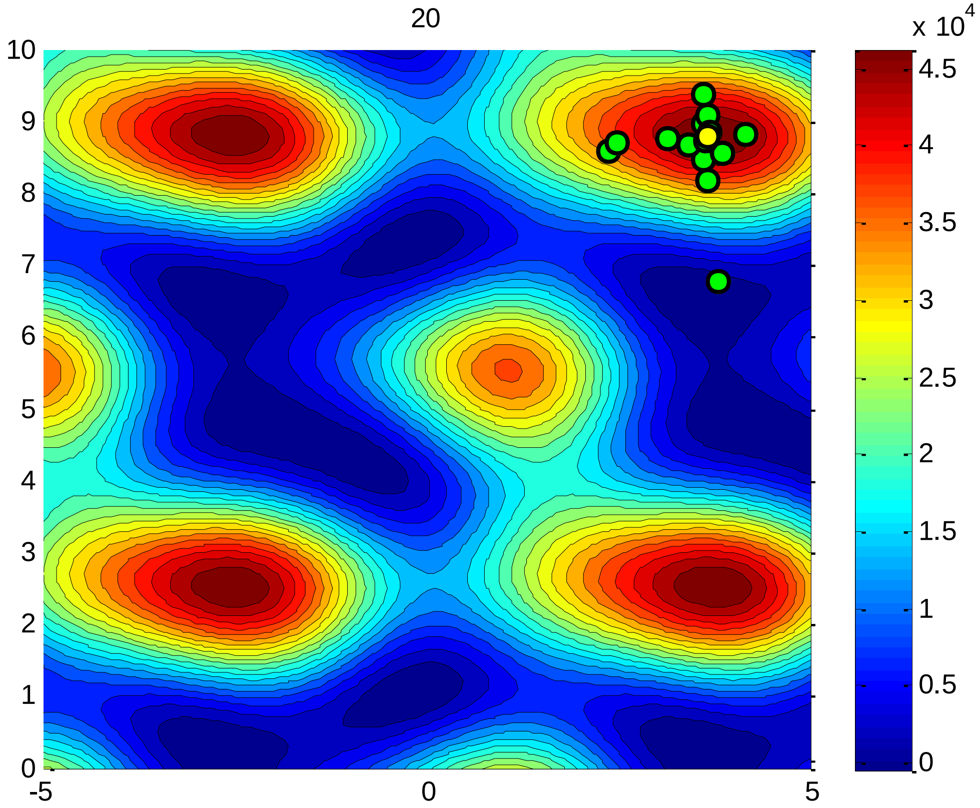
Simulation 1



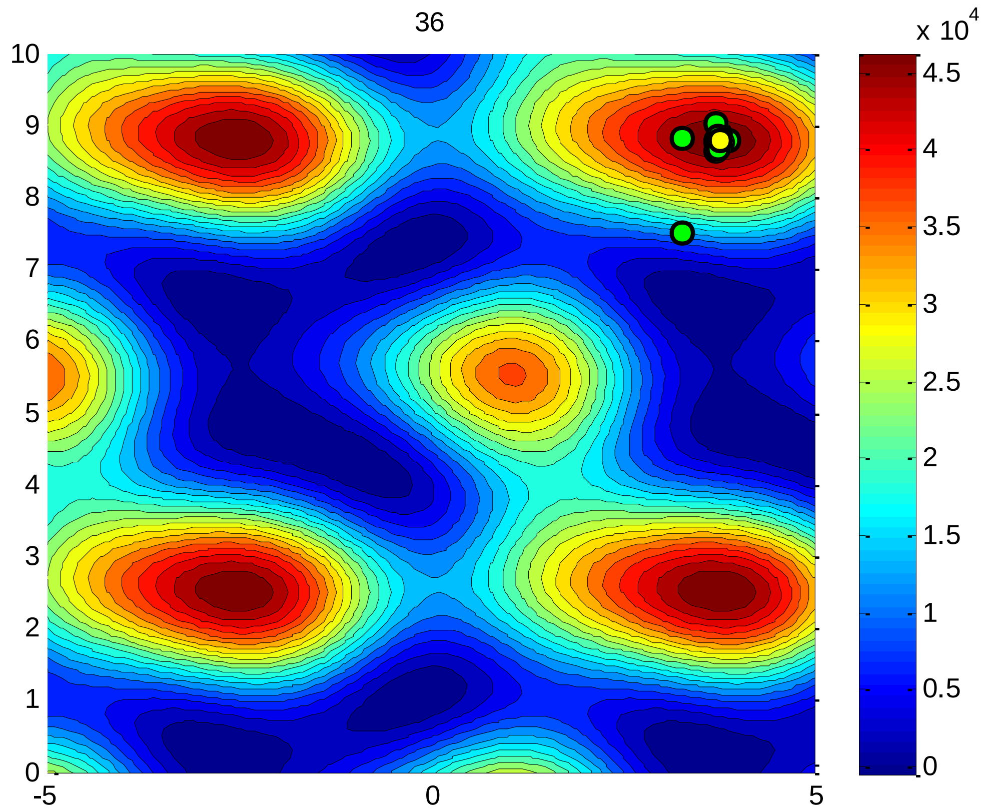
Simulation $_2$



Simulation$_3$



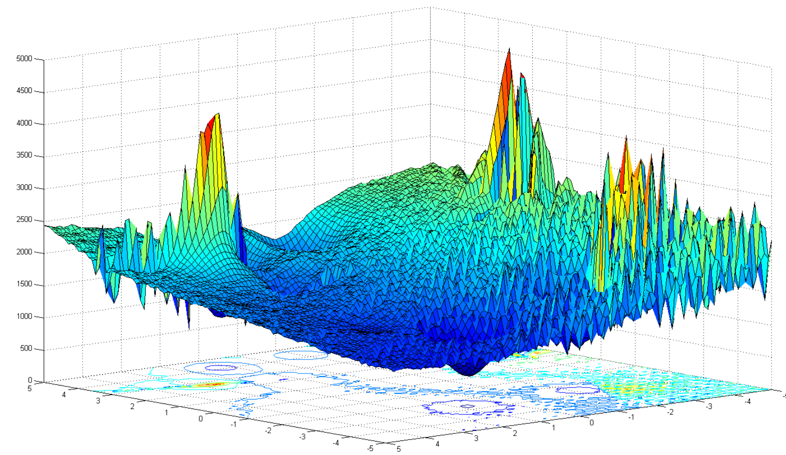
Simulation$_4$



Unimodal and multimodal benchmark functions

- Sphere Function

$$f(x)=\sum_{i=1}^{n} x_i^{2}, x \in [-5,5]^n$$



- Rosenbrock Function

$$f(x)=\sum_{i=1}^{n-1}\left(100\left(x_{i+1}-x_i^{2}\right)^2+\left(x_i-1\right)^2\right), x \in [-10,10]^n$$

- Rastrigin Function

$$f(x)=\sum_{i=1}^{D}\left(x_i^2-10\cos(2\pi x_i)+10\right)$$

- Ackley Function

$$f(x)=20+e-20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n}x^2}\right)-\exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)\right), x \in [-32,32]^n$$

