TELKOMNIKA Indonesian Journal of Electrical Engineering
Vol.12, No.1, January 2014, pp. 38 ~ 47
DOI: http://dx.doi.org/10.11591/telkomnika.v12i1.3368

A Chaos Cloud Particle Swarm Algorithm Based Available Transfer Capability

Hongsheng Su*, Ying Qi
Dept. of Electrical Engineering, Lanzhou Jiaotong University, Lanzhou, China
88 West Aning Road, Lanzhou 730070, China
*Corresponding author, e-mail: 15002606147@163.com

Abstract

A mathematical model for ATC based on optimal power flow was built under the static security constraints, where the maximum of all load nodes in receiving area was considered as aim function. In view of the defects of slow convergence and low accuracy in ATC optimization algorithms, a chaos cloud particle swarm optimization algorithm based on golden section criteria (CCGPSO) was proposed. This method classified the particle swarm into three populations based on golden section judge principles according to fitness level. They are called chaos cloud particles, cloud particles and standard particles respectively. Each population was operated by different processing operations and updating modes. Comparing with other methods, the numerical simulation results of CCGPSO in IEEE-30 bus system demonstrate the higher efficiency and validity in ATC calculation. It is more suitable for solving the large-scale non-linear multi-constraint engineering practical problems.

Keywords: *Available Transfer Capability (ATC), particle swarm optimization algorithm, golden section, chaos algorithm, cloud theory*

1. Introduction

Available Transfer Capability (ATC) is the remaining transfer capability in the actual physical transmission network, which can be used for commercial transactions on the basis of existing transmission contracts [1]. In the electricity market, ATC is an important indicator to measure the safe and reliable operation of power grid, and also it is the key condition to ensure the implementation of electric power transactions. Modern power system has developed into the large-scale AC-DC interconnected system. There are a large number of frequent changes in transactions, which make the transmission system overloads, circulation increases, capacity margin decreases. The security and stability issues of power system become more and more prominent. The regional power grid companies want to take advantage of the existing transmission network to deliver more power, in order to achieve the optimized allocation of global resources. Therefore, the calculation of available transfer capability has been a significant problem. It can not only display the power system security and stability margin to reduce the blocking probability of occurrence, but also it can guide the behavior of the market participants.

According to whether the ATC is regarded as a random variable, the research method is summed up as two kinds: one is the probabilistic algorithm; the other is the deterministic algorithm. The probability algorithm [2] uses the theory of probability and mathematical statistics analysis to determine the available transmission capability. Considering many uncertain factors of system, but it is difficult to take into account the continuous state changes of the power system, it is not suitable for engineering calculation. The deterministic algorithm, that does not consider the random probability in the case of ATC. This kind of algorithm is used to calculate the transmission capability between two regions on the basis of the system is in a certain state. It includes the continuous flow method [3], optimal power flow method [4] and sensitivity analysis, etc. Optimal power flow method not only can deal with all kinds of system constraints quickly, but also optimize and scheduling the system resources. It is very suitable for ATC calculation

The study of ATC algorithm has very important practical significance and theoretical value, experts and scholars, market participants have carried out a great deal of research work.

Received May 29, 2013; Revised July 16, 2013; Accepted September 9, 2013

A lot of optimization techniques are applied to ATC calculation and have obtained some achievements. In the reference [5], an improved particle swarm algorithm was put forward to calculate the ATC. The algorithm adopted the strategy of adaptive inertia weight adjustment with phase decrease and achieved coordination with global search and local search. But the issues of the adaptive weighting parameter, dissipation factor and other parameters to be further solved. In the reference [6], the genetic algorithm (GA) was used to establish an ATC model. Although GA has an excellent performance in solving large-scale nonlinear constrained optimization problem, its operation is complicated and the calculation time is too long. It is not conducive to engineering applications. Reference [7] used a particle swarm algorithm based on chaos cloud model. The particle was divided into the excellent particles and normal particles. The cloud model algorithm and excellent particles were applied to search the optimal solution in a state of convergence. The chaos algorithm and normal particles were applied in the outside space of the convergence region. This process mode improved the accuracy and speed of the algorithm. But the particle can not be divided completely accurate by judging the convergence state, because the particles distribution location and number are non-uniform. Reference [8] constructed an ATC model using the Benders decomposition method. It presented that partition the static security constrained ATC problem into a base-case master problem and a series of sub-problems relevant to various contingencies. This method reduced the dimension of the optimization problem and improved the efficiency. Different algorithms have their respective advantages and disadvantages. The algorithm using a single search mechanism is simple, but there are some defects in processing large-scale multi-constraint nonlinear optimization problems, such as poor robustness and easy to fall into local optimum. Therefore, it is an inevitable trend to solve the ATC problem by combining suitable optimization algorithm with PSO and give full play to their respective advantages.

