ICIC Express Letters
Volume **4**, Number **6(A)**, December **2010**

ICIC International ©2010 ISSN 1881-803X
pp. **2207-2212**

A PARTICLE SWARM OPTIMIZATION ALGORITHM WITH EQUILIBRIOUS DISTRIBUTION PARAMETER FOR GLOBAL PATH PLANNING OF AUV

Wei Zu, Hongxing Chang, Guoliang Fan and Jianqiang Yi

Institute of Automation
Chinese Academy of Sciences
Beijing 100190, P. R. China
wei.zu@ia.ac.cn

Received March 2010, accepted May 2010

Abstract. *This paper presents a novel method for solving global path planning problem of AUV using a modified particle swarm optimization (PSO). Firstly, a polar coordination model of the obstacle environment is established, and a candidate path representation method is adopted. Secondly, we give an equilibrious distribution parameter and propose a modified PSO evolutionary strategy which can avoid particles clustering within a sub-area of the problem scope. Thirdly, an AUV global path planning algorithm based on EDPSO is designed. Simulation experiments indicate the algorithm is effective and can provide a safe path in the sea field environment.*
Keywords: Particle swarm optimization, Global path planning, Autonomous underwater vehicle

1. Introduction.

Planning a collision-free path is a fundamental issue for an autonomous underwater vehicle (AUV) to execute its tasks. The goal of path planning is to generate a collision-free trajectory for an AUV to move from an initial configuration to a goal configuration. There are many methods suggested by researchers to solve this problem. Most of Classic path planning approaches [1-3], such as cell decomposition, road map and potential field, have some weakness in common. Particle Swarm Optimization (PSO), firstly presented by Kennedy and Eberhart [4,5] in 1995, is a kind of heuristic and random-search algorithm where particles collaborate as a population to reach a collective goal. There have been some proposed solutions for obstacle avoidance with PSO [6-8].

Compared with other evolutionary technology such as GA, PSO has many advantages, such as fewer control parameters and quick convergence, while it has some shortcomings. One shortcoming is that candidate particles will cluster within a sub-area of the whole solution scope. It will result in local optimization results. In this paper, a novel method for solving global path planning problem of AUV using a modified PSO with equilibrious distribution parameter (EDPSO) is proposed. The parameter can measure the diversity of candidate particles and guarantee the escaping from the sub-optimum trap.

2. Environment Modeling and Problem Description.

2.1. Polar coordination model.

In this paper, we build the obstacle environment under the polar coordinate using electronic chart data which is stored in the grid environment and shown in Figure 1. The proposed polar coordination model is described in Figure 2: Take the starting point S as the origin, and the line SG as the polar axis. Divide line SG into n equal segments with $n-1$ points, and further draw $n-1$ circles which have the same origin S. Take random points P_i on every circle and construct a path: $Path = \{S, P_1, \ldots, P_i, P_{n-1}, G\}$.

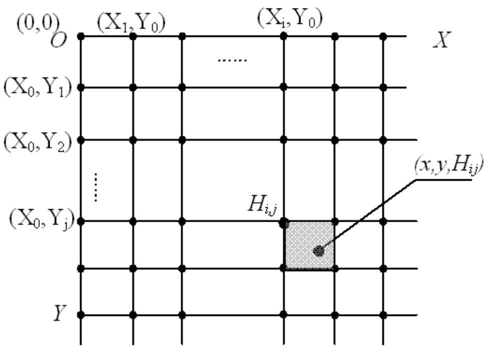

FIGURE 1. Electronic chart data model

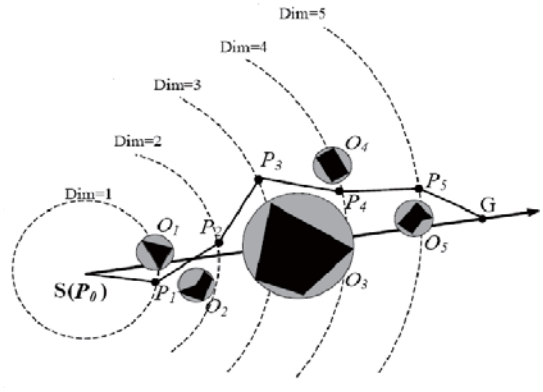

FIGURE 2. Polar coordination environment model

The ith point P_i (R_i, α_i) on a candidate path can be described through mathematical conversion as follows:

