International Journal of Hybrid Information Technology
Vol.8, No.6 (2015), pp.237-246
http://dx.doi.org/10.14257/ijhit.2015.8.6.23

A Improved Algorithm of Quantum Particle Swarm with Fast Convergence

Zhang Chun-na[1], Li Yi-ran[2] and Li Jun-feng[3]

(1.School of Software, University of Science and Technology Liaoning, Anshan Liaoning 114051, China;
2. College of Applied Technology, University of Science and Technology Liaoning, Anshan Liaoning 114011, China
3. The cold rolling plant of Ansteel co.,Ltd. Anshan Liaoning, 114021, China)
E-mail：zcn1979@163.com

Abstract

An improved algorithm of quantum particle swarm with chaos is presented to solve the problem that the traditional particle swarm algorithm is easy to fall into local optimum and converges very slowly. Through analysis the current state of particles in the iteration, to determine and deal with the particle of poor performance, and keep the normal state continue to complete the search optimal solution, which effectively inhibit premature phenomenon of the particles, and improve the overall search ability of particle swarm. At the same time, in order to improve the performance of the algorithm, introducing chaos mechanism, further enhance the search ability of particle. The experiments of benchmark function show that the improved algorithm has obvious advantages compared with the other two algorithms, it has higher stability and accuracy and faster convergence speed at the same time.

Keywords: *quantum particle swarm; chaotic mechanism; particle performance; convergence*

1. Introduction

Particle swarm optimization (PSO) is a kind of swarm intelligence search algorithm. Which is used to simulate the foraging behavior of groups inhabit biological, because the algorithm has the advantage of simple structure, less parameters, and without the gradient information. It is respected by many scholars after algorithm is proposed; the current has been applied to multi-objective optimization, pattern recognition, neural network, and many other areas [1-3]. However, the excessive dependent on the optimal particle's search ability, the particle of poor performance is easy to cause slowly convergence and fall into local optimum in the late. Based on this, the PSO algorithm is faced with two problems to be solved: 1) the path of the particles is reasonable in the process of search; 2) Whether can enhance the search ability of poor performance particles, to escape from local constraints.

In order to better reflect the ability of the behavior of particles, this paper proposes a quantum particle swarm improved algorithm with chaos. The algorithm is based on the principle of quantum mechanics, the particle is quantized, don't need to consider the speed direction vector in the evolution equation, which makes the particle search strategy is more simplified, easy to control [4]; At the same time, in order to reflect the ergodicity of particles, introducing chaos mechanism, enhance the ability of particle activity; Aiming at the inherent premature phenomenon of PSO, plan to do mutation operation for the particle of poor performance, make it have the ability to follow the optimal particle’s moving, get

ISSN: 1738-9968 IJHIT

rid of the bondage of local optimum [5, 6]. The experimental show that the ability of particle move has been significantly improved, diversity also enhanced.

2. The Related Problem Analysis

2.1. Basic Particle Swarm Algorithm

PSO algorithm is a kind of evolutionary behavior simulation by Communication cooperation between individuals to get the optimal solution. In Q dimensional space, initialize n particles $\{x_1, x_2, \cdots, x_n\}$, Among them, the position of the i particles in space is $x_i = \{x_{i1}, x_{i2}, \cdots, x_{iQ}\}$, the particle velocity is $v_i = \{v_{i1}, v_{i2}, \cdots, v_{iQ}\}$, $i = 1, 2, \cdots, n$, the direction of travel is random[7]. Each particle's trajectory follows two positions: the current optimal extreme value Pb and the overall optimum extreme value Gb. In the process of iteration calculate the particle's fitness value, update the particle's velocity and position, after every iteration to update Pb and Gb. After repeated iteration until the termination condition is satisfied, to get the optimal solution. The particle state related formula is as follows:

$$v_{iq}^{'} = \alpha v_{iq}^{"} + \beta^0 (x_{Pb} - x_{iq}) + \beta^1 (x_{Gb} - x_{iq}) \tag{1}$$

$$x_{iq}^{'} = x_{iq}^{"} + v_{iq}^{'} \tag{2}$$

In the formula, α is inertia weight, to balance the velocity relationship between the current and previous particle in the moving process. β^0、$\beta^1 \in [0,1]$ distributed at random.