We present the chaotic cloud particle swarm algorithm after summarizing the existing research achievement, which solves ATC problem by using the optimal power flow model under static security constraints [9]. Firstly, we introduce chaotic variables into PSO to search entire spaces, in order to avoid falling into local extremum. Secondly, according to the particle swarm fitness value, we use the golden section grouping criteria to divide particle swarm into three sub swarms, standard particles, chaotic cloud particles and cloud particles. Each population has different processing operations and updating modes. Since the particles' distribution without regularity and randomness, the general optimization method can not search the entire solution space completely. The golden section enables particle swarm is classified accurately, greatly improving the speed and precision of the optimization.

2. Mathematical Model of Available Transfer Capability

In a specific operating state, ATC is referred to as the maximum transmission increment under the premise of not influencing the existing power transactions and violating the operational limits in system. At the same time, non-sending area generator active power output and non-receiving area load are constant. Besides the sending area generator active output and receiving area active load both increase[10]. This paper adopts the optimal power flow model based on static security constraints. OPF takes the maximum active output of load nodes in receiving area as the objective function and the power flow equations as equality constraints,the branch load, the output voltage and various stability constraint conditions of generator as inequality constraints in solving ATC problem. From a mathematical point of view, the ATC problem is described into a pure nonlinear programming mathematical problem.

$$A_{\mathrm{ATC}} = \max\left(\sum_{i\in\Gamma_{im}} \Delta P_{\mathrm{L}i} \right) \tag{1}$$

Equality constraint equation:

$$\begin{cases} P_{\mathrm{G}i} - P_{\mathrm{L}i} - V_i \sum\limits_{j=1}^{n} V_j (G_{ij} \cos\theta_{ij} + B_{ij} \sin\theta_{ij}) = 0 \\ Q_{\mathrm{G}i} - Q_{\mathrm{L}i} - V_i \sum\limits_{j=1}^{n} V_j (G_{ij} \sin\theta_{ij} - B_{ij} \cos\theta_{ij}) = 0 \end{cases} \quad (2)$$

Inequality constraint equations:

$$\begin{cases} P_{Gi}^{\min} \le P_{Gi} \le P_{Gi}^{\max} \\ Q_{Gi}^{\min} \le Q_{Gi} \le Q_{Gi}^{\max} \\ P_{Li}^{\min} \le P_{Li} \le P_{Li}^{\max} \\ Q_{Li}^{\min} \le Q_{Li} \le Q_{Li}^{\max} \\ -\overline{S_{ij}} \le S_{ij}^{E} \le \overline{S_{ij}} \end{cases} \quad (3)$$

In (1), $\Delta P_{\mathrm{L}i}$ represents load active increment of node i, Γ_{Lm} represents receiving area load nodes collection. In (2) and (3), $P_{\mathrm{G}i}$, $Q_{\mathrm{G}i}$ are generator active, reactive power. $P_{\mathrm{L}i}$, $Q_{\mathrm{L}i}$ represent active and reactive power of load nodes; V_i, θ_i are the nodes voltage amplitude and phase angle, S_{ij} is the line heat stability constraint from endpoint i to j.