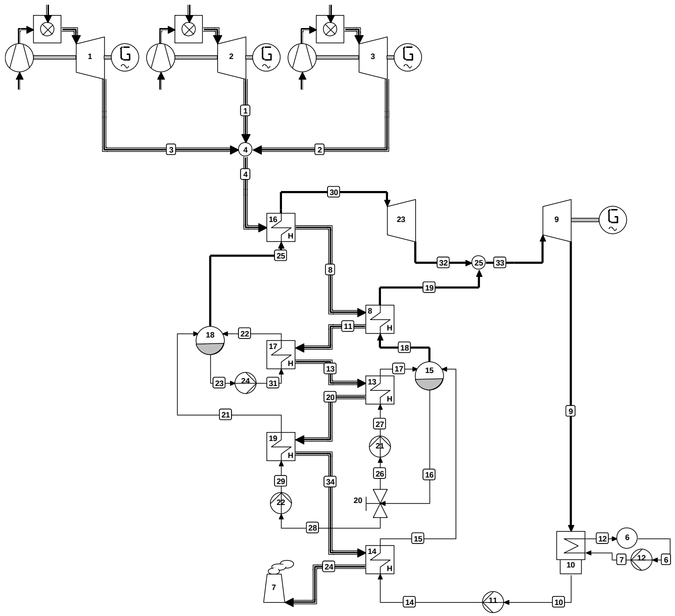
- Hybrid Composition Functions

State of art PSO methods

- Fully informed PSO (FIPS)
  - each individual is influenced by its neighbors
- Cooperative particle swarm optimizer (CPSO-H)
  - one-dimensional swarms search each dimension separately before integrating the results by a global swarm
- Comprehensive learning particle swarm (CLPSO)
  - developed to encourage the diversity of the swarm
- Example-based learning particle swarm (ELPSO)
  - developed to keep balance between diversity and convergence speed
- Self-adaptive learning based particle swarm (SLPSO)
  - simultaneously adopts four PSO based search strategies

Implementation of PSO for power plants optimization problems I

- They are designed to help decision makers
  - It can assist process designers in the development of a cost-effective power plant concept
  - A superstructure (design case) shall be developed which includes a limited number of the most likely design alternatives with estimated values of the process variables

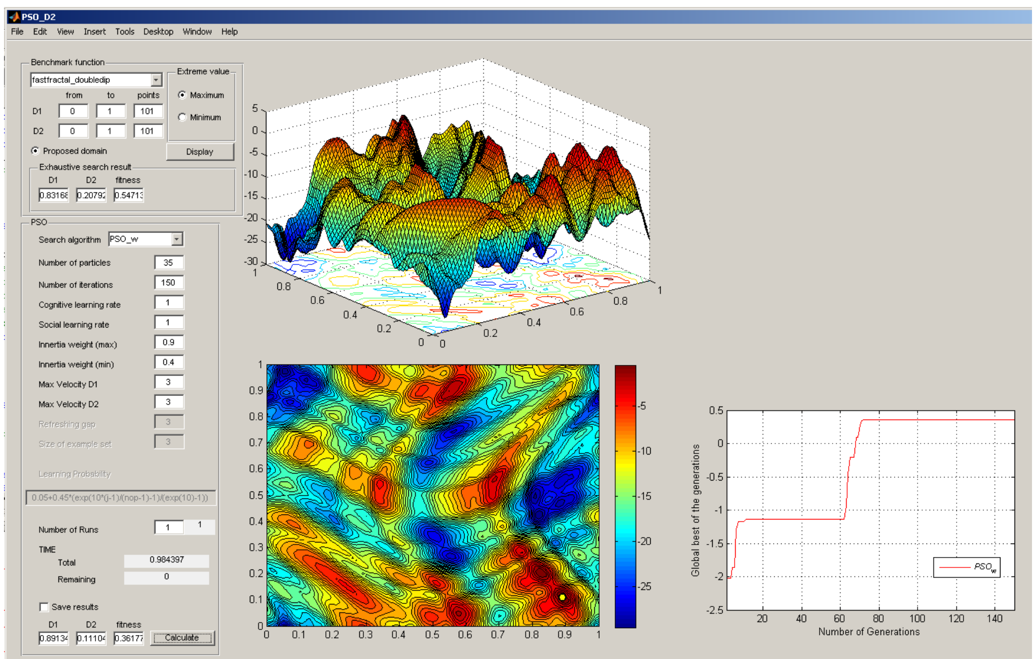


Implementation of PSO for power plants optimization problems II

- They are created to achieve a more sufficient and more effective service
  - It require a more specified and detailed modeling of the equipments (off-design case) of the operating power plant.
  - The accuracy of the provided solution of the search algorithm heavily depends on the thermodynamic model (search space) created in the power plant simulation software.
  - To decide the number of dimensions of the power plants optimization problem the number of degrees of freedom shall be determined which refers to the independent process variables of the off-design model, having impact on operating conditions.

Which PSO shall be chosen?

- How does a power plant search space look like and what propetries does it have?
- How can the convergence problems within the power plant simulation program be solved?



References I.

