Swarm Intelligence: Charged System Search

Intelligent Robotics Seminar

Alireza Mollaalizadeh Bahnemiri

15. December 2014

Content

- What is Swarm Intelligence?
- Charged System Search
- Improved Charged System Search
- Experiments
- Conclusion
- Questions

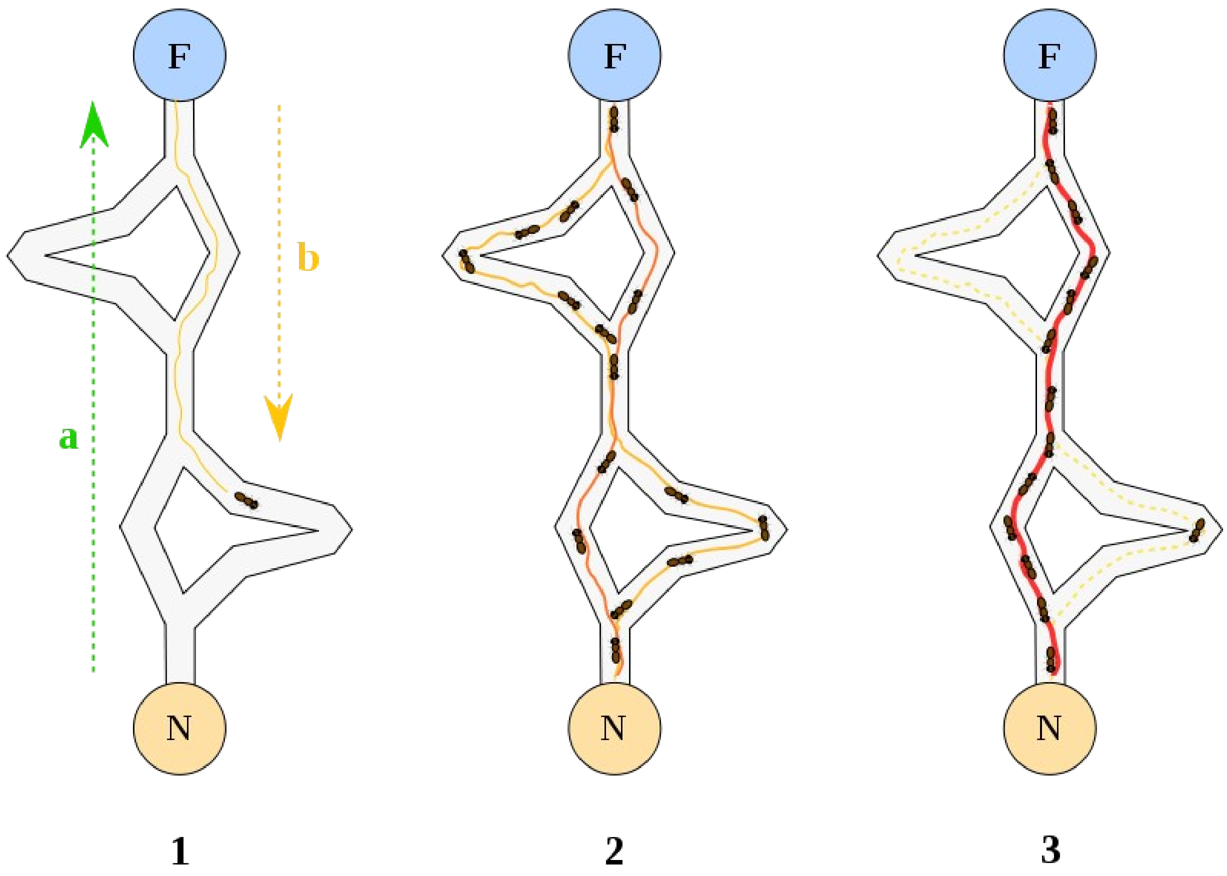
Swarm Intelligence (SI)

