International Journal of Control and Automation
Vol. 9, No. 5 (2016), pp. 1-10
http://dx.doi.org/10.14257/ijca.2016.9.5.01

An Improved Virtual Force-Directed Particle Swarm Optimization Positioning Algorithm

Jiachen Ma, Qiang Liu and Wei Xie

School of Astronautics, Harbin Institute of Technology, Harbin 150001, China
hitmajiachen@163.com

Abstract

This paper proposed an improved particle swarm optimization positioning algorithm based on virtual force-directed method for node localization of wireless sensor networks. The improved algorithm adopted adaptive inertia weight and adaptive mutation operation on global optimum, which overcomes the disadvantage of traditional particle swarm optimization algorithm that is easy to be trapped in local optimum. Fast convergence to near optimal solutions can be achieved after inertia weight is adjusted to be bigger, and smaller inertia weight can result in high precision solution. Through adaptive mutation on the global optimum, the improved algorithm can jump out of the current search area to maximize the coverage of the network nodes and the convergence speed. Compared with the virtual force-directed particle swarm optimization algorithm, the simulation results indicate that the improved algorithm has the advantages of faster convergence speed, lower energy consumption, higher precision and better stability.

Keywords: *particle swarm optimization; virtual force-directed; wireless sensor networks; adaptive mutation*

1. Introduction

Node deployment is an important issue for wireless sensor networks, which reflects the perception of the sensor network for physical world. Recently, the mobile deployment strategy of wireless sensor networks has become a hot research topic. The virtual force algorithm has been widely applied in the field of wireless sensor networks for dynamic deployment. And it has a significant effect on the self-organizing dynamic deployment optimization of mobile sensor nodes, but it has several disadvantages, such as large energy consumption, repeat coverage and network coverage uniformity.

Although the particle swarm optimization algorithm can achieve the layout optimization of wireless sensor network node, it is easy to fall into the "premature" so as to limit the search range of the particles [1-4]. A wireless sensor network coverage optimization strategy based on multi-particle swarm optimization particle evolution is proposed [3], which improves the stability and avoids the premature of elementary particle swarm optimization effectively. But the energy consumption of the nodes has not been reduced.

A hybrid algorithm that combining the particle swarm optimization algorithm and virtual force-directed algorithm for adaptive layout strategy of wireless sensor networks is proposed in this paper, which has better effect on node deployment. It can quickly get out of local optimum and effectively avoid the premature of the original algorithm to achieve maximum of coverage and convergence speed of the network nodes. In addition, it also realizes improvement of positioning accuracy, reduction of computation time and energy consumption.

ISSN: 2005-4297 IJCA

2. Virtual Force-Directed Particle Swarm Optimization Algorithm

2.1. Virtual Force Algorithm

The virtual force-directed particle swarm optimization algorithm is assumed that the wireless sensor network is a virtual physical system that includes force, acceleration and the mass. Sensor nodes, obstacles and hot zone can be applied to attract or repel by the sensor nodes. The moving direction and speed of sensor nodes vary depending on the quality of the node itself and the force of attraction and repulsion of neighbor nodes [5]. These sensor nodes always moves until it reaches the force balance or the upper limit of movable distance [6-9]. Standard virtual force algorithm assumes that the distance between node s_i and s_j is d_{ij}. Node s_i is subjected to gravitational force from node s_j if d_{ij} is bigger than the distance threshold d_{th} and less than the sensor node communication distance d. On the contrary Node s_i will be subjected to repulsion. Virtual force is calculated as follows [10]:

$$F_{ij} = \begin{cases} \left(w_r \left(\frac{1}{d_{ij}} - \frac{1}{d_{th}} \right), a_{ij} + \pi \right) & d_{ij} < d_{th} \\ 0 & d_{ij} = d_{th} \\ (w_a (d_{ij} - d_{th}), a_{ij}) & d_{th} < d_{ij} < d \\ 0 & d_{ij} \geq d \end{cases} \tag{1}$$

where a_{ij} is the orientation of a line segment from node s_i to node s_j, w_a and w_r respectively denote virtual force gravitational coefficient and repulsion coefficient for adjusting the density level of the sensor node after the layout of the virtual force algorithm optimization. In addition to the force among nodes, virtual force algorithm also includes the force of the obstacles and the hot zone needed to be measured. So all virtual forces on node S_j can be expressed as:

