A PSO-Based Mobile Robot for Odor Source Localization in Dynamic Advection-Diffusion with Obstacles Environment: Theory, Simulation and Measurement

Wisnu Jatmiko,Kosuke Sekiyama and Toshio Fukuda

IEEE Computational Intelligence Magazine - 2007 (JCR: 2.833)

February 8, 2012

Paper outline

1. Introduction/Motivation
2. PSO framework
   1. Standard PSO
   2. Proposal #1: Detection and Responding PSO (DR-PSO)
   3. Proposal #2: Charged PSO (C-PSO)
3. Implementation Framework
   1. Environment
   2. Robot behavior
   3. Obstacle-free experiments
   4. Obstacle-filled experiments
4. Extension with Wind
   1. Odor-gated rheotaxis (OGR): particles use not only odor particle concentration but also wind direction
   2. Implementation I: used forbidden area
   3. Implementation II: x_θ parameter
5. Conclusions

Goal

- Odor source localization using a robot swarm - each robot is modeled as a particle
- Standard PSO is not appropriate to this problem because it doesn't react to changing environments

Standard PSO

- Standard PSO equations:

$$\begin{aligned}\mathbf{V}_i(t) =& \chi(\mathbf{V}_i(t-1) + c_1\, rand()(\mathbf{p}_i(t-1) - \mathbf{x}_i(t-1)) \\ & + c_2\, rand()(\mathbf{p}_g(t-1) - \mathbf{x}_i(t-1))) \qquad (1) \\ \mathbf{x}_i(t) =& \mathbf{x}_i(t-1) + \mathbf{V}_i(t) \qquad (2)\end{aligned}$$
,

where $\chi$is a constriction factor, c_1 and c_2 are learning parameters, such that $c_1 = c_2 = 2$ (????). p_i represents the best local data and p_g the best global (assumed to be shared among robots/particles)

PSO: Non-stationarity problem

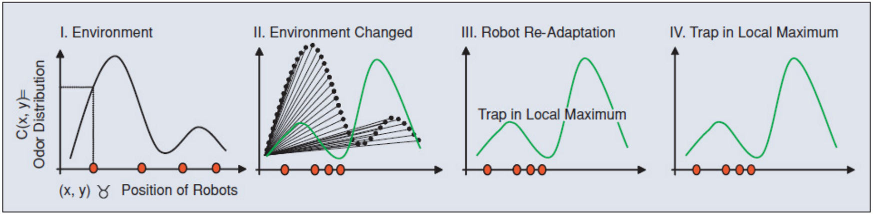


Figure: PSO and changing environments

Detection and Responding PSO I

- Standard PSO but, whenever a change is detected, particles spread randomly for a fixed amount of time