3. Particle Swarm Optimization

Particle swarm optimization algorithm is a kind of evolutionary computation technology. Its application in power system includes the state estimation, reactive power optimization and voltage control [11] and so on. The basic idea of PSO is find the optimal solution through cooperation and information sharing between the groups. PSO is derived from birds foraging in the process of migration and swarm behavior simulation. Each particle determines their search direction and search scope by their speed. This speed is dynamically adjusted by the individual flight experience and group flight experience. In a searching space, each particle records and follows the current optimal particle in solution space.

Let X and V denote the particle's position and its corresponding velocity in search space respectively. Therefore, the i-th particle is represented as $X_i=(X_{i1},X_{i2},\ldots,X_{in})$. The velocity of particle is represented as $V_i=(V_{i1},V_{i2},\ldots,V_{in})$. The best previous position of the i-th particle is recorded and represented as $P_i=(P_{i1},P_{i2},\ldots,P_{in})$. Its fitness is *Pbest*. The best one among all the particles in the population is represented as $P_g=(P_{g1},P_{g2},\ldots,P_{gn})$. The fitness is *gbest*.

For each iteration, particles update their velocity and position according to the following formulas:

$$V_i^{k+1} = \omega V_i^k + c_1 r_1 (P_i^k - X_i^k) + c_2 r_2 (P_g^k - X_i^k) \quad (4)$$

$$X_i^{k+1} = X_i^k + V_i^{k+1} \quad (5)$$

Where ω is the inertia weight coefficient; c1, c2 are acceleration factors and generally set to 1.5~2.0. r1, r2 are uniform random value in the range [0, 1]. x_i^k is the current position of particle. v_i^k is the current velocity of particle.

From (4), we can see that the first item on the right of equation is the inertia velocity of particle, which reflects the memory behavior of particle. The second item is the "cognition", which represents the private thinking of the particle itself. The third item is the "society", it reflects the information sharing and mutual cooperation among groups. The inertia weight ω largely influences the balance between PSO local search ability and the global search ability. The particles search the optimal solution by cooperation and competition.

4. Cloud Model Algorithm

The cloud model is an uncertainty conversion model between the qualitative knowledge and quantitative expression, and it greatly reflects the relationship between the randomness and fuzziness. Let *U* is a quantitative domain and *C* is a qualitative concept of *U*. If the value $x \in U$ is a random realization of qualitative concept *C*, and the certainty degree of *x* to *C* is a random number that has stable tendency, the distribution of *x* in theory field *U* is known as cloud, recorded as cloud *C*(*x*). Every *x* is called a cloud drop [12]. The cloud model can be represented integrally by the expected value E_x, entropy E_n and hyper entropy H_e.

Digital characteristic parameters for the cloud theory:

(1) Expected value E_x represents distribution of cloud droplets.

(2) Entropy E_n is an uncertainty measure of the qualitative concept, determined by the fuzziness and randomness of the nature concept. It reflects the discrete degree of the cloud droplets.

(3) Hyper entropy H_e is the dispersion degree of entropy. It is determined by the fuzziness and randomness of the entropy E_x.

The cloud generator has established the mapping relationship of interdependence between qualitative and quantitative. Given a group of cloud droplets that conform to normal distribution as sample (x_i, m_i), generate three groups digital values (E_x, E_n, H_e). When the three eigenvalues are given, it is called the X condition cloud generator [13].

The method of producing the X condition cloud generator:

(1) To generate the normal random number $E_n{}'$ with expected value E_n, variance H_e;

(2) Calculate $C_T\left(x_i\right) = \exp\left[-\frac{(x_0 - E_x)^2}{2\left(E_n^{'}\right)^2}\right]$;

(3) (x_0, $C_T(x_i)$) is a cloud droplet. Repeat the above steps, until produce N cloud droplets.

5. Chaos Optimization Algorithm

The motion state with the randomness obtained by the deterministic equation is called chaos. The chaos is a universal phenomenon in nonlinear system, widely exists in the natural and social phenomenon, whose behavior is complex and seemingly random. But in fact, it has ordered regularity. The chaotic variables have the following characteristics: randomness, it behaves clutter as random variables [14]. Ergodicity, that is, which can traverse all states as own rules without any repeatability. Regularity, namely this variable is derived from iterative equations. Chaos optimization algorithm is a novel optimization method. It utilizes the ergodicity of chaos system to achieve the global optimum. Moreover, it does not require the objective function satisfy the continuity and differentiability properties, so a technology based on chaotic search is more superiority than the other random search.