- SI is the collective behavior of decentralized, self-organized systems. *(by Gerardo Beni and Jing Wang in 1989, in the context of cellular robotic systems.)*
- Rules:
  - Following simple local rules by each agent
  - Decentralized behavior of each agent
  - Local interaction of agents with each other the environment (which cause complex behavior)

Samples

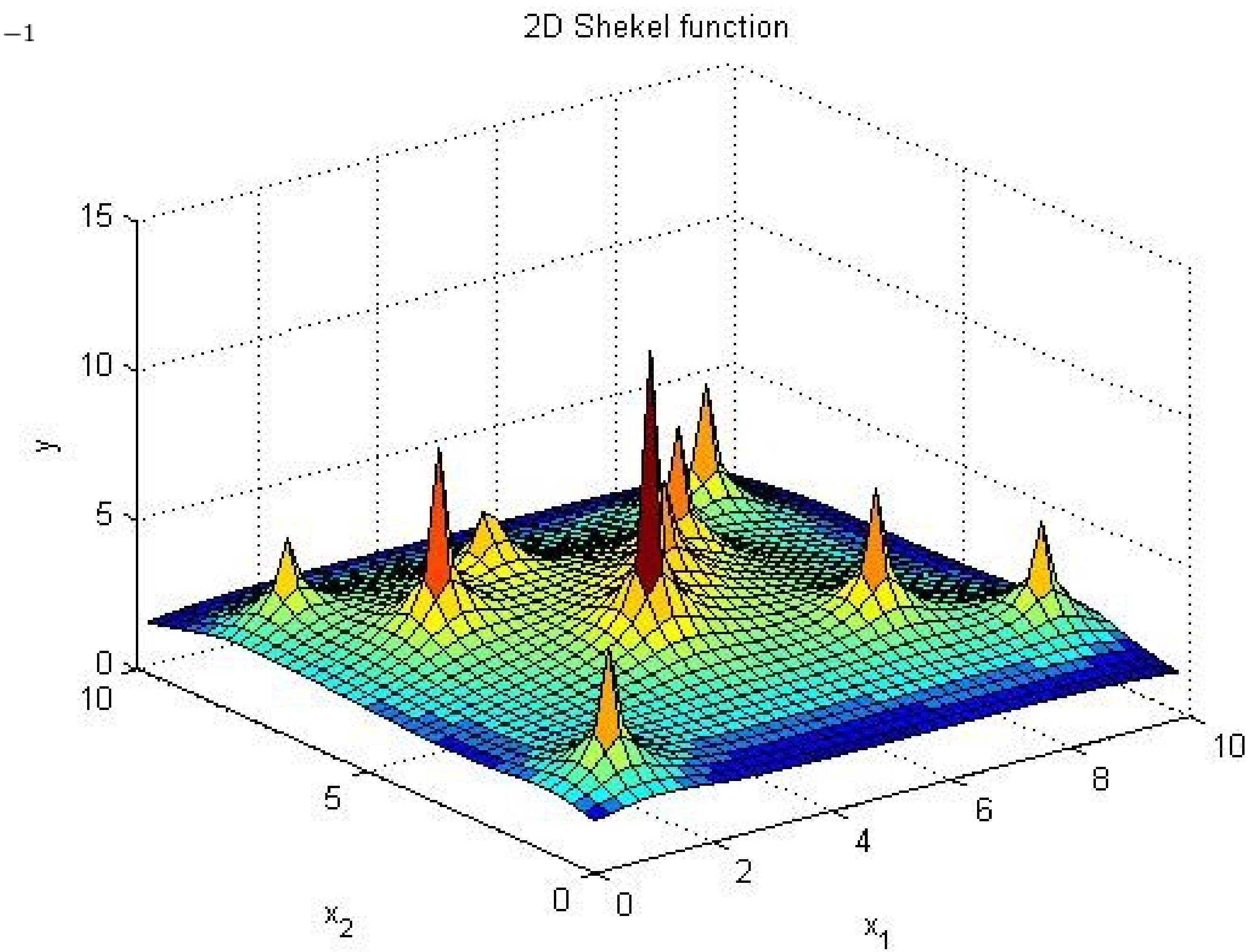
- Ant Colony Optimization
- Artificial Bee Colony Algorithm
- Artificial Immune Systems
- River Formation Dynamics
- Particle Swarm Optimization
- Charged System Search
- …