$$\begin{cases} \mathrm{R}_i = \sqrt{x'^2_i + y'^2_i} \\ \alpha_i = a\tan(y'/\mathrm{R}_i) \end{cases} \tag{1}$$

where x'_i, y'_i is latitude and longitude of P'_i in S-$X'Y'$ coordinate which is obtained though O-XY coordinate rotation. Two different coordinates S-$X'Y'$ and O-XY conversion relation is described as follows:

$$\begin{bmatrix} x'_i \\ y'_i \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} + \begin{bmatrix} -x_s\cos\theta - y_s\sin\theta \\ x_s\sin\theta - y_s\cos\theta \end{bmatrix} \tag{2}$$

2.2. **A candidate path representation.** In our algorithm, a candidate path is represented by a sequence of path waypoints as shown in Figure 3, in which R_i is polar distance of ith waypoint, α_i is its polar angle, and H_i is its water depth data.

$$\begin{cases} \alpha_i = \sum_{j=0}^{i} \Delta\alpha_j \\ R_i = i * R_{basic} \end{cases} \tag{3}$$

where $\Delta\alpha_i$ is the turning angle of ith waypoint on the path, and R_{basic} is the first dimension circle radius. The particle representation can be simplified into $P = \{\Delta\alpha_0, H_0, \Delta\alpha_1, H_1, \ldots, \Delta\alpha_i, H_i, \Delta\alpha_n, H_n\}$.

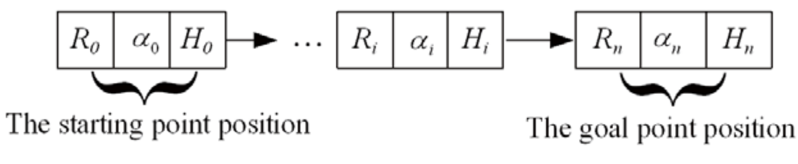

FIGURE 3. The representation of a particle

3. A PSO Algorithm with Equilibrious Distribution Parameter (EDPSO).

3.1. **Traditional PSO.** Each particle is treated as a point in the n-dimensional problem space. A particle represents a candidate solution to the problem. The particle is represented as $X_i = (x_{i0}, x_{i1}, \ldots, x_{in-1})$, $i = 1, 2, \ldots, PNum$, where $PNum$ is the swarm size and n is the total dimension number of each particle. Each particle adjusts its trajectory toward its own previous best position $pBest$ and the previous best position $gBest$ attained by the whole swarm [4]. The particles are manipulated according to the following equations:

$$v_{id} = w * v_{id} + c_1 * r_1 * (p_{id} - x_{id}) + c_2 * r_2 * (p_{gd} - x_{id}) \tag{4}$$

$$x_{id} = x_{id} + v_{id} \tag{5}$$

where c_1 and c_2 are acceleration constants, r_1 and r_2 are random numbers within the interval of $[0, 1]$. Changing velocity this way enables each particle to search around its individual best position and global best position.

3.2. **Modified PSO evolutionary strategy.** In order to effectively avoid particles clustering within a sub-area of the problem scope, we define a new parameter which can measure the swarm particles equilibrium of distribution degree. In addition, a new particle evolutionary strategy is presented which can direct the particle rational flying behavior.