Compared with other equations, the logistic equation is simpler, has smaller amount of calculation. So we use this equation to construct chaotic sequence:

$$x_{i+1} = \mu x(1 - x_i) \tag{8}$$

$\mu \in [3.57,4]$, is called logistic parameter. In this interval, the system is in the chaotic region, particle trajectories exhibit chaotic characteristics[15].

Therefore, after each iteration, each particle does the chaos ergodic motion by chaotic particle swarm, in order to make the whole particle swarm search all solution spaces, and will not stay in local optimum of the extreme point.

6. Chaos Cloud Particle Swarm Optimization Algorithm Based on Golden Section (CCGPSO)

6.1. CCGPSO Algorithm Description

With all particles flying toward the optimal solution in the PSO optimization process, parts of particles fly away from the initial solution. More and more particles lose the diversity, which makes the later convergence speed become observably slow. So the chaos is introduced into the particle swarm algorithm. Firstly, a set of chaotic variables are produced with the same number of optimized variables. Secondly, chaos is introduced into optimization variables by means of carrier. Finally, the ergodic range of chaotic motion is enlarged to the range of optimization variables, then chaotic variables can be used to search optimal solution directly.

But a lot of chaos transformation and inverse transformation would greatly increase the amount of calculation; result in the calculation become slower. The cloud algorithm has a fast global searching ability. So the chaotic particle swarm optimization algorithm is combined with the cloud theory in this paper, cooperate with the golden section simultaneously, which is based on fitness evaluation criteria. According to the fitness value, the particle is divided into three parts. Each part applies different processing operation and updating mode to enhance its search ability to overcome the shortcoming of dependent on initial value and easily fall into local extreme.

The particles have different fitness value, and their distribution position and quantity are not identical. So it is not accurate to divide particle only from the average fitness of all particles. Therefore, we adopt the golden section point to evaluate the fitness in this paper.

First of all, we calculate the average fitness value f_{avg} of all particles and find out the optimal fitness f_{min}, the worst fitness f_{max}, two assessment requirements $f_{golden1}$ and $f_{golden2}$ are established in accordance with the golden section[16], that is

$$f_{\text{avg}} = \frac{1}{n}\sum_{i=1}^{n} f_i \tag{9}$$

$$f_{\text{range}} = f_{\max} - f_{\min} \tag{10}$$

$$f_{\text{golden1}} = 1.618 f_{\text{avg}} \tag{11}$$

$$f_{\text{golden 2}} = f_{\min} + f_{\text{range}} / 1.618 \tag{12}$$

The golden section evaluation criteria are as follows:

$$f_{\text{GCPO 1}} = \begin{cases} f_{\text{golden 1}}, & f_{\text{golden 1}} \geq f_{\text{golden 2}} \\ f_{\text{golden 2}}, & f_{\text{golden 1}} < f_{\text{golden 2}} \end{cases} \tag{13}$$

$$f_{\text{GCPO 2}} = \begin{cases} f_{\text{golden 1}}, & f_{\text{golden 1}} < f_{\text{golden 2}} \\ f_{\text{golden 2}}, & f_{\text{golden 1}} \geq f_{\text{golden 2}} \end{cases} \tag{14}$$

The particles whose fitness values are higher than f_{GCPO1} are called standard particles. They are closer to the optimal solution, so they can update their speed and displacement according to the basic particle swarm algorithm. Particles whose fitness values are lower than f_{GCPO2} are called chaos cloud particles. They are far from the optimal solution, so they need to be operated by the chaotic particle swarm calculation first and then the cloud calculation. The particles will traverse all solution spaces in this way. The particles whose fitness values are between above two populations are called cloud particles. Their distance from the optimal solution are moderate, the X condition cloud generator enables each particle adapt inertia weight dynamically.