$$F_i = \sum_{j=1, j\neq 1}^{k} F_{ij} + F_{ia} + F_{ir} \tag{2}$$

The original position of node (x_{0ld}, y_{old}) can be updated according to the new location (x_{new}, y_{new}):

$$\begin{cases} x_{new} = x_{old} + \frac{F_x}{F_{xy}} \times \text{MaxStep} \times e^{-\frac{1}{F_{xy}}} \\ y_{new} = y_{old} + \frac{F_x}{F_{xy}} \times \text{MaxStep} \times e^{-\frac{1}{F_{xy}}} \end{cases} \tag{3}$$

where MaxStep is the allowable maximum of moving distance of the sensor nodes, F_{xy} is virtual force acting on each sensor node, F_x and F_y is the x-axis and y-axis component of the virtual force.

The density distribution of the final nodes depends on the distance between different nodes, the number of nodes in the hot zone and the number of nodes in obstacle, as shown in the formula (1) and (3). If the distance threshold value is too small, the distribution of the nodes will be too dense. Conversely, the nodes will sparsely and easily formed to cover the blind. Distance threshold is often determined empirically, it is not be adaptive to meet the requirements of different environments. As the nodes have equal status, node mobility is only based on the information of the neighbor nodes. when the neighbor node is changed, it will cause the node to move and lead to the change of the location information of the neighbor nodes. So this movement is continuous, which result in the entire network that cannot achieve the global optimization.

2.2. Particle Swarm Optimization Algorithm

For standard particle swarm optimization (PSO), if one of the particles found the best position, the other particles will rapidly move to close it. If the optimal position is a local optimum rather than the global optimum, the particles may fall into the local minimum and cannot get rid of the optimum, which is called "premature" phenomenon [11-12]. During the evolution of multi-particle swarm optimization, the total number m of particles is divided into j sub-populations and the individual particles in the sub-populations that experienced the best position in the d-dimensional space is denoted as $\mathbf{P}_i = (p_{i1}, p_{i2}, \cdots, p_{id})$ (also called p_{best}), the best position that each sub-species population of all particles experienced is expressed as g_{best}. Each sub-population adjust the direction of flight according to their own flight experience and their own global optimal of sub-population.

The speed and position of each generation of particles evolved according to the following formulas:

$$\begin{aligned} V_i^{k+1} &= \omega V_i^k + c_1 r_1 (pbest_i^k - X_i^k) \\ &+ c_2 r_2 (gbest_i^k - X_i^k) \end{aligned} \tag{4}$$

$$X_i^{k+1} = X_i^k + V_i^{k+1} \tag{5}$$

where i is the i-th particle, k is the k-dimensional of particles, c_1 and c_2 are the accelerating factors, r_1 and r_2 are independent random variable in the rang [0,1], ω is called inertia weight. It can be expressed as follows:

$$\omega = \omega_{max} - \frac{\omega_{max} - \omega_{min}}{iter_{max}} \times t \tag{6}$$

ω_{max}, ω_{min} are the maximum and minimum of ω, $iter_{max}$ is the maximum iterating, t is the current iteration.

2.3. The combination of virtual force algorithm and PSO

The combination of virtual force algorithm and PSO can effectively direct the particles fly to the optimal positions, which improves the search ability and converge speed. The velocity of each particle is updated according to two factors: the hiotorical position and virtual forces of sensor nodes [5]. The formula is defined as follows:

$$\begin{aligned} V_i^{k+1} &= \omega V_i^k + c_1 r_1 (pbest_i^k - X_i^k) \\ &+ c_2 r_2 (gbest_i^k - X_i^k) + c_3 r_3 m_i^k \end{aligned} \tag{7}$$

where ω, r_1, r_2, c_1, c_2, $pbest_i^k$, $gbest_i^k$, X_i^k are as same as formula (4) and (5). c_3 is a acceleration factor, r_3 is also independent random variable in the rang [0,1], m_i^k is the proleptic motion suggested by virtual force of i-th particle in k-th dimension, which is computed by the following formula:

$$m_i^k = \begin{cases} \dfrac{F_x^{(i,(k+1)/2)}}{F_{xy}^{(i,(k+1)/2)}} \times MaxStep \times e^{\frac{-1}{F_{xy}^{(i,(k+1)/2)}}}, j = 1,3,5 \mathrm{L} \ 2n-1 \\ \dfrac{F_y^{(i,(k+1)/2)}}{F_{xy}^{(i,(k+1)/2)}} \times MaxStep \times e^{\frac{-1}{F_{xy}^{(i,(k+1)/2)}}}, j = 2,4,6 \mathrm{L} \ 2n \end{cases} \tag{8}$$