Ant's Behavior



Multi-modal Function

$$f(\vec{x}) = \sum_{i=1}^{m} \left(c_i + \sum_{j=1}^{n} (x_j - a_{ji})^2 \right)^{-1}$$



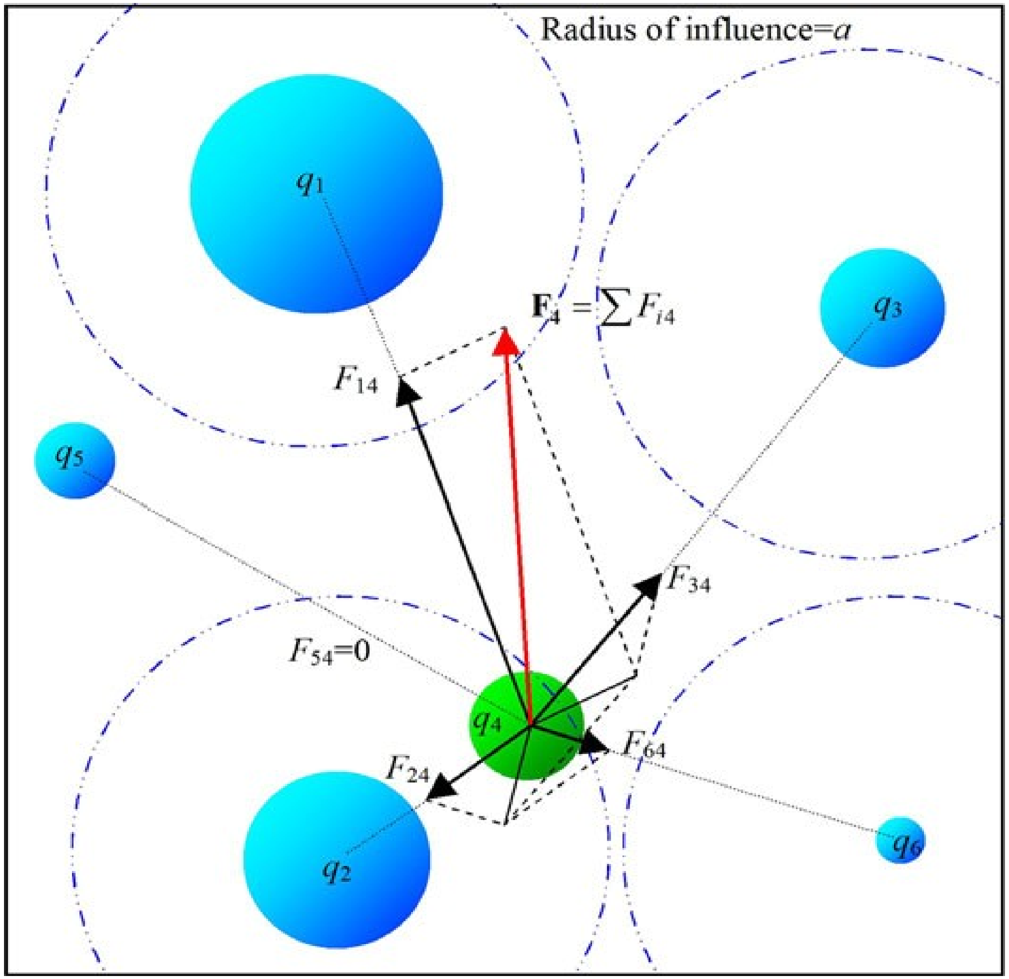
[http://en.wikipedia.org/wiki/Shekel_function]