6.2. The Algorithm Process

(1) Input the original data, obtain the system nodes and branches information, acquire the range of the control variables. Set the maximum number of iterations, produce the particle swarm and its initial position and velocity, get individual optimal value and global optimal value by calculating each particle's fitness in current position.

(2) Chaos initialize population: randomly generate n vectors range between 0 to 1, that is $x_1=(x_{11},x_{12},\ldots, x_{1n})$. According to the logistic equation $x_{i+1}=\mu x_i(1-x_i)$ to obtain n chaos variables, and the chaos variables range is enlarged to the corresponding range of the optimization variables.

(3) Load flow calculation for each initialization particle respectively, balance node to be selected in sending area. We calculate state variables of the system running state to obtain the fitness value and individual extremum, compared to get global extremum.

(4) Calculate the average fitness value f_{avg} of the entire particle swarm, establish two evaluation criterias f_{golden1}, f_{golden2}. The particle swarm is divided into three parts: chaos cloud particles, cloud particles, standard particles according to the golden section evaluation criteria. If particles are the chaos cloud particles, go to step (5) and then (6); If particles are the cloud particles, go to step (6); If particles are the standard particles, go to step (7).

(5) Chaos optimization for optimal location. Using the following formula: $x_i=(x_{11},x_{12},\ldots,x_{1n})$ $x_i=(p_{gi}-a_i)/(b_i-a_i)$ (a_i and b_i is the range of optimization variables), map the optimal position to the logistic equation defined domain [0,1]. According to the logistic equation produce m chaotic variables, each chaotic variables re-transformed into the optimization variables by doing reverse transformation and obtain the m variables after several iterations .

(6) Make use of the X condition cloud generator, enable each particle adapt inertia weight dynamically and adjust the displacement and velocity dynamically.

(7) Update the displacement x_{id}^{k+1} and velocity v_{id}^{k+1} of the particles, set inertial parameters ω, modify speed and displacement of cross-border particles state variables in accordance with the standard particle swarm.

(8) Calculate the fitness value, compare it to the current individual optimal value. Set the minimum value of the individual optimal solution of all particles as particle swarm global optimal solution P_g.

(9) The termination condition judgment: If the current iterations reach the maximum, conduct the step (8) and output target function value; if not satisfied, then set the iterations $k=k+1$ and return to step (3).

7. Example Analysis

Simulation calculation is carried out in IEEE-30 node system. The system has 41 branches, 6 generators, 22 load nodes. It is divided into three regions, each region has 2 generators, 7 interconnection lines. Figure 1 is IEEE-30 nodes system, where node 1 is a balance node, it is used to balance the whole network power; 2, 13, 22, 23, 27 are PV nodes; The remaining nodes are PQ nodes. The specific network structure as shown in figure 1.

In this paper, ATC calculation and simulation is accomplished in MATLAB. Its parameters are set as follows: the generator node active power is 100MW, and take the trend value of the ground state as the original variables. Particle swarm algorithm parameters: particle swarm size is 50, the maximum iterations is 100, inertia weight ω_{max}= 0.9, ω_{min}= 0.4, learning factors $c_1=c_2=2.0$; Cloud algorithm parameters: set global optimal position as expected value E_x, current fitness variance δ^2 as entropy E_n, take E_n /10 as hyper entropy value H_e; Chaos algorithm parameters: logistic parameter $\mu=4$.