1) *Equilibrious Distribution Parameter.* The diversity measure method proposed by Riget [9] adopt a parameter "particle-dimension-distance" to measure the distance between different particles, but the method can not measure the distance between dimension vectors of particles. A new parameter "particle-distribution-degree" which can improve the sufficiency of PSO diversity measure in evolutionary process is given here.

Definition 3.1. ***Equilibrious Distribution Parameter.*** *"particle-distribution-degree"*

$$dis(S) = \frac{1}{Dim} \cdot \sum_{i=0}^{Dim} \sqrt{\sum_{l=1}^{N} \left(\frac{PNum}{N} - a_{il}\right)^2} \tag{6}$$

where Dim is the dimensionality of the problem, N is the equal separation size of the particle swarm, $PNum$ is the swarm size. a_{il} is the sum of dimension vectors in ith dimension and lth separation area.

a) If particles distribute equally in problem scope, the value $dis(s)$ will be zero.

b) If particles cluster in the same dimension area, $dis(s)$ will satisfy Equation (7).

$$dis(S) = \frac{1}{Dim} \cdot \sum_{i=0}^{Dim} \sqrt{\left(\frac{PNum}{S} - PNum\right)^2 + \sum_{l=1}^{S-1} \left(\frac{PNum}{S}\right)^2} = PNum\sqrt{1 - \frac{1}{S}} \tag{7}$$

In addition, we adopt the parameter "particle-dimension-distance" suggested by Riget [9] to measure the diversity degree which is defined as follow:

$$diversity(S) = \frac{1}{|S|} \cdot \sum_{i=1}^{|S|} \sqrt{\sum_{j=1}^{Dim} (p_{ij} - \overline{p_j})^2} \tag{8}$$

where S is the swarm, $|S|$ is the swarm size, Dim is the dimensionality of the problem, p_{ij} is the jth value of the ith particle and $\overline{p_j}$ is the jth value of the average point $\overline{p}$.

2) *Particle Evolutionary Strategy.* At the base of above improvement, we propose a modified PSO evolutionary strategy in order to increase the algorithm calculation efficiency in this section. The flow of EDPSO Algorithm is shown as follows:

Function *Diversity_Measure*
Input: swarm S parameters and thresholds dis_Max, div_Low **Output:** $dis(S)$, $diversity(S)$

Steps:
1. Calculate $dis(S)$ using Equation(6);
2. Calculate $diversity(S)$ using Equation(7);
3. If $diversity(S) < dis\_Low$ and $dis(S) > dis\_Ma$
    satisfy diversity conditions and go to next step:
  else $diversity(S) > dis\_Low$, or $dis(S) < dis\_Ma$
    lost diversity, and recalculate velocity and position with Equations (4)-(5);
4. return to Step 2 in EDPSO.

Algorithm EDPSO

Input: set parameter $c_1 = c_2 = 1$, $w_{\max} = 0.95$, $w_{\min} = 0.2$, $pDim = 20$, $pNum = 30$;
Fitness function $fitness(X_i)$;

Output: Global particle $pBest$.

Steps:

1. Initialize particles $X_1, X_2, \ldots, X_{pNum}$,
2. For every particle X_i
   a. Calculate particle new velocity V_i^{t+1} using Equation (4);
   b. Calculate particle new position $estimated\_X_i^{t+1}$ using Equation (5);
   c. For every dimension X_{ij}
      if $fitness(X_{ij}^{t+1}, X_{ij}^t) > fitness(X_{ij}^{t+1})$
      and $fitness(X_{ij}^{t+1}, X_{ij}^t) > fitness(estimated\_X_i^{t+1})$
         $estimated\_X_i^{t+1}$ is not a better position, and $X_{ij}^{t+1} = X_{ij}^t$;
      else
         $estimated\_X_i^{t+1}$ will be adopted, and $X_{ij}^{t+1} = estimated\_X_i^{t+1}$
     end for dimension X_{ij}
   d. Calculate particle new position X_i^{t+1} using jth dimensional vector X_{ij}^{t+1} with Equation (5);
   e. run $Diversity\_Measure$ ( ) function;
  end every particle X_i

4. **AUV Global Path Planning Algorithm Using EDPSO.** This section describes an AUV path planning algorithm based on EDPSO. Firstly, the obstacle collision avoidance strategy is introduced.