Charged System Search (CSS)

- Optimization algorithm based on some principles from **physics** and **mechanics**

- **Coulomb law** from electrostatics and the **Newtonian laws** of mechanics

Charged System



[A. Kaveh · S. Talatahari, *A novel heuristic optimization method: charged system search,* Springer-Verlag 2010]

CSS Parameters (I)

- Charged Particles (CPs)

$$x_{i,j}^{(o)} = x_{i,\min} + \text{rand} \cdot (x_{i,\max} - x_{i,\min}), \quad i = 1, 2, \ldots, n, \tag{1}$$

$$v_{i,j}^{(o)} = 0, \quad i = 1, 2, \ldots, n. \tag{2}$$

- Magnitude of Charge (*q(i)*)

$$q_i = \frac{\text{fit}(i) - \text{fitworst}}{\text{fitbest} - \text{fitworst}}, \quad i = 1, 2, \ldots, N, \tag{3}$$

CSS Parameters (II)

- Distance Between CPs

$$r_{ij} = \frac{\left\| \mathbf{X}_i - \mathbf{X}_j \right\|}{\left\| (\mathbf{X}_i + \mathbf{X}_j)/2 - \mathbf{X}_{best} \right\| + \varepsilon}, \quad (4)$$

- Attraction Probability

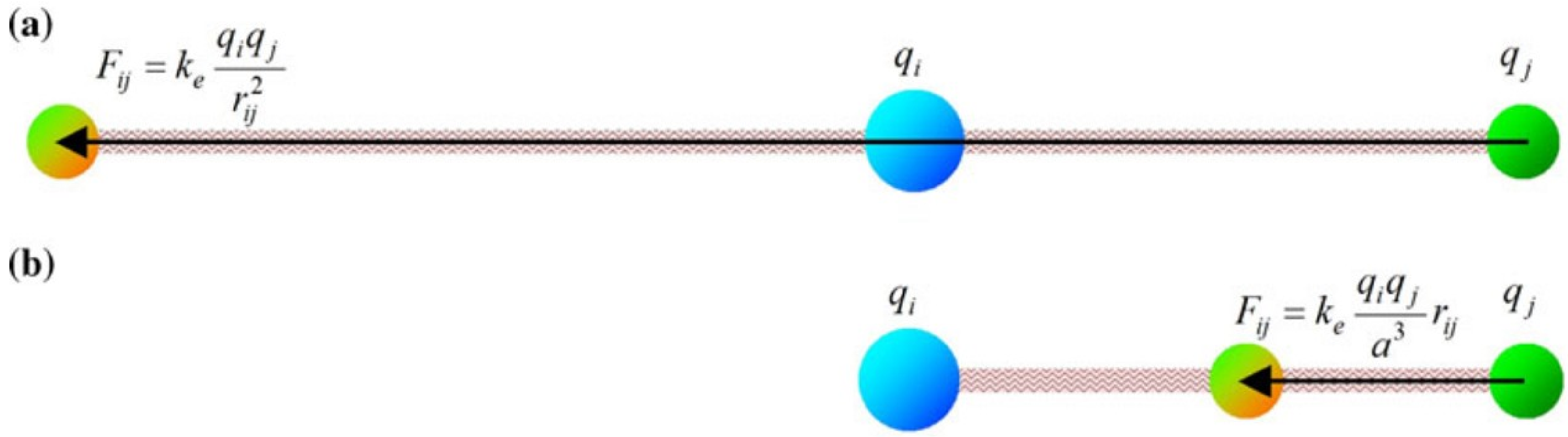
$$p_{ij} = \begin{cases} 1 & \text{fit}(j) > \text{fit}(i), \\ 0 & \text{else.} \end{cases} \quad (5)$$