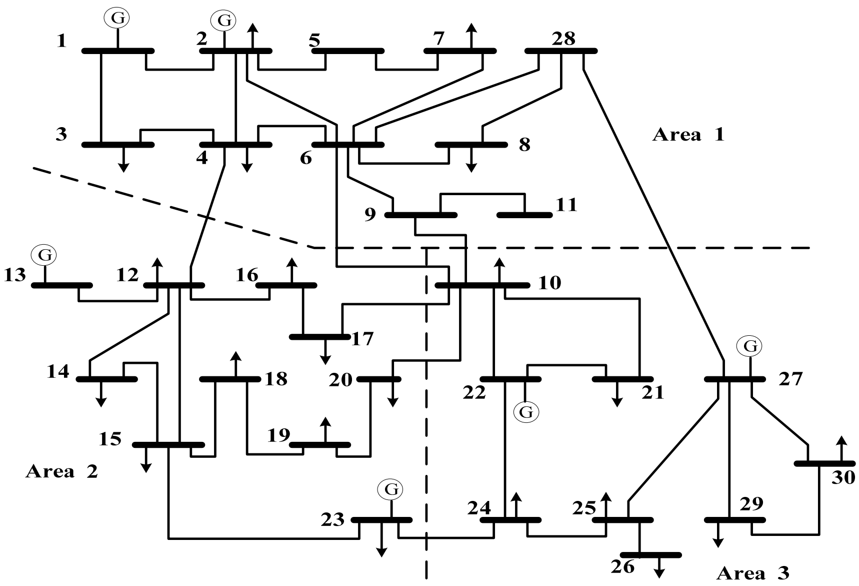

Figure 1. IEEE-30 node system

Table 1. Comparison of three algorithms calculation optimal solution

sending area -receiving area	ATC/MW		
	Chaos-PSO	Cloud-PSO	CCGPSO
1-2	104.75	105.72	109.53
2-1	34.32	54.15	57.83
1-3	102.75	104.59	103.07
3-1	57.17	73.01	92.85
2-3	31.21	33.23	46.98
3-2	53.07	62.53	74.54
total	383.27	433.23	484.80

Results in the table are average value after 20 times calculation. Seen from Tab.1, during the transmission from region 1 to region 3, the ATC value calculated by CCGPSO is slightly smaller than the value obtained by cloud particle swarm optimization, but CCGPSO obtained the gross of regional optimal solutions is obviously superior to the value acquired from chaotic particle swarm algorithm and the clouds. This shows that, in the same transmission network as well as the allocation of resources, CCGPSO is able to utilize existing resources to deliver more power, its practicality advantage in the power system is more prominent. Take transmission region of $ATC_{3\text{-}1}$ for an example, the chaotic particle swarm and cloud algorithm get into premature convergence in ATC calculation, they have the poorer searching ability than CCGPSO.

Table 2 lists the related value of ATC from area 2 to 3, the corresponding convergence properties contrast curve as shown in Figure 2. From Table 2, it is known that variance of CCGPSO is smaller after 20 times calculation. It shows that the algorithm has strong stability; From the iterations view, CCGPSO convergence need iterate 25 times, Cloud-PSO need 9 times, this is due to the fusion of three optimization algorithms make CCGPSO has better global search and the ability to overcome local extremum. However, this algorithm increases the amount of calculation, the iterations is slightly increased.

Table 2. Comparison of three algorithms calculation results

algorithms	ATC$_{2-3}$/MW		
	average value (MW)	sample variance (MW)	iterations
Chaos-PSO	27.31	7.23	40
Cloud-PSO	31.74	6.14	10
CCGPSO	41.23	3.65	25