For the problem of the PSO's prematurity is mainly embodied in the algorithm's convergence speed [8], considering the nature of algorithm, the particle's trajectories followed the optimal particle in the evolutionary process. If there is particle of poor performance, the rate has fallen dramatically or oscillation in one dimension, it is very easy to fall into local optimum and appear premature phenomenon. In order to solve this problem, this paper starts from two aspects: first, introducing chaos mechanism improved the particle's traverse ability and increased the diversity of solution. Moreover, particle of poor performance get rid of oscillation by variation.

2.2. The Chaos Mechanism

The chaos phenomenon is widespread in nature, it belongs to the category of nonlinear and has the characteristics of periodicity and regularity, sensitive to initial conditions, can search to all internal state in accordance with the laws of its own in a given area, and does not repeat[9,10]. In this way, can take advantage of the nature of the chaos to search optimization, search steps are as follows:

(1) Define initial region, set the initial state vector of N dimension $R_0 = (R_{01}, R_{02}, \cdots, R_{0N})$, adjacent between each value in R_0, and the difference is very small.

(2) The initial vector R_0 was calculated by using logistics equation, generated chaotic sequence $c_1, c_2, \cdots, c_n$. Here, after several iterations, the system will be completely in a state of chaos. Vector layer can be expressed as:

$$c_{i+1} = c_i (1 - c_{i-1}) \lambda \tag{3}$$

Among them, λ is iterative control parameter?

(3) Set space particles X_i, using the type（3） to get a better position of X_i, as $X_i^{'}$.

$$X_i^{'} = r \cdot rnd \cdot c_j + X_i \tag{4}$$

Among them, r is activity radius of the particle X_i, $rnd \in [-1,1]$， $j \in [0, n]$.

The main idea of the particle swarm optimization algorithm based on chaos mechanism reflected as follows: on the one hand, using the chaos sequence to initialize the particle's position and velocity, as has the characteristics of periodicity, both keep the diversity of the particles, and enhance the search ability of the particles. Moreover, chaotic state can make the motion of the particle is persistent.

The particle is initialized by chaos: the X_i respectively assigned the initial value in the type (4), and revised particle swarm iteration speed:

$$\begin{aligned} v'_{i,j}(t+1) = {} & a v''(t) + b^0(t)(x_{lb}(t) - x_{i,j}(t)) \\ & + b^1(t)(x_{gb}(t) - x_{i,j}(t)) \end{aligned} \tag{5}$$

In the formula, a is constant in $(0,1]$, b is the normal distribution random number in $N[0,1]$, $i \in [1,n]$, $j \in [1,m]$, n is the particle number, m is a spatial dimension. For $v''(t)$

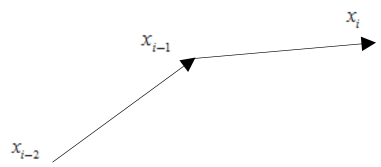
$$v''(t) = \begin{cases} v_{i,j}(t) & q = 0 \\ N[0,1] \cdot d \cdot \tilde{v} & q = 1 \end{cases} \tag{6}$$

$$q = \begin{cases} 0 & f(x_{gb}(t-1)) > f(x_{gb}(t)) \\ 1 & f(x_{gb}(t)) = f(x_{gb}(t-1)) = \cdots = f(x_{gb}(t-5)) \end{cases} \tag{7}$$

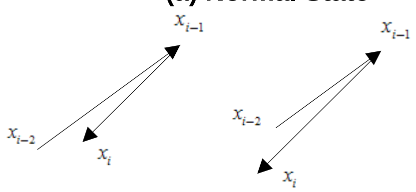
In the formula, $\tilde{v} = v_{\max} \cdot c_i / 1.1$, $d = f(x_{gb}) - f(x_T)$, c_i is a new chaotic sequence, $f(x_{gb})$ is the satisfactory solution, $f(x_T)$ is the target solution.