Detection and Responding PSO II

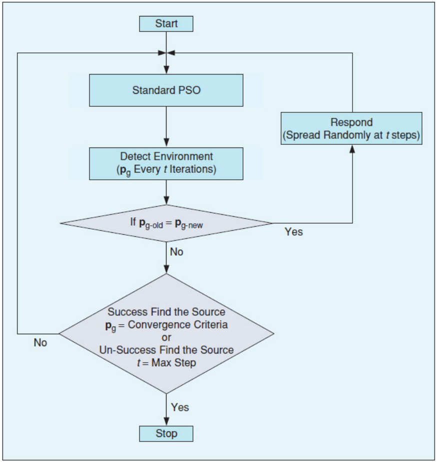


Figure: DR-PSO response to changes in environment

Detection and Responding PSO III

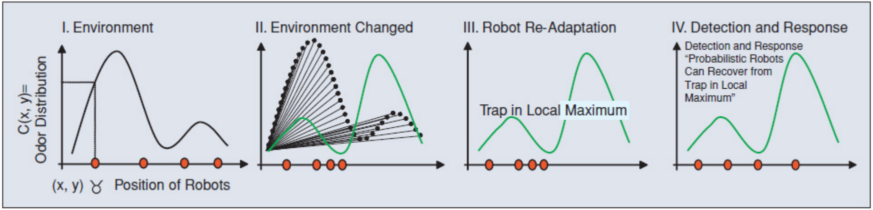


Figure: DR-PSO response to changes in environment

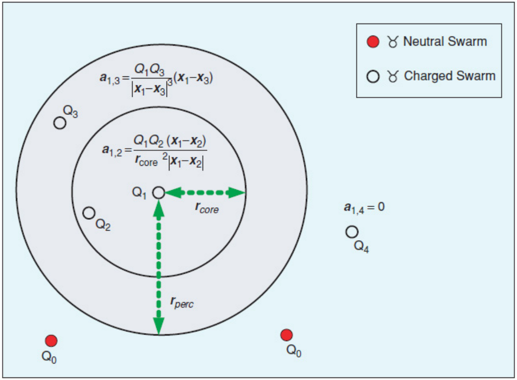
Charged PSO I

- Coulomb law is added to the particle dynamic model to keep diversity of the position of the particles
- Some particles are charged, some are not

$$\mathbf{a}_{ip} = \begin{cases} \frac{Q_i \cdot Q_p (\mathbf{x}_i - \mathbf{x}_p)}{r_{core}^2 |\mathbf{x}_i - \mathbf{x}_p|} & |\mathbf{x}_i - \mathbf{x}_p| < r_{core} \\ \frac{Q_i \cdot Q_p}{|\mathbf{x}_i - \mathbf{x}_p|^3} (\mathbf{x}_i - \mathbf{x}_p) & r_{core} < |\mathbf{x}_i - \mathbf{x}_p| < r_{perc} \\ 0 & r_{perc} < |\mathbf{x}_i - \mathbf{x}_p| \end{cases} \quad \mathbf{a}_i(t) = \sum_{p \neq i}^{N} \mathbf{a}_{ip}$$

$$\mathbf{V}_i(t) = \chi(\mathbf{V}_i(t-1) + c_1\, rand()(\mathbf{p}_i(t-1) - \mathbf{x}_i(t-1)) + c_2 Rand()(\mathbf{p}_g(t-1) - \mathbf{x}_i(t-1))) + \mathbf{a}_i(t) \quad (5)$$

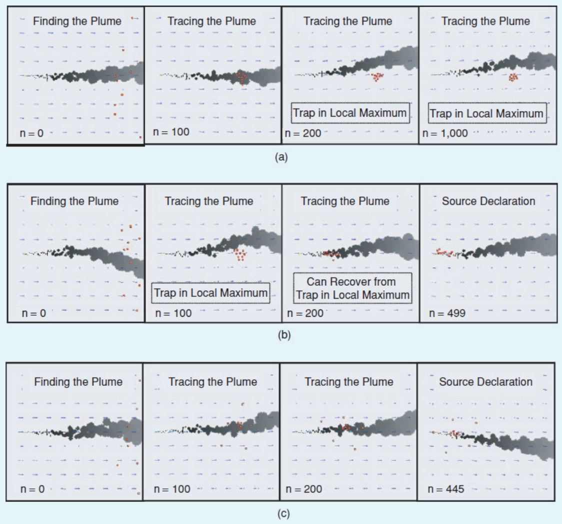
Charged PSO II



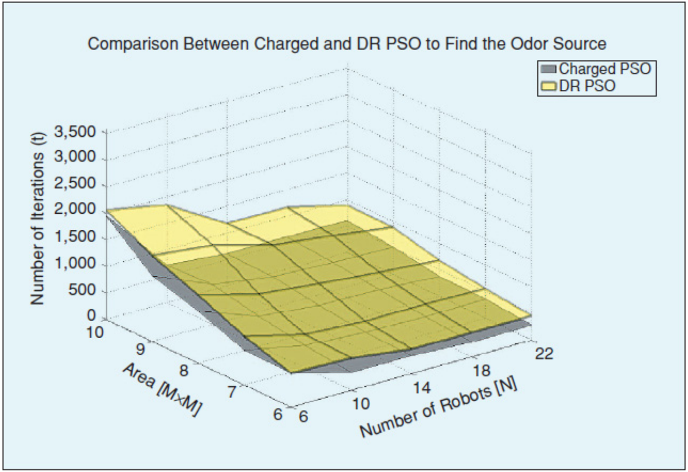
Common settings

- Robots first search for a trace of odor, then PSO drive them toward the source and finally the source is declared
- The Advection-Diffusion odor model is used (Farrell et al.) to generate a dynamic plume
- Odor and position sensors model random noise