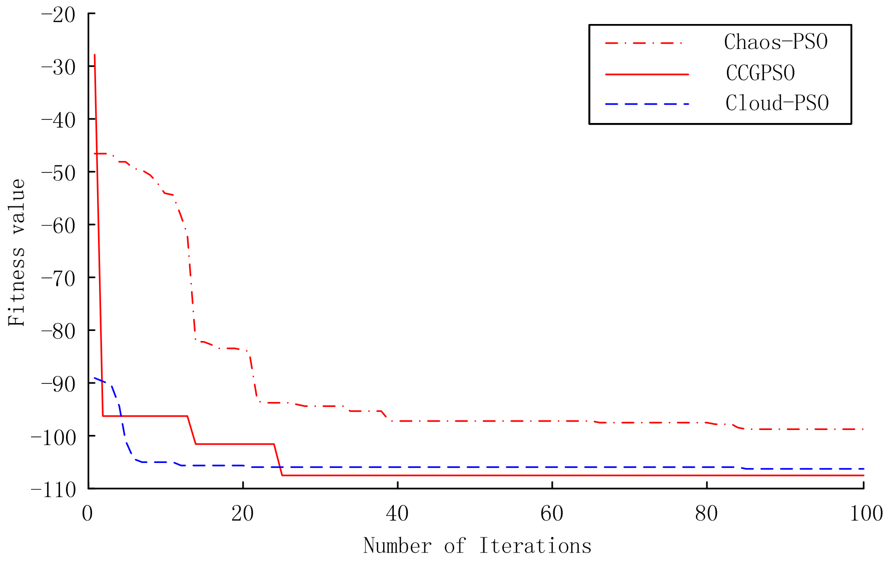

Figure 2. Three algorithms convergence curve between areas 2-3

Table 3 lists the related value of ATC from region 2 to 1, the corresponding convergence properties contrast curve as shown in Figure 3.

Table 3. Comparison of three algorithms calculation results

algorithms	ATC$_{2-1}$/MW		
	average value (MW)	sample variance (MW)	iterations
Chaos-PSO	31.34	11.09	11
Cloud-PSO	48.78	9.66	38
CCGPSO	53.34	5.56	29

From table 3, it is known that variance of CCGPSO is smaller than others, which shows that the algorithm has stronger stability; CCGPSO has more long iterative time, this is because in search later period particle swarm need time to divide particle into three populations. When three populations start to search the optimal solution together, the operation processes become slightly complicated and make iterations increase. However, CCGPSO obtained the maximum average value just last 20 times calculation; it implies that CCGPSO has higher global search ability and better accuracy.

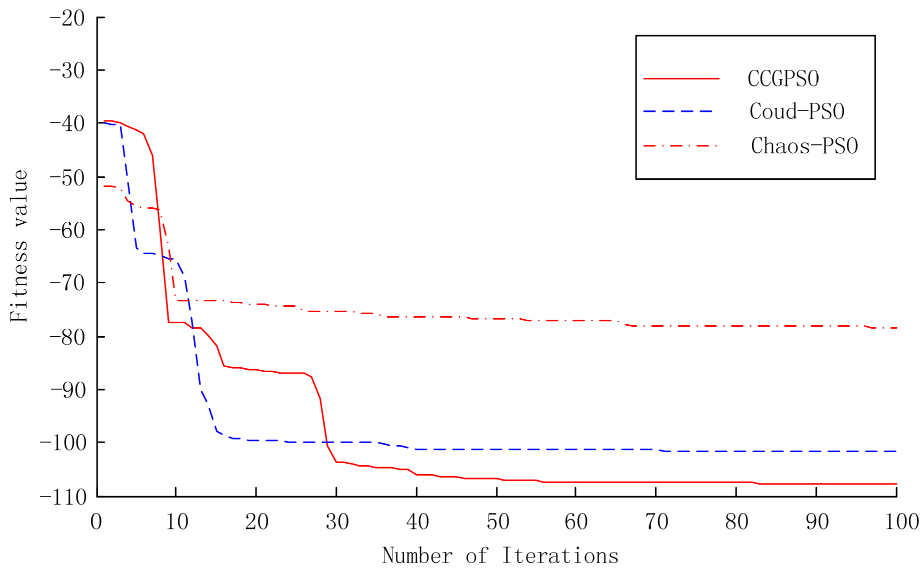

Figure 3. Three algorithms convergence curve between areas 2-1

The convergence curves of three algorithms in figure2 and figure3 show that, at the beginning of iteration, CCGPSO has larger slope, which implies that CCGPSO algorithm has continued and rapid convergence ability. But after a while it trapped into local extremum. This moment the golden section grouping criteria enable particle swarm quickly escapes from local extremum and start to group optimization. Each population continue to search optimal solution along the direction of the current global optimal solution, meanwhile they update their position and velocity according to respective method. Although this process elapses time slightly long, the convergence value is significantly better than the Cloud-PSO algorithm and Chaos-PSO algorithm, it shows higher accuracy and better searching ability. At the beginning, Cloud-PSO converges faster and has less iteration, but the horizontal part of the line in figure shows that search entries into a phase of stagnation, easily into a precocious period. Its accuracy is inferior to CCGPSO; the convergence extremum from Chaos-PSO algorithm is far less than CCGPSO, it has lower accuracy.