- Extended Attraction Probability

$$p_{ij} = \begin{cases} 1 & \frac{\text{fit}(i) - \text{fitbest}}{\text{fit}(j) - \text{fit}(i)} > \text{rand} \vee \text{fit}(j) > \text{fit}(i), \\ 0 & \text{else.} \end{cases} \quad (6)$$

Force Calculation

- Coulomb's Law:



$$\mathbf{F}_j = q_j \sum_{i,i \neq j} \left(\frac{q_i}{a^3} r_{ij} \cdot i_1 + \frac{q_i}{r_{ij}^2} \cdot i_2 \right) p_{ij} (\mathbf{X}_i - \mathbf{X}_j), \quad \begin{cases} j = 1, 2, \ldots, N, \\ i_1 = 1, i_2 = 0 \Leftrightarrow r_{ij} < a, \\ i_1 = 0, i_2 = 1 \Leftrightarrow r_{ij} \geq a, \end{cases} \tag{7}$$

Velocity and Position

- Update Position

$$\mathbf{X}_{j,\text{new}} = \text{rand}_{j1} \cdot k_a \cdot \frac{\mathbf{F}_j}{m_j} \cdot \Delta t^2 + \text{rand}_{j2} \cdot k_v \cdot \mathbf{V}_{j,\text{old}} \cdot \Delta t + \mathbf{X}_{j,\text{old}}, \quad (8)$$

$$k_v = 0.5(1 - \text{iter}/\text{iter}_{\max}), \quad k_a = 0.5(1 + \text{iter}/\text{iter}_{\max}), \quad (9)$$

- Update Velocity

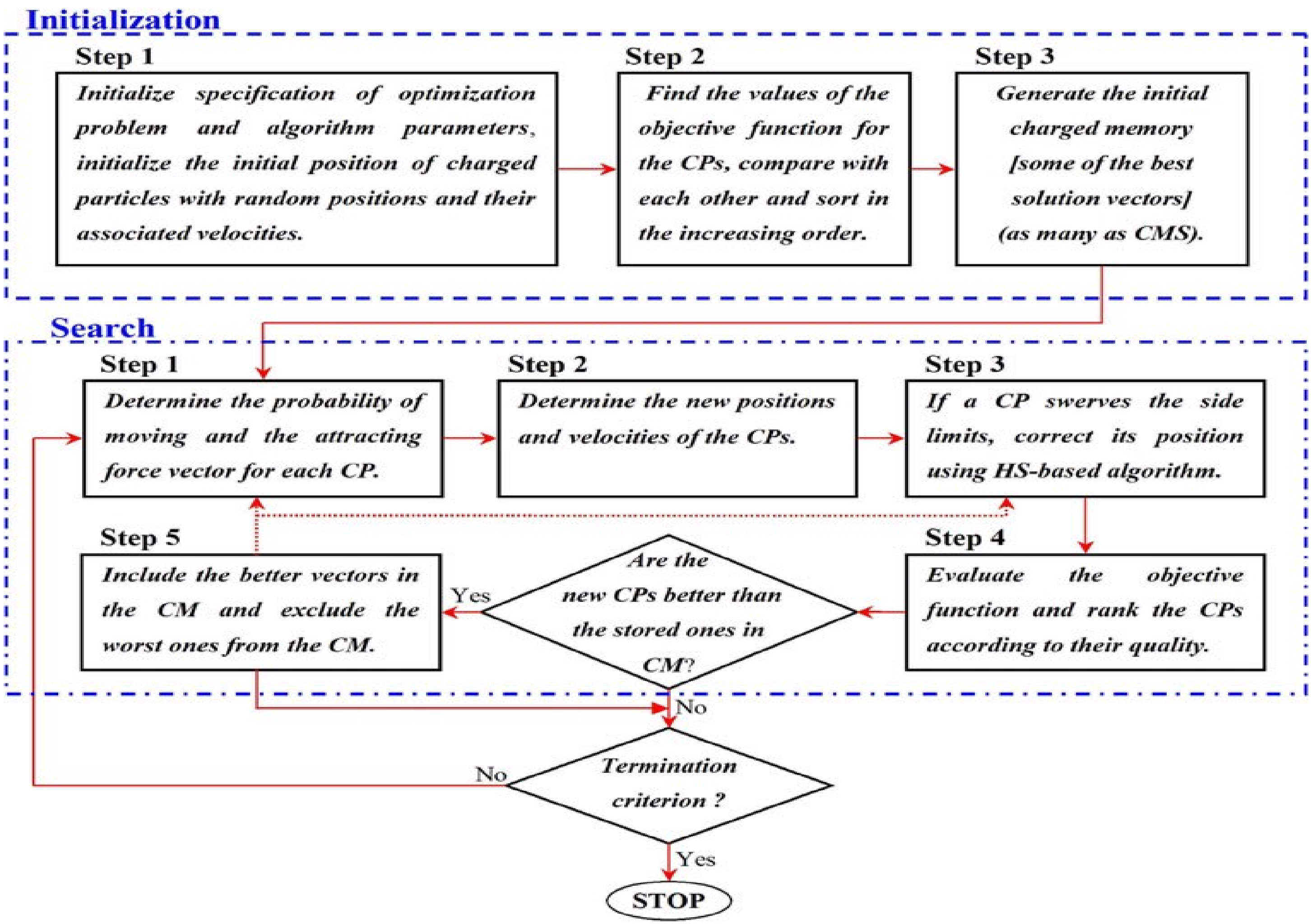
$$\mathbf{V}_{j,\text{new}} = \frac{\mathbf{X}_{j,\text{new}} - \mathbf{X}_{j,\text{old}}}{\Delta t}, \quad (10)$$