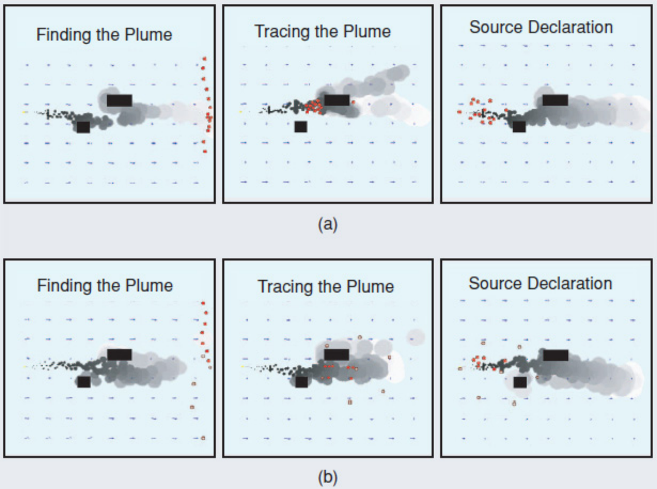
Obstacle-free I



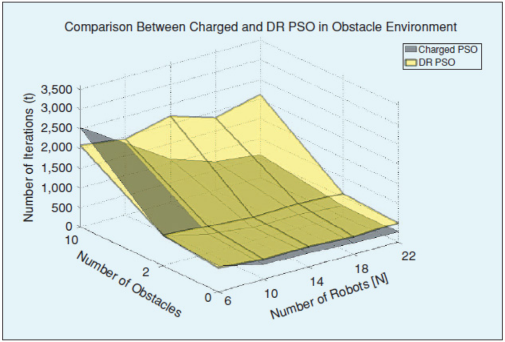
Obstacle-free II



Obstacle-filled I



Obstacle-filled II



Odor-gated Rheotaxis (OGR)

- Robots are able to perceive odor-particle concentration and wind direction
- Bio-inspired technique

Implementation I: Use forbidden area

$$\mathbf{V}_i^*(t) = \chi(\mathbf{V}_i(t-1) + c_1\, rand()(\mathbf{p}_i(t-1) - \mathbf{x}_i(t-1)) + c_2\, rand()(\mathbf{p}_g(t-1) - \mathbf{x}_i(t-1))) \quad (21)$$

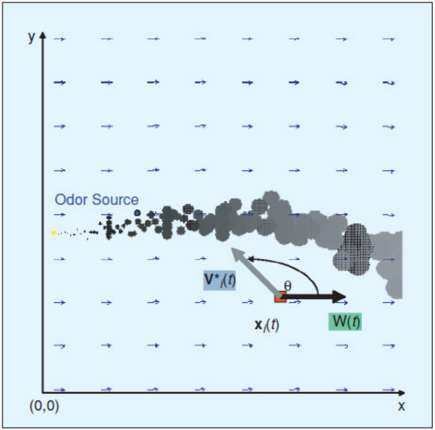
$$\mathbf{V}_i(t) = \begin{cases} 0 & \text{if } \theta < |\theta_{\text{forbidden}}| \\ \mathbf{V}_i^*(t) & \text{Otherwise} \end{cases} \quad (22)$$