1) ***Collision Avoidance Strategy.*** We can adjust the AUV heading to avoid collision of obstacles, and avoid entering some danger or forbidden areas. We define FZ_i as danger and forbidden areas in the ith dimension of a path which can be described in Figure 4 and Figure 5.

$$FZ_i = \sum_{i=0}^{obs\_num} Scope_{\theta_i} \tag{9}$$

where obs_num is the obstacle number in current dimension area, and $Scope_{\theta ij}$ is the angle scope of jth obstacle forbidden area. $Scope_{\theta ij}$ can be calculated as follow:

$$Scope_{\theta} = [orientation(P_i, O_{ij}) - \Delta\theta_{ij}, orientation(P_i, O_{ij}) + \Delta\theta_{ij}] \tag{10}$$

$$\Delta\theta_{ij} = \arctan \frac{r_{oij} + \varepsilon}{\text{distance}(P_i, O_{ij})} \tag{11}$$

where ε is the safe coefficient, r_{oij} is the radius of jth obstacle circle in ith dimension.

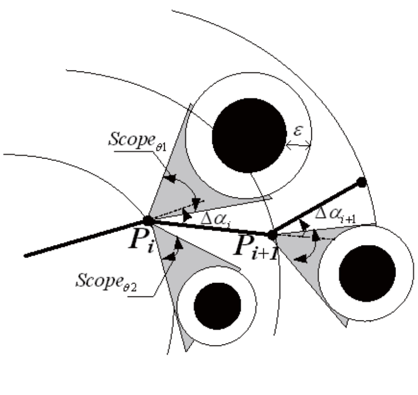

FIGURE 4. Collision avoidance modeling

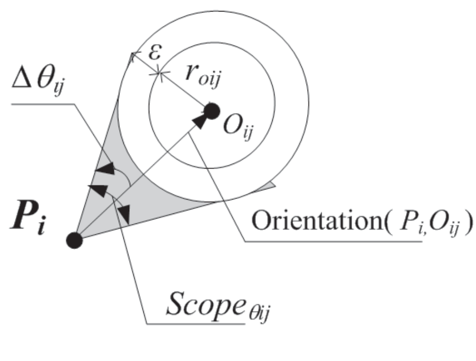

FIGURE 5. Forbidden area angle scope $S_{\theta ij}$

2) ***Cost Function.*** In our algorithm, the cost function of a candidate path is defined according to the following equations.

$$f(X) = \sum_{i=0}^{Dim-1} (\omega_1 \Delta l_i + \omega_2 \Delta \alpha_i + \omega_3 \Delta \mathrm{H}_i) \tag{12}$$

where Δl_i is the length of the ith segment on candidate path, and $\Delta \alpha_i$ is the turning angle of the ith waypoint and ΔH_i is its water depth adjusting value, ω_1, ω_2 and ω_3 are the weighting coefficients.

5. **Tests and Results.** Here some tests are carried out to illustrate the proposed algorithm in following environments. The algorithm is implemented on a Pentium 4 PC, and the same set of parameter values are set as: $pDim = 20$, $w_{\max} = 0.95$, $w_{\min} = 0.2$, $run_{Max} = 100$, $c_1 = c_2 = 1$.

Simulation 1: We adopt an environment of 60×38 grids size which is mathematically modeled based on an electronic chart data (range of Lng. 0.0895°×Lat. 0.0462°). Through adjusting values of the weighting coefficients ω_1, ω_2 and ω_3 in Equation (12), we get the different paths as shown in Figure 6.