CSS Parameters (III)

- Charged Memory
  - The best particles for influencing other ones
  - Controls Exploitation

- Termination Criterion
  - After *n* steps (performance representation)
  - Reaching a threshold

CSS Algorithm



Improved CSS

How to Improve CSS ?

- Repulsive Force
- Artificial Bee Colony
- Bayesian Charged System Search

Artificial Bee Colony (ABC)

- Each Employed Bee (EB), tries to improve its position
- The movement is performed only in one dimension:

$$\theta_i^* = \theta_i + rand \times (\theta_i - \theta_k)$$

- Reform a new set of EB's with roulette wheel selection mechanism

Bayesian Optimization Algorithm (BOA)

1) Let t=0

2) Randomly generate initial population P(0) of size n

3) Evaluate the population

4) Select a set of promising solutions from P(t) with a selection method

5) Learn a Bayesian network B using the selected individuals

6) Generate a new population O(t) according to the joint distribution encoded by B

7) Evaluate the solutions in O(t)

8) Create a new population P(t + 1) by replacing all or some individuals from P(t) with O(t)

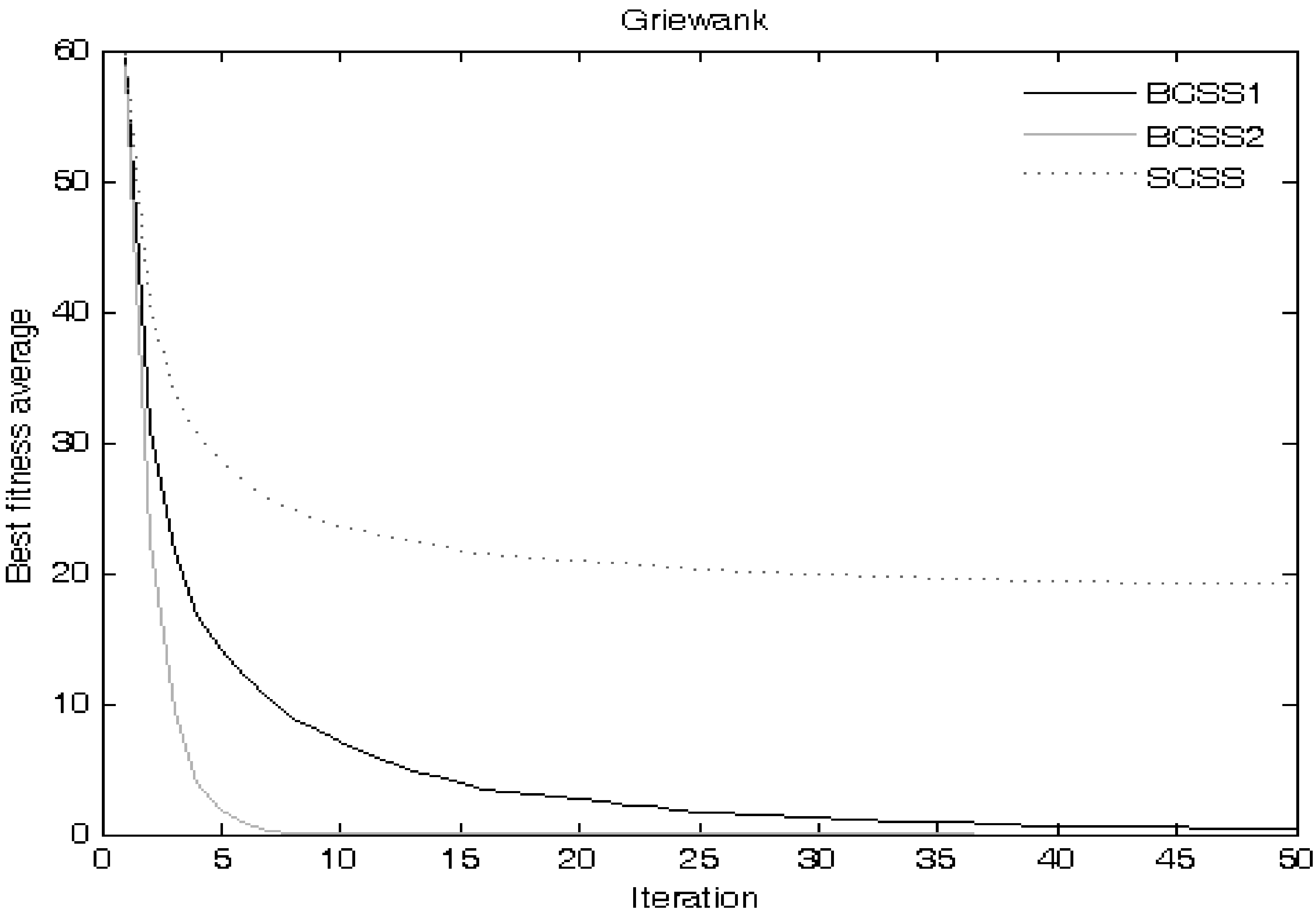
9) Let t=t + 1

10) If the termination criteria are not met, go to (4)