$$\mathbf{x}_i(t) = \mathbf{x}_i(t-1) + \mathbf{V}_i(t) \quad (23)$$

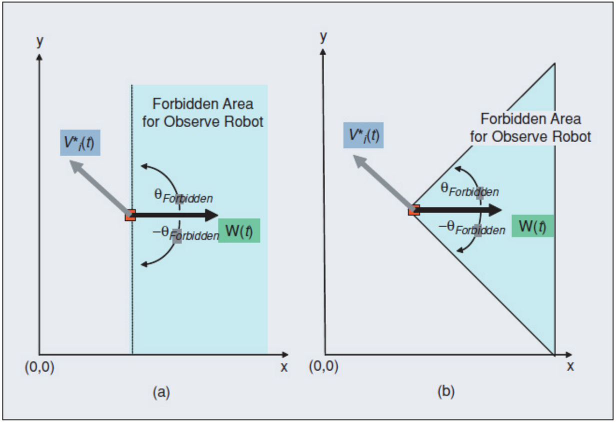
,

where θ is the angle between the particle's current velocity and wind direction

Implementation I: diagram I



Implementation I: diagram II



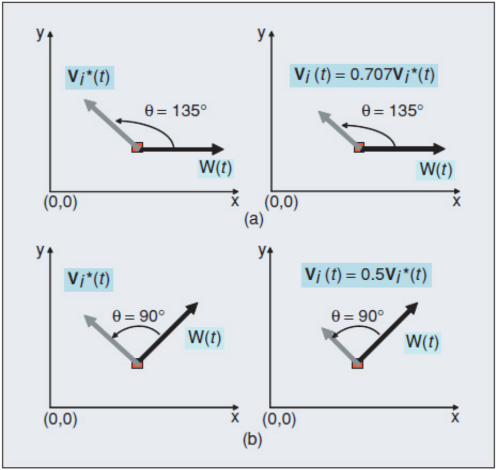
Implementation II: χ_θ parameter

- Instead of avoiding movements within forbidden area, velocities are weigthed with a new parameter: χ_θ

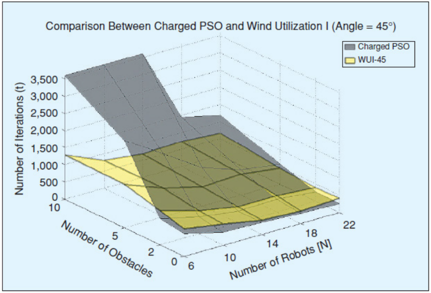
$$\mathbf{V}_i^*(t) = \chi(\mathbf{V}_i(t-1) + c_1\, rand()(\mathbf{p}_i(t-1) - \mathbf{x}_i(t-1)) + c_2 Rand()(\mathbf{p}_g(t-1) - \mathbf{x}_i(t-1))) \quad (25)$$

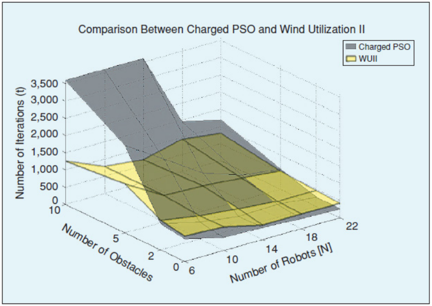
$$\mathbf{V}_i(t) = \chi_\theta \mathbf{V}_i^*(t) \quad (26)$$

Implementation II: diagram



C-PSO vs. Wind-use Implementation I & II





Wind-use Implementation I vs. II

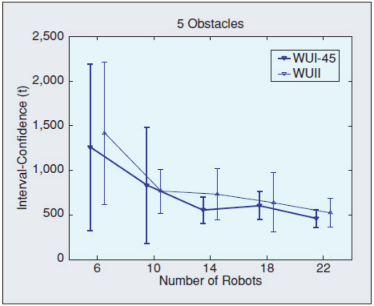


FIGURE 21 Performance of the WU-I and WU II algorithms in a five-obstacle environment.

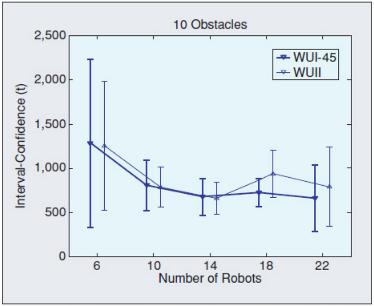


FIGURE 22 Performance of the WU-I and WU II algorithms in a ten-obstacle environment.