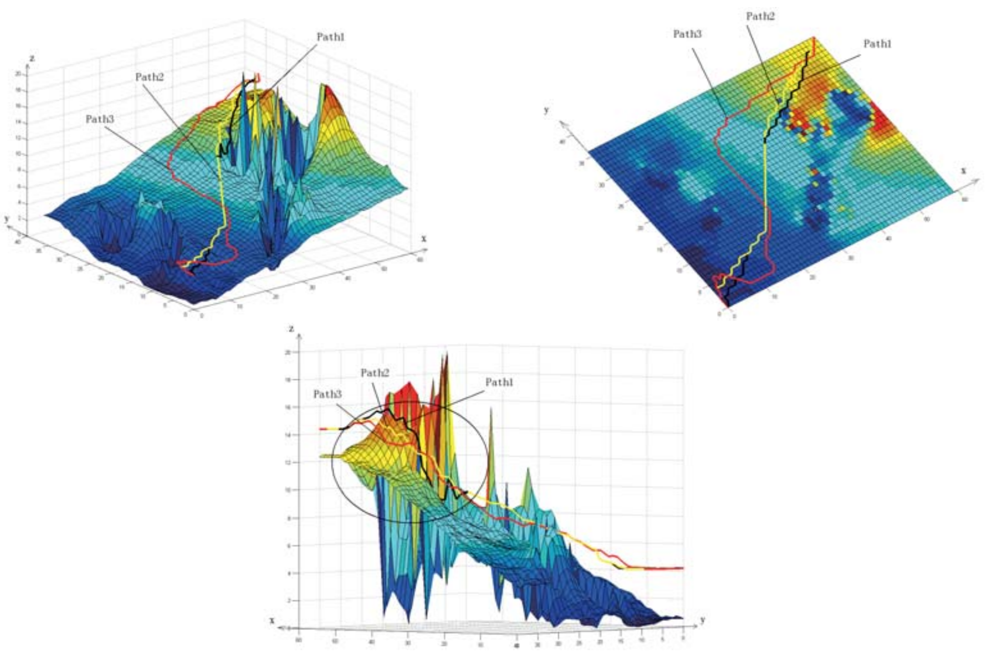

FIGURE 6. Path planning results in environment 1 in different visual angles

TABLE 1. Parameter setting and path results

$Path_i$	$Path_1$	$Path_2$	$Path_3$
$w_i(i = 1, 2, 3)$	$w_1 = 0.5, w_2 = 0.5,$ $w_3 = 0$	$w_1 = 0.5, w_2 = 0.5,$ $w_3 = 8$	$w_1 = 0.5, w_2 = 0.5,$ $w_3 = 12$
Path length/ km	*28.374*	*15.116*	*21.635*

From the results we get the conclusion that through adjusting the value of the weighting coefficients ω_1, ω_2, ω_3, we can adjust the different paths with different smooth degree. The candidate Path 2 is the best result.

Simulation 2: We adopt an accurate mathematical model of 170×170 grids size (Lng. 0.0025°× Lat. 0.0025°). We select values of the weighting coefficients $w_1 = 0.5$, $w_2 = 0.5$, $w_3 = 8$ in cost function and get the result in Figure 7. From the result we can get the conclusion that the EDPSO algorithm can solve the global path planning problem and get feasible paths.

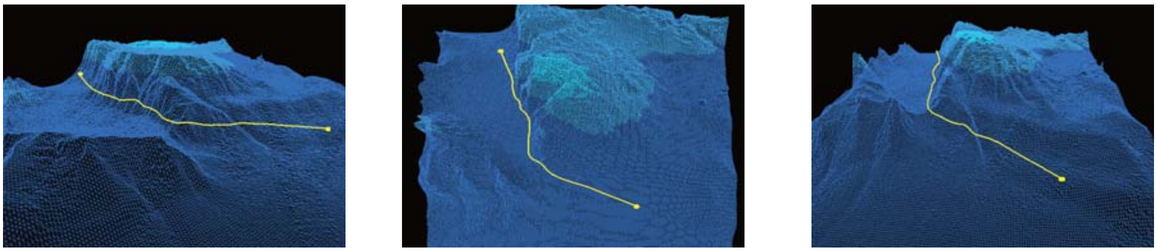

Figure 7. Path planning result in environment 2 in different visual angles

Table 2. Path results in different environments

	Experiment 1		*Experiment 2*	
Test contents	*Time/s*	*Length/km*	*Time/s*	*Length/km*
Results	*2.542*	*15.116*	*11.679*	*57.267*