where the subscript of each parameter indicates the coordinate of the virtual force, the superscript presents the index of particles and the index of wireless sensor nodes which

the virtual force exerts on. The correlative virtual forces are carried out by formula (1).
In the PSO algorithm, it exists problems that premature convergence and easy to fall into local minima [7]. To overcome these disadvantages, this paper adopts the idea that changable inertia weight and proposed an improved particle swarm optimization algorithm based on adaptive, which improves the original PSO algorithm from two aspects of the inertia weight value and the global optimal position.

3. Improved Algorithm Based on Adaptive

In this paper , the improved adaptive algorithm mainly include two aspects： the inertia weight adaptive value and global optimal position which is improved from the fitness value of adaptive mutation operation.

3.1. Adaptive Inertia Weight

The inertia weight has an important role in the parameters of PSO algorithm, and it reflects the reaction of the previous flight speed of the particles on the current value. Global search ability becomes stronger and convergence gets faster when its value is large, but the drawback is the lack of precision of solution. whereas, if the value is small, both the local search ability and the accuracy of the solution can be promoted, but the convergence rate becomes slower and is more likely to fall into local minimum. The suitable ω value can balance the global search ability and local search ability, which can lead to get the optimal solution.

To obtain the value of the adaptive inertia weight in this paper, there are has mainly two processes: Iterative pre- positioning algorithm to achieve fast convergence to the optimal solution; The global search ability will be improved as better fitness value, which can speed up the rate of the gathering to the global optimum position and improve the local search ability, and resulting in a highly accurate solution. Specific improvement measures are as follows:

$$\omega(i) = \left[\omega_1 + (\omega_2 - \omega_1) \times \frac{f(t) - f_{min}}{f_{max} - f_{min}}\right] \times \left(\frac{T_{max} - T}{T_{max}}\right)^3, i = 1, 2, \cdots, N \tag{9}$$

In general, $\omega_1 = 0.3$ 、 $\omega_2 = 0.8$, $\omega_2 > \omega_1$. T is the current iteration number of times, N is the total number of particles in particle swarm. In order to avoid $\omega(i)$ is too small in the iteration value, we set a lower limitation which is 0.2, in another word, $\omega(i) = 0.2$, if $\omega(i)$ is lower than limitation. $f(i)$ is the fitness value of the i-th particle, $f_{\max}$ and $f_{\min}$ are the maximum and minimum values of all populations particles fitness. The particle velocity equation is as follows:

$$\begin{aligned} V_i^{k+1} &= \omega(i)V_i^k + c_1 rand()\,(pbest_i^k - X_i^k) \\ &\quad + c_2 rand()\,(gbest_i^k - X_i^k) \end{aligned} \tag{10}$$

3.2. Adaptive Mutation Operation on Global Optimum Position

Fitness value of the particle can reflect the level of the current position of particles. So current fitness value of all particles can be considered as a sample, whose variance can be used to quantitatively evaluzate the degree of aggregation of the entire population. If the dense of population is getting bigger, it indicates the search capabilities of the entire population gets worse. At the moment, the mutation operation on global optimum position can ensure the entire population to jump out of the current search area.
The formula of particles population fitness value variance δ^2 is expressed as follows:

$$\delta^2 = \sum_{i=1}^{n} \left(\frac{f(t) - f_{avg}}{F}\right)^2 \tag{11}$$

where n is the number of particles in entire population, $f(i)$ is fitness value of the i-th particle, f_{avg} is average fitness value of all the particles in the population, F is a normalization factor, and $F = \max(1, |f(i) - f_{mvg}|)$.

The formula of mutation rate of the global optimum position is:

$$p_c = p_{max} - (p_{max} - p_{min})\frac{\delta^2}{n} \tag{12}$$

where p_{max} and p_{min} respectively indicates that the maximum value and the minimum value of g_{best} and $p_{max} = 0.3$, $p_{min} = 0.4$. g_{best_k} is the k-dimensional component of g_{best} .

Assuming that the unknown node $K(x, y)$ has three or more neighbors communicating anchor node $K_i(x_i, y_i)$, $i = 1, 2, \cdots n.$ The distance between unknown node and anchor node by RSSI ranging method is $r_i = \sqrt{(x' - x_i)^2 + (y' - y_i)^2}$, where (x', y') is the estimated value of unknown node.