2.3. The Particle's Performance Determine and Improve

In evolution, the particle's state can be divided into two kinds: normal walking and oscillation does not forward, had mentioned in front, the obvious symbol of the particle's performance degradation is oscillation does not forward in one dimension, as shown below:



(a) Normal State



(b) Abnormal State

Figure 1. Particle Moving Trajectory

Figure 1 is abstraction of the particle motion trajectory, (a) is normal motion state, (b) shows the particle reciprocating oscillation within a certain range. Determine a particle whether oscillation does not forward by type (8):

$$\frac{\max(|x_i - x_{i-1}|,|x_{i-1} - x_{i-2}|)}{|x_i - x_{i-2}|} > h \tag{8}$$

If the position of the particle's before or after generation satisfies the equation (8) in the iteration, then shown that the particle exist oscillation phenomenon, leading search ability decline, and even premature convergence. x_i、x_{i-1}、x_{i-2} Represent the particle's position of before or after generation.

For determine the particle's performance, this paper set up a basic principle: particles in the process of the flight each generation follow the optimal particle, explain the particle's performance is good. On the contrary, if particle of successive generations appear reciprocating oscillation or stagnant phenomenon, it suggest that the particle's performance fell. Here, defines two parameters for analysis: oscillation factor and stagnation factor. The following were explained:

(1) The oscillation factor, assume that the particle of before and after two generations meet type (8) in the iteration, determine the particle takes place oscillation, initialization setting oscillation factor $\theta_1 = 0$, once oscillation θ_1 incremented 1. For each generation particle is suitable for the type (8) in the iterative, until the particle appears the mutation.

(2) The stagnation factor: the factor used to detect the working state of the optimal particle, set the initial stagnation factor $q_2 = 0$, detect the current optimal particle's position is or not coincident the previous position in the iteration. If no change, then q_2 incremented 1; otherwise, q_2 is set to 0.

To better illustrate the particle's performance, the following define the particle performance indicators, combine oscillation factor and stagnation factor, and to set the weight parameters for them, the correlation formula is as follows:

$$P = d_1 q_1 + d_2 q_2 \tag{9}$$

In the formula, P is the particle performance indicators, d_1、d_2 are respectively oscillation factor and stagnation factor weighting parameters, this paper set $d_1 = 0.5$, $d_2 = 2.5$.

3. Algorithm analysis

3.1. The Quantum Particle Swarm With Chaotic

Quantum particle swarm algorithm embodies two characteristics of quantum science: superposition and probability expression characteristics [11]. The two can make the particles exhibit more status, which increases the diversity of the population as a whole. At the same time, in order to increase the diversity, the quantum particle swarm optimization algorithm with chaotic is put forward. Introducing chaos mechanism, so that the global search ability of particle reinforced, and placing the particle in the quantum space that represented by the wave function, particles can freely search for possible solutions, its status is not limited by the position and velocity vector. The following is wave function to identify the state of the particle:

$$\int_{-\infty}^{+\infty} L_p dxdydz = 1 \tag{10}$$

By monte carlo algorithm transforms the quantum state obtained iterative formula:

$$P_{mb} = \frac{1}{l}\sum_{i=1}^{l} P_i \tag{11}$$

$$P_{mk} = \theta P_{ik} + (1-\theta) P_{gk} \tag{12}$$

$$L_p(t+1) = P_{ik} \pm a\,|\,P_{mk} - L_p(t)\,|\ln^{\frac{1}{\lambda}} \quad (13)$$

In the above formula, P_i is the particle's current location, P_{gb} is the global extremum, P_{mb} is the optimal median, l is the number of particle, P_{mk} is a random point between P_{ik} and P_{gk} , a is expansion contraction factor, it is a quantitative indicators, used to control the speed of convergence of the algorithm, In the iteration take a random number between $(0,1)$, if $a > 0.5$, is a positive value, otherwise take a negative value.

3.2. The Improved Algorithm

Setting a threshold r according to actual condition measure the performance of the particles, and a calculation is made for formula (9), when the particle's performance index is lower than r, using the formula （5） reassign new speed to the particle, so as to change the running state of the particle, to escape local bound. In quantum particle swarm optimization (PSO) algorithm, a particle may be defined as $x_i^0 = \{v_{i1}, \cdots, v_{ij}, \cdots, v_{im}\}$, among them, v_{ij} represents j-th cluster center vector of i-th particle, The fitness function is still using the PSO's objective function $J(u,Z)$. Based on this, this paper presents an improved algorithm ICQPSO based on PSO, the specific steps of the algorithm are as follows:

Step 1: arrange for the sample data;

Step 2: Determine the size of population and initialize particle. Determine the center vector of the clustering for each particle in the specified range of the data vector, randomly selected local and global optimal individual;

Step 3: Calculate the fitness initial value. According to the fitness function calculate fitness value of each particle, select the best location of the particles as the location of the initial iteration;

Step 4: Referencing the fitness initial value implemented mutation operation for particle, calculated the particle's performance metrics, determined the current performance status of each particle, If it does not satisfy the threshold, then using the formula (5) modified the particle's velocity, and comparing with the extreme. If good, then replace it；

Step 5: Produce a new generation individual x_i^t.