- H. Moran and H. Shapiro: Fundamentals of Engineering Thermodynamics (fifth ed.), John Wiley & Sons, New York (2003).
- Singiresu S. Rao, Engineering Optimization: Theory and Practice, (fourth ed.), John Wiley & Sons, Inc. (2009).
- Han Huang, Hu Qin, Zhifeng Hao, Andrew Lim Example-based learning particle swarm optimization for continuous optimization, Information Sciences (2010) doi:10.1016/j.ins.2010.10.018.
- Christoph Koch, Frank Cziesla, George Tsatsaronis: Optimization of combined cycle power plants using evolutionary algorithms, Chemical Engineering and Processing 46 (2007) 1151–1159
- Russell Eberhart, James Kennedy: A New Optimizer Using Particle Swarm Theory, Micro Machine and Human Science, Proceedings of the Sixth International Symposium (1995).
- J. J. Liang, A. K. Qin, Ponnuthurai Nagaratnam Suganthan, S. Baskar: Comprehensive Learning Particle Swarm Optimizer for Global Optimization of Multimodal Functions IEEE Transactions on Evolutionary computation, Vol. 10, No. 3, (2006)
- Singiresu S. Rao: Engineering Optimization: Theory and Practice, John Wiley & Sons, Inc. Fourth Edition (2009)
- Shi, Y. H., Eberhart, R. C.: A Modified Particle Swarm Optimizer, IEEE International Conference on Evolutionary Computation, Anchorage, Alaska, May 4-9, 1998.
- Eberhart, R. C., Shi, Y.: Particle Swarm Optimization: Developments, Applications and Resources, IEEE International Conference on Evolutionary Computation, Seoul, South Korea, (2001).
- Clerc, M., Kennedy, J., "The particle swarm-explosion, stability, and convergence in a multidimensional complex space," IEEE Transactions on Evolutionary Computation, vol. 6, no. 1, pp. 58-73, 2002.
- Kennedy, J. Small worlds and mega-minds: effects of neighborhood topology on particle swarm performance. Proceedings of IEEE Congress on Evolutionary Computation (CEC 1999), Piscataway, NJ. pp. 1931-1938, 1999

References II.

- Kennedy, J. and Mendes, R. Population structure and particle swarm performance. Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2002), Honolulu, Hawaii USA. 2002
- K. E. Parsopoulos and M. N. Vrahatis, “UPSO—A unified particle swarm optimization scheme,” in Lecture Series on Computational Sciences, 2004, pp. 868–873.
- R. Mendes, J. Kennedy, and J. Neves, “The fully informed particle swarm: Simpler, maybe better,” IEEE Trans. Evol. Comput., vol. 8, pp. 204–210, Jun. 2004.
- M. Lovbjerg and T. Krink, “Extending particle swarm optimizers with self-organized criticality,” in Proc. Congr. Evol. Comput., Honolulu, HI, 2002, pp. 1588–1593.
- T. M. Blackwell and P. J. Bentley, “Don’t push me! Collision-avoiding swarms,” in Proc. IEEE Congr. Evol. Comput., Honolulu, HI, 2002, pp. 1691–1696.
- M. Lovbjerg, T. K. Rasmussen, and T. Krink, “Hybrid particle swarm optimizer with breeding and subpopulations,” in Proc. Genetic Evol. Comput. Conf., 2001, pp. 469–476.
- X. Xie, W. Zhang, and Z. Yang, “A dissipative particle swarm optimization,” in Proc. Congr. Evol. Comput., Honolulu, HI, 2002, pp. 1456–1461.
- K. E. Parsopoulos and M. N. Vrahatis, “On the computation of all global minimizers through particle swarm optimization,” IEEE Trans. Evol. Comput., vol. 8, pp. 211–224, Jun. 2004.
- F. van den Bergh and A. P. Engelbrecht, “A cooperative approach to particle swarm optimization,” IEEE Trans. Evol. Comput., vol. 8, pp. 225–239, Jun. 2004.
- Yu Wang, Bin Li, Thomas Weise, Jianyu Wang, Bo Yuan, Qiongjie Tian, “Self-adaptive learning based particle swarm optimization”, Inform. Sci. (2010), doi:10.1016/j.ins.2010.07.013
- Swarm Intelligence Group (Peking University): PSO and its variants, PowerPoint Presentation
- Softwares: Matlab; CycleTempo

Thank you for your attention!