The fitness value function is defined as follows:

$$\text{fitness}(k) = \sum_{i=1}^{n} |d_i - r_i| \tag{13}$$

Node localization process is as follows:

(1) A certain number of anchor nodes and unknown nodes are deployed randomly in the search space (target area), then the anchor node starts sending its own information (including the node ID, position) to the adjacent nodes.

(2) Unknown node calculates the distance between itself and the anchor node and gets neighbor connectivity information of the anchor node and RSSI model.

(3) By the improved algorithm, neighbor connectivity anchor nodes of unknown node calculate and obtain the results of itself positioning.

4. Simulation

In this experiment, it is assumed that the wireless sensor nodes are deployed in 100m ×100m dimensional planar area, 100 of sensor nodes randomly distributed within this area, where the number of anchor nodes is 50. Parameter settings in the localization algorithm based on particle swarm optimization of adaptive strategies: $\omega_1 = 0.3$, $\omega_2 = 0.8$, $p_{max} = 0.3$, $p_{min} = 0.4$, $\omega(i)_{min} = 0.2$, $c_1 = c_2 = 2$. Total population of particles is $N = 30$, the total number of iterations, $T = 200$, the maximum dimensional position of the particles is 100m, and the maximum speed is 10 m/s. In this paper, the set of nodes C regional coverage, the proportion of coverage area and monitoring area ratio of the total area of the node set C are adopted from document [9] and denoted by R, which is as followed:

$$R = \frac{\sum_{j=1}^{m \times n} c_{x,y}(p_j)}{m \times n} \tag{14}$$

In order to reduce interference of random errors for the test, 200 positioning experiments were implemented to obtain the final experimental data. Node parameters of the sensor are shown in Table 1.

Table 1. Main Parameters of Sensor Nodes

Type	r/m	d/m	MaxStep/m	λ	β	d_{th}/m
Anchor nodes	7	14	3	0.5	0.6	11
Mobile nodes	14	3	0.5	0.6	11	14

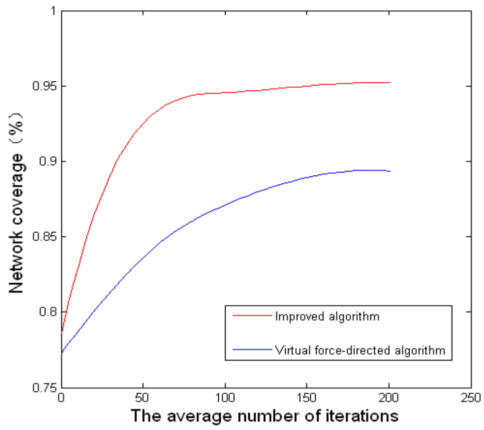

Figure 1. Coverage Comparison Between Original and Improved Algorithm

Figure 1 shows the coverage comparison between original and improved algorithm. The simulation results show that the coverage of virtual force-directed particle swarm optimization was 87.4%, while the coverage of the improved algorithm can reach 95.6%. And the improved algorithm has faster convergence that global optimal solution of stable network coverage is obtained after 80 iterations, while the original algorithm takes about 175 times before convergence.

In order to verify the performance of the algorithm, the two algorithms were carried out by 30 independent experiments in the same simulation environment. The comparison results ares shown in Table 2.

Table 2. Performance Comparison Between Two Algorithms.

Performance	The average coverage	The number of iterations	Computation time
Virtual force-directed algorithm	86.4%	168	148.4s
Improved algorithm	94.2%	122	112.6s

It can be concluded that the improved algorithm can reach 94.2% at average coverage that is significantly higher than 86.4% of original algorithm and has a faster convergence rate. Computation time is also lower, has reduced by 35.8s than the original algorithm 148.4s. From three aspects: average coverage, number of iterations and computation time, the improved algorithm has greater satisfaction than the original algorithm.

During the node localization process of wireless sensor network, ranging error directly determines the positioning progress and stability. Therefore in this experiment, range error is considered as a prerequisite to test the accuracy of the positioning algorithm. Under the same distance measurement error conditions, the virtual force directed particle swarm optimization algorithm and the improved algorithm respectively completed 100 times positioning calculation, repeating the positioning operation under the different ranging error condition. The experimental evaluations are as follows：

Average location error：

$$\mathrm{AVE} = 0.01 * \sum_{i=1}^{100} \sqrt{(x_i - x)^2 + (y_i - y)^2} \tag{15}$$

Positioning variance:

$$\mathrm{MSE} = 0.01 * \sum_{i=1}^{100} (\sqrt{(x_i - x)^2 + (y_i - y)^2} - \mathrm{AVE})^2 \tag{16}$$

Both algorithms positioning results as shown in Table 3 and Table 4.