Step 6: Referencing the particle swarm algorithm set the terminating conditions of iteration. If satisfied may be terminated, otherwise return to step 3.

Pseudo code is as follows:

```
void public ICQPSO(int QP, int Cluster Scale, double LO, double GO, double o1, double o2)
{
    CPopulation c_QP=PopulationInit(OP, ClusterScale, LO, GO);
        while(i< GO)
        {
            c_QP.fitness= CalculateFitness(QP);
            P_gb = Goptimum (c_QP.fitness);
            if (TerminationExe( P_gb ,i))
                break;
            while(j< LO)
            {
                CParticle c_Particle=PartGroup(c_QP);
                Initchaoic(j, c_Particle);
                if (!CapaExe( r ,o1,o2))
                    Initchaoic(j, c_Particle, c_Particle.v);
```

```
                if (TerminationPart(c_Particle, P_mb, P_gb))
                        break;
    }
            }
     }
```

4. Experimental Analysis

The simulation experiments respectively studied the ability of optimization and convergence performance of the improved algorithm. In terms of optimization ability, this paper selected 7 benchmark functions to complete the test, including *Schwefel* 、*Rosenbrock* 、*Noncontinuous* 、*Griewank* 、*Penalized* 、*Penalized*2 、*Ackley* . Benchmark function parameter information is as follows:

Table 1. Benchmark Function

name	dimension	value range	optimal value
Schwefel	30	$[-10,10]^D$	-12569.5
Rosenbrock	30	$[-10,10]^D$	0
Noncontinuous	30	$[-5.12,5.12]^D$	0
Griewank	30	$[-600,600]^D$	0
Penalized	30	$[-50,50]^D$	0
*Penalized*2	30	$[-50,50]^D$	0
Ackley	30	$[-100,100]^D$	0

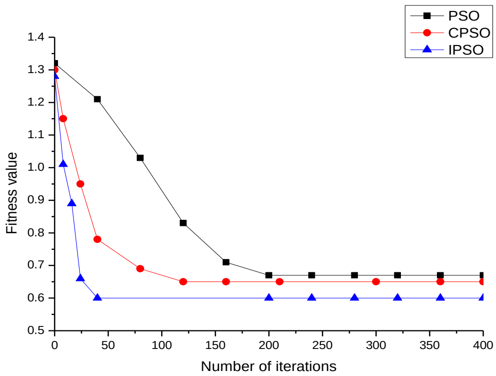
In the experiment, the benchmark function randomly is tested 100 times, then take the average. This paper selects three kinds of particle swarm optimization algorithm to compare, they are the standard particle swarm algorithm (PSO), the particle swarm optimization algorithm with chaotic (CPSO), and the improved algorithm is proposed in this paper(ICQPSO). The size of the population is 20, the maximum number of iterations is 1200, the particle number of three kinds of algorithm is consistent, all is 30, the oscillating factor and stagnation factor respectively set 0.5 and 2.5, threshold is 80. There are three kinds of evaluating indicators: Mean Best Fitness (MBF) , Standard Deviation (SD), Time efficiency(TE), among them, MBF is used to test the precision of algorithm, SD is used to test the stability of algorithm, TE is reflects the time efficiency of the algorithm. The following table is the appropriate comparison results.

Comparative analysis three kinds of algorithm from table 2, the data of benchmark function show that the improved ICQPSO algorithm has great advantage in the contrast of MDF and SD, the stability and accuracy compared with the traditional PSO algorithm has significantly improved, mainly because of the chaotic mechanism is introduced in the algorithm, enhance the global search ability, as for the advantage of CPSO algorithm embodied in the improved aspects of poor performance particle, judging the state of particles, in order to activate particles, which makes the algorithm more easy to obtain the optimal solution. But the algorithm also has some defects, the time efficiency of some benchmark functions slightly inferior compared with other algorithms, among them, *Schwefel* 、*Noncontinuous* 、*Griewank* three performance difference is obvious, The main reason is in the process of to determine the performance of the particles that need to be repeated calculation, if there is many particle does not meet the requirements need to set parameters. This paper focuses on the particle performance optimization, so the defects of time efficiency is tolerable.