8. Conclusion

This paper presents the chaos cloud particle swarm optimization algorithm based on the golden section evaluation criteria. This method divided the particle swarm into standard particle, chaos-cloud particle and cloud particle using the golden section judge principles according to the fitness value, each population is operated by different algorithm. This algorithm solves the problems of easily falling into local optimum in basic PSO and repeatedly search parts of solutions in chaos optimization. The algorithm proposed in this paper has higher accuracy and quicker convergence speed in ATC calculation and it can make full use of power resources. So it has more prominent practical value in the large-scale non-linear multi-constraint engineering problems.

References

[1] Zailin Piao, Dongdong Wang, Jun Wang. Available Transfer Capability Calculation Using Improved Particle swarm Optimization Algorithm. *Journal of Shenyang Agricultural University*. 2012; 41(1): 110-112
[2] Gengyin Li, Ming Zhou, Yajing Gao. *Sequential Monte Carlo Simulation Approach for Assessment of Available Transfer capability*. Proceedings of the CSEE. 2008; 28(25): 74-79.
[3] Ejebe GC, Tonj J, Waight JG, et al. Available transfer capability calculation. *IEEE Trans on Power System*. 1998; 13(4): 1521-1527.

[4] Feng Wang, Xiaoming Bai. *Optimal Power Flow Based Transfer Capability Calculation*. Proceeding of The CSEE. 2002; 22(11): 35-40.
[5] Haitao Huang, Hua Zheng, Lizi Zhang. *Available Transfer Capability Research Based on Modified Particle Swarm Optimization Algorithm*. Proceeding of the CSEE. 2000; 26(20): 45-49.
[6] Xiaoxiong Liu, Juan Wang, Yan Wu, Yu Liu. The Optimization of Lateral Control Augmentation Based on Genetic Algorithm. *TELKOMNIKA Indonesian Journal of Electrical Engineering*.2013; 11(6): 2956-2961
[7] Chaolong Zhang, Chunri Yu, et al. Particle Swarm Optimization Algorithm Based on Chaos Cloud Model. *Journal of Computer Applications*. 2012; 32(7): 1951-1954.
[8] Shaaban Mohamed, Haoming Liu, Weixing Li, et al. *ATC Calculation with Static Security Constraints Using Benders Decomposition*. Proceedings of CSEE. 2003; 23(8): 7-11.
[9] Feng Wang, Xiaoming Bai. *Optimal Power Flow Based Transfer Capability Calculation*. Proceeding of The CSEE. 2002; 22(11): 35-40.
[10] Don H, Park JK, Balho HK, et al. Security Constrained Optimal Power Flow for The Evaluation of Transmission Capability on Korea Electric Power System. Power Engineering Society Summer Meeting. Seattle. 2001; 476-482.
[11] Fukuyama Y, Yishida H. *A Particle Swarm Optimization for Reactive Power and Voltage Control in Electric Power System*. 2001 Congress on Evolutionary Computation. Korea. 2001: 87-93.
[12] Hongsheng Su. Distribution Grid Reactive Power Optimization Based on Improved Cloud Particle Swarm Algorithm. *TELKOMNIKA Indonesian Journal of Electrical Engineering*. 2013; 11(1): 468-475.
[13] Deyi Li, Haijun Meng, Xuemei Shi. Membership Cloud and Membership Cloud Generator. *Computer Research and Development*. 1995; 32(6): 15-20.
[14] Junmin Liu, Yuelin Gao. Chaos Particle Swarm Optimization Algorithm. *Computer Research*. 2008; 28(2): 322-325.
[15] Tao Zhao, Xinyin Xiong, Yaowu Wu. Reactive Power Optimization of Distribution System Based on Chaos Algorithm. *Relay*. 2003; 31(3): 20-25.
[16] Chao Wang. The Research on The Power Distribution Network Reactive Power Planning Based on The Chaotic Hydrometeor Particles Swarm Algorithm. Shanghai Jiao Tong University; 2011.