Table 3. The Positioning of the Virtual Force Directed Algorithm

Ranging error	AVE	MSE
5%	2.26	1.26
10%	4.46	4.47
15%	6.67	8.37
20%	7.44	12.74
25%	11.42	19.42

Table 4. The Positioning of the Improved Algorithm

Ranging error	Adaptive threshold	The average number of iterations	AVE	MSE
5%	6	7.49	2.28	0.78
10%	12	4.94	2.94	2.64
15%	20	4.06	4.22	4.22
20%	26	4.96	4.98	7.98
25%	32	6.86	6.24	9.94

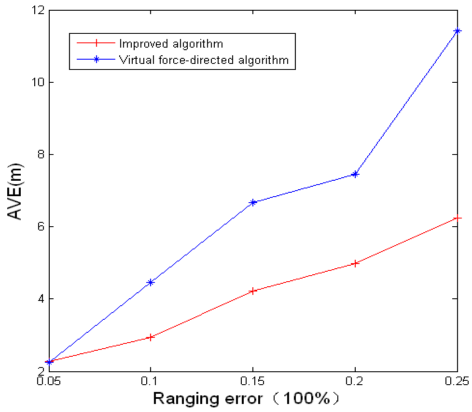

Figure 2. Relation Between Ranging Error and Average Location Error

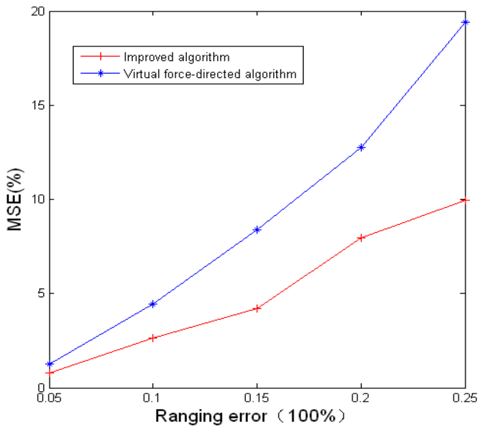

Figure 3. Relation Between Ranging Error and Positioning Variance

Figure 2 shows relationship between ranging error and average location error and Figure 3 shows relationship between ranging error and positioning variance. Figure 2 and Figure 3 show that both the average positioning error and positioning variance of the proposed algorithm are less than the original algorithm, which indicates that the proposed algorithm has better stability than the virtual force-directed particle swarm optimization algorithm.

From Figure 2 to Figure 3, it can be inferred that the performance of the two positioning algorithm has small gap when the system ranging error is small. But the excellent positioning performance of the proposed algorithm begin to emerge when the distance error becomes large, which shows that the improved algorithm can reduce the impact of ranging error on positioning accuracy to a large extent.

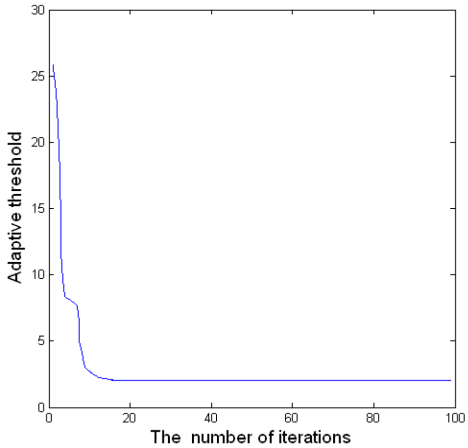

Figure 4. Relation Between Fitness Value and the Number of Iterations

Figure 4 is a diagram of the fitness value relationship with the number of iterations when the ranging error is 5%. Figure 4 shows the positioning process of the improved algorithm when the ranging error is 5%. It can be concluded that the improved algorithm can converge to higher accuracy of the optimal solution whose the number of iterations is less than 10 times. So it has faster convergence ability, lower energy consumption, which is suitable for higher requirements of wireless location system.