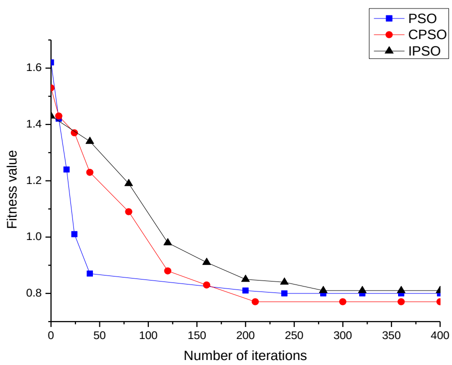
Table 2. The Comparison Results Of Benchmark Function Test

benchmark function	The type of indicators	PSO	CPSO	ICQPSO
Schwefel	MBF	-7297.182	-6529.330	-99087.202
	SD	9.27e+0.02	9.27e+0.02	9.27e+0.02
	TE	6.31	6.35	6.48
Rosenbrock	MBF	45.362	12.195	3.986
	SD	29.398	23.113	4.859
	TE	4.15	4.20	4.13
Noncontinuous	MBF	43.194	48.224	0
	SD	17.975	5.237	0
	TE	4.10	4.12	4.25
Griewank	MBF	0.984	0.019	0
	SD	1.573	0.025	0
	TE	7.12	7.24	7.28
Penalized	MBF	8.775	7.639	4.98e-32
	SD	4.292	3.641	2.35e-32
	TE	68.22	61.19	65.43
*Penalized*2	MBF	32.293	19.786	2.51e-26
	SD	31.104	19.729	1.56e-26
	TE	69.34	67.25	68.36
Ackley	MBF	10.193	4.537	2.54e-15
	SD	1.772	0.895	5.12e-15
	TE	6.494	5.981	5.610

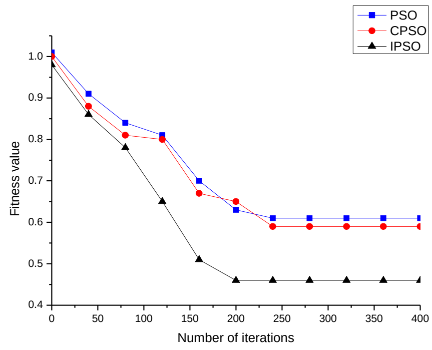
The following respectively selected four benchmark functions to compare the convergence of three algorithms, the comparison results see Figure 1：



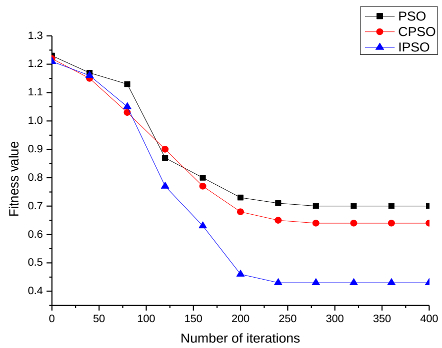
(a) Function Convergence of Schwefel



(b) Function Convergence of Rosenbrock



(c) Function Convergence of Griewank



(d) Function Convergence of Ackley

Figure 2. Compares the Convergence of Benchmark Functions

From figure 2 analysis that (a), (b) the overall convergence curves are ideal; early decline speed advantage is not obvious, but the late has the obvious advantages compared to the other two algorithms. Mainly because of the chaotic mechanism is introduced in the algorithm, at the same time, the particle is quantized, reduce parameters, and for each particle's state to determine, as the iteration proceeds, the traditional particle swarm algorithm is easy to fall into local optimum, poor convergence, while the ICQPSO algorithm by means of determining the particle's state, makes some particle's velocity of poor performance is improved. Therefore, when the local optimum is very close to the global optimum in the late of the algorithm search, this strategy can not only guarantee the stability and precision of convergence, but also can improve the convergence speed.