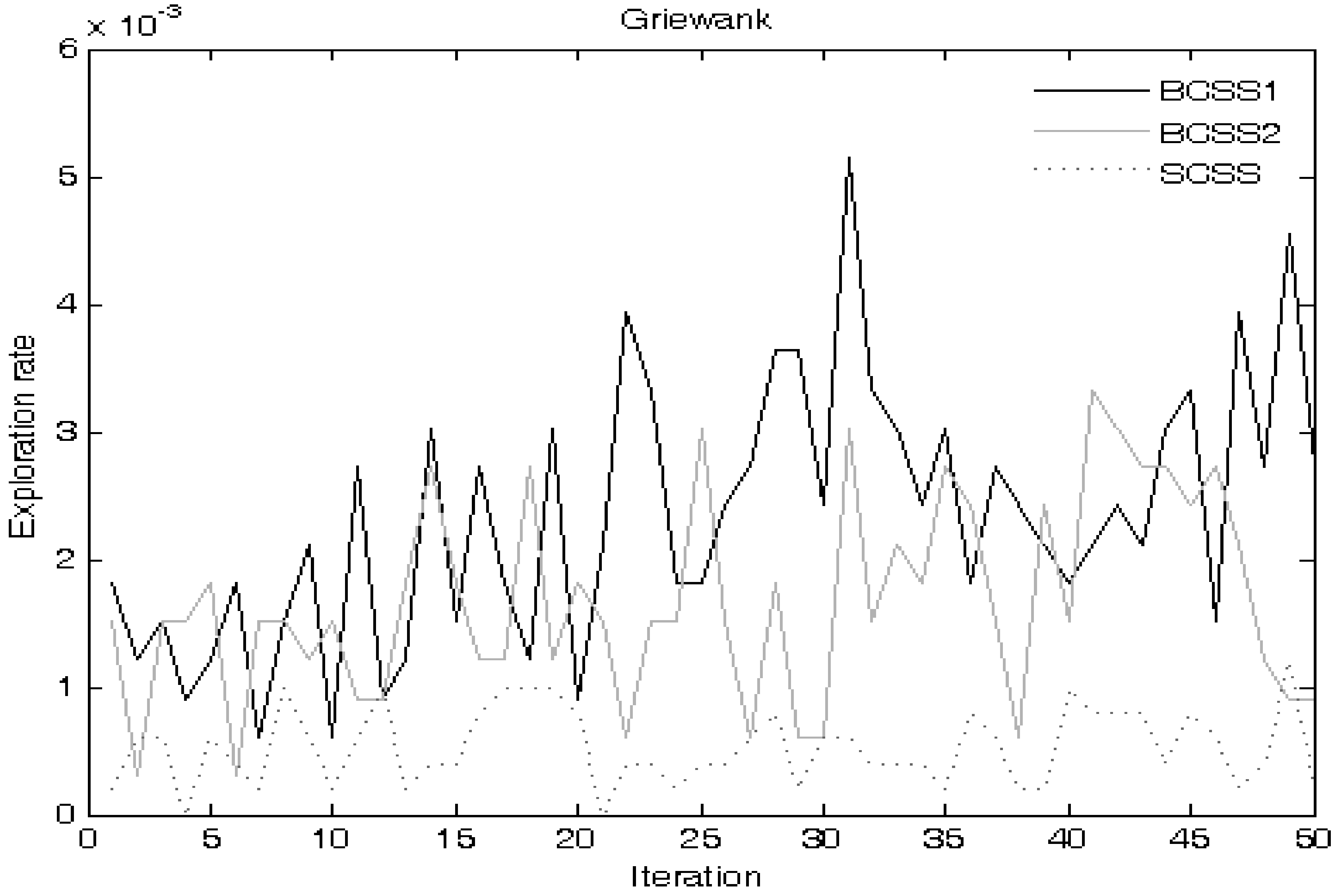
Bayesian CSS

- Hybrid method with BOA and CSS
- Enhance in Exploitation

Experiments (I)



Experiments (II)



Conclusion

- Swarm Intelligence
- SI can be efficient way for dealing with Multi-modal and Stochastic functions
- Charged Systems Search
- CSS can be extended in Exploration by applying ABC
- BOA is useful for increase the Exploitation in CSS

References

[1] Kennedy, J. E. (1995). Particle swarm optimization. *Proceedings of ICNN 1995 - IEEE International Conference on Neural Networks*, (pp. 1942–1948). Perth.

[2] Kaveh, A., & Talatahari, S. (2010). A Novel Heuristic Optimization Method: Charged System Search. *Acta Mechanica* , 213(3-4), 267–289.

[3] Karaboga, D., & Basturk, B. (2007). A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *Journal of Global Optimization*, 39, 459–471.

[4] Neapolitan, R. E. (2003). Learning Bayesian Networks. *Prentice Hall* .

Thanks for your attention!
Questions ?