5. Conclusion

This paper studies on the network optimization strategy of node deployment process of wireless sensor network, combining virtual force and particle swarm optimization algorithm. A particle swarm optimization algorithm based on adaptive adjustment for virtual force method is proposed, which can jump out of the current search area through adaptive mutation on the global optimum position. The improved algorithm can effectively avoid the "premature" of the original algorithm, which can obtain optimal coverage and convergence speed. The simulation results show that the improved algorithm can effectively improve the coverage of the network and the accuracy of node positioning, increase the convergence speed of the network, reduce the number of iterations and decrease time-consuming.

Acknowledgements

This work was supported by Natural Scientific Research Innovation Foundation in Harbin Institute of Technology (HIT.NSRIF 201606).

References

[1] T. Panigrahi, G. Panda and B. Mulgrew, "Distributed DOA estimation using clustering of sensor nodes and diffusion PSO algorithm , warm and Evolutionary Computation", , vol. 9, **(2013)**, pp. 47-57.

[2] A. Munir, A. Gordon-Ross and S. Lysecky, "A lightweight dynamic optimization methodology and application metrics estimation model for wireless sensor networks, Sustainable Computing: Informatics and Systems", vol. 3, **(2013)**, pp. 94-108.

[3] M. A. Cavuslu, C. Karakuzu and F. Karakuzu, "Neural identification of dynamic systems on FPGA with improved PSO learning, Applied Soft Computing", vol. 12, **(2012)**, pp. 2707-2718.

[4] M. A. Mohandes, "Modeling global solar radiation using Particle Swarm Optimization (PSO), Neurocomputing", vol. 86, **(2012)**, pp. 3173-3145.

[5] X. Wang, S. Wang and D. Bi, "Virtual force-directed particle swarm optimization for dynamic deployment in wireless sensor networks. Advanced Intelligent Computing Theories and Applications. With Aspects of Theoretical and Methodological Issues", Springer Berlin Heidelberg **(2007)**.

[6] H. T. Dorrah, A. M. El-Garhy and M. E. El-Shimy, "PSO based optimized fuzzy controllers for decoupled highly interacted distillation process", Ain Shams Engineering Journal, vol. 3, **(2012)**, pp. 251-266.

[7] A. Alireza, "PSO with Adaptive Mutation and Inertia Weight and Its Application in Parameter Estimation of Dynamic Systems, Acta Automatica Sinica", vol. 37, **(2011)**, pp. 541-549.

[8] W. H. Liao, Y. Kao and Y. S. Li, "A sensor deployment approach using glowworm swarm optimization algorithm in wireless sensor networks", Expert Systems with Applications, vol. 38, **(2011)**, pp. 12180-12188.

[9] H. Yu and W. Xiaohui, "PSO-based Energy-balanced Double Cluster-heads Clustering Routing for wireless sensor networks", Procedia Engineering, vol. 4, **(2011)**, pp. 171-175.

[10] C. Walha, H. Bezine and A. M. Alimi, "An Adaptive Particle Swarm Optimization Method for Solving the Grasp Planning Problem", Procedia Engineering, vol. 41, **(2012)**, pp. 426-435.

[11] B. Singh and D.K. Lobiyal, "Energy-aware Cluster Head Selection Using Particle Swarm Optimization and Analysis of Packet Retransmissions in WSN", Procedia Technology, vol. 4, **(2012)**, pp. 171-176.

[12] H. Wang, H. Jiang.K. Xu and G. Li. Reactive Power Optimization of Power System based on Improved Particle Swarm Optimization. Proceedings of 2011 4th International Conference on Electric Utility Deregulation and Restructuring and Power Technologies (DRPT 2011), **(2011)** July 6-9; Weihai, China.

Authors

Jiachen Ma, He received the master and Ph.D. in control theory and control engineering from Harbin Institute of Technology, China. And research interests on intelligent robot, embedded system and application, precise machine vision inspection and automotive and ship electronics. He is a professor of school of astronautics of Harbin Institute of Technology, China.

Qiang Liu, He received the master degree in Control Science and Engineering from Shan Dong University of Science and Technology. And research interests on intelligent control, multi-robot System. He is now a Ph.D. Candidate in control theory and control engineering of school of astronautics in Harbin Institute of Technology.

Wei Xie, He received the master degree in Traffic information and control engineering from Southwest Jiaotong University. And research interests on robot control, intelligent information processing, intelligent service robot and integration for intelligent production line. He is now a senior system analyst and lecturer of Harbin Institute of Technology (Wei Hai).