5. Conclusion

The local search ability is not strong for the traditional particle swarm optimization algorithm, which easy to fall into local optimal. Based on this, this paper presents a quantum particle swarm algorithm with chaos. The algorithm is introduced into the chaos mechanism, enhance the search ability and diversity of particles; the particle is quantized, reduce the parameters in the algorithm to improve the operation efficiency; in the process of algorithm iteration, to determine the state of the particle and to deal with the particle of poor performance, which is innovation point in this paper. Algorithm set two particle state parameters: oscillation factor and stagnation operator, and then integrate the two, planning the particle performance indicators, and set a threshold to determine particles, change the status about the particle of poor performance, which can not only guarantee the stability and precision of algorithm, and can effectively improve the convergence speed. The experiment shows that the improved ICQPSO algorithm convergence ascending is obvious.

References

[1] M. G. Epitropakis, V. P. Plagianakos and M. N. Vrahatis, "Evolving cognitive and social experience in Particle Swarm Optimization through Differential Evolution", a hybrid approach, Information Sciences, vol. 216, **(2012)**, pp. 50-92.

[2] I. L. Schoeman and A. P. Engelbrecht, "A novel particle swarm niching technique based on extensive vector operations", Natural Computing, vol. 9, no. 3, **(2010)**, pp. 683-701.

[3] A. Khare and S. Rangnekar, "A review of particle swarm optimization and its applications in Solar Photovoltaic system", Applied Soft Computing, vol. 13, no. 5, **(2013)**, pp. 2997-3006.

[4] K. Yang, K. Maginu and H. Nomura, "Cultural algorithm-based quantum-behaved particle swarm optimization", International Journal of Computer Mathematics, vol. 87, no. 10, **(2010)**, pp. 2143-2157.

[5] S. Rana, S. Jasola and R. Kumar, "A review on particle swarm optimization algorithms and their applications to data clustering", Artificial Intelligence Review, vol. 35, no. 3, **(2011)**, pp. 211-222.

[6] K. Thanushkodi and K. Deeba, "On performance analysis of hybrid algorithm (improved PSO with simulated annealing) with GA, PSO for multiprocessor job scheduling", WSEAS Transactions on Computers, vol. 10, no. 9, **(2011)**, pp. 287-300.

[7] E. Araujo and L. S. Coelho, "Particle swarm approaches using Lozi map chaotic sequences to fuzzy modelling of an experimental thermal-vacuum system", Applied Soft Computing, vol. 8, no. 4, **(2008)**, pp. 1354-1364.

[8] A. Safari and H. Shayeghi, "Iteration particle swarm optimization procedure for economic load dispatch with generator constraints", Expert Systems with Applications, vol. 38, no. 5, **(2011)**, pp. 6043-6048.

[9] S. Mukhopadhyay, S. Banerjee, "Global optimization of an optical chaotic system by Chaotic Multi Swarm Particle Swarm Optimization", Expert Systems with Applications, vol. 39, no. 1, **(2012)**, pp. 917-924.

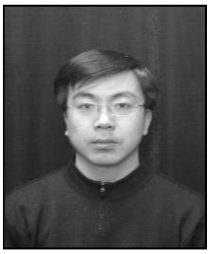
[10] M. Pluhacek, R. Senkerik, D. Davendra, Z. K. Oplatkova and I. Zelinka, "On the behavior and performance of chaos driven PSO algorithm with inertia weight", Computers & Mathematics with Applications, vol. 66, no. 2, **(2013)**, pp. 122-134.

[11] S. Ch, N. Anand, B. K. Panigrahi and S. Mathur, "Streamflow forecasting by SVM with quantum behaved particle swarm optimization", Neurocomputing , vol. 101, **(2013)**, pp. 18-23.

Authors

C. N. Zhang, received the Master's degree in computer application technology from University of Science and Technology Liaoning, in 2007. Currently, she is a lecturer at School of Software Engineering at University of Science and Technology Liaoning. Her research interests include Distributed computing and data mining.

Y. R. Li, received the Master's degree in computer application technology from University of Science and Technology Liaoning, in 2008. Currently, he is a lecturer at School of applied technology college at University of Science and Technology Liaoning. His research interests include Distributed computing and data mining.



J. F. LI, received the Master's degree in mechanical design and application from University of Northeastern University Liaoning, in 2005. Currently, he is a engineer at Cold rolling mill of Angang New Steel Limited Company Liaoning. Her research interests include Optimization design and Mechanical equipment.