6. **Conclusions.** In this paper, we give an equilibrious distribution parameter and propose a modified PSO evolutionary strategy in order to effectively increase PSO global optimization ability. Then, we design a path planning method for an AUV in a sea field environment. Simulation results in different sea field environments show EDPSO algorithm is feasible for AUV global path planning problem.

Acknowledgment. This research is partly supported by National Natural Science Foundation of China (No. 60904006 and 60975060), the National 863 Program (No. 2007AA04-Z239) and CAS Innovation Projects (No. ZKYGC09A02)

REFERENCES

[1] J. C. Latombe, *Robot Motion Planning*, Kluwer, Norwell, 1991.
[2] P. C. Zhou, B. R. Hong and J. H. Yang, Chaos genetic algorithm based path planning method for mobile robot, *Journal of Harbin Institute of Technology*, vol.10, no.7, pp.880-883, 2004.
[3] E. H.-C. Lu, C.-W. Huang and V. S. Tseng, Continuous fastest path planning in road networks by mining real-time traffic event information, *ICIC Express Letters*, vol.3, no.4(A), pp.969-974, 2009.
[4] J. Kennedy and R. C. Eberhart, Particle swarm optimization, *Proc. of the IEEE International Conference on Neural Networks*, pp.1942-1948, 1995.
[5] M. Clerc and J. kennedy, The particle swarm-explosion, stability and convergence in a multidimensional complex space, *IEEE Trans. on Evolutionary Computation*, vol.6, no.1, pp.58-73, 2002.
[6] Y. Zhao, W. Zu and H. Zeng, A modified particle swarm optimization via particle visual modeling analysis, *Computers and Mathematics with Applications*, vol.57, no.6, pp.2022-2029, 2009.
[7] Y. Q. Qin, D. B. Sun, N. Li and Y. G. Cen, Path planning for mobile robot using the particle swarm optimization with mutation operator, *Proc. of the 3rd International Conference on Machine Learning and Cybernetics*, pp.2473-2478, 2004.
[8] B. Sun and W. D. Chen, Particle swarm optimization based global path planning for mobile robots, *Control and Design*, vol.20, no.9, pp.1052-1059, 2005.
[9] J. Riget and J. S. Vesterstroem, A diversity-guided particle swarm optimizer-the ARPSO, *EVALife Tech. Rep. No. 2002-02*, University of Aarhus, Aarhus, 2002.
[10] X. Dai, X. Ning and Y. Shi, A novel path planning algorithm for mobile robots based on cloud model, *ICIC Express Letters*, vol.3, no.4(A), pp.877-881, 2009.
[11] Y. Yuan and X. Chen, Mobile robot path planning using swarm intelligence, *Computer Engineering and Applications*, vol.5, no.43, pp.52-55, 2007.
[12] H. Wang and X. Wei, Research on global path planning based on ant colony optimization for AUV, *Journal of Marine Science and Application*, vol.18, no.3, pp.58-64, 2009.
[13] X. Deng, J. Yi and D. Zhao, An optimal two-stage path planner, *International Journal of Information Technology*, vol.11, no.11, pp.21-30, 2005.
[14] Y. Kim and D.-W. Gu, Real-time path planning with limited information for autonomous unmanned air vehicles, *Automatica*, vol.44, no.3, pp.696-712, 2008.