Parallel Cooperative Particle Swarm Optimization Based Multistage Transmission Network Planning

YI-XIONG JIN[1]; JUAN SU[2]
1. Department of Electrical Engineering; 2. Department of Information Center of Library
Shanghai University of Electric Power
Yang-pu District, Shanghai, 200090
CHINA

Abstract: - A cooperative hierarchy model for multistage transmission network planning is presented. Both the pertinence and independence of every stage are taken into account and the localized deterioration phenomenon of meta-heuristics algorithms is prevented in this model. A parallel cooperative particle swarm optimization is also put forward, whose optimization functions and decision variables are different in every processor, which makes the algorithm can deal with the contradiction of different planning stage, so better search direction and calculation speed are obtained.

Key-Words: - Multistage, Transmission network planning, Particle swarm optimization, Parallel algorithm

1. Introduction

Multistage Transmission Network Planning is a discrete, multimodal and nonlinear optimization problem for large-scale system. Its task is to specify where, how much and when new circuit for expansion should be installed at the lowest construction and operation cost on the premise of security. The research about this problem mainly focuses on two points: one is high dimension problem. The number of decision variables and constraint equations of MTNP is very large; the other is the coordination problem of different planning stage. A circuit can not be built on former stage while dismantled on later stage. The second problem is the main difference between MTNP and Single-stage Transmission Network Planning (STNP). Some methods such as Composition & Decomposition (C&D) algorithm [1], stratified method [2], Genetic Algorithm (GA) and parallel Ant Colony Algorithm(ACA) [3, 4], can improve the convergence performance of this problem effectively, but some following questions exist: C&D algorithm and stratified method divide the problem into several sub-problems, and iterate the sub-problems in turn. Since there is no general search guidance, mode loss occurred when the optimal process turn from one sub-problem to another. That is to say, mode variety lost when a sub-problem is started based on extreme point which is got by former sub-problem. For the purpose of prevent the mode loss, loose constraint method is usually adopted, however, to make certain the sound loose degree is not easy, and it will also enlarge the search space at the same time; parallel algorithm can improve its calculation speed in general, however, It doesn't consider the independence of different planning stage because there is not difference in its optimization function and decision variables of different processor. Both the pertinence and independence of every planning stage are taken into account in this cooperative hierarchy model for MTNP presented in this paper. A Parallel Cooperative Particle Swarm Optimization (PCPSO) is also put forward in this paper based on the hierarchy model. Other than former parallel algorithm, the optimization function and decision variables of PCPSO are different in different processor, which makes the algorithm can deal with the contradiction of different planning stage, so a better search direction and calculation speed are obtained.

2. PSO Algorithm

Particle swarm optimization was introduced in 1995 by Kennedy and Eberhart [5]. Several modifications to the original swarm algorithm have been made to improve its performance [6~11].

Process for implement the original PSO is as follows:

1) Initialize a population of particles with random positions and velocities on m dimensions in the problem space;

2) Evaluate the desired optimization fitness function in m variables for each particle;

3) Compare particle's fitness evaluation with particle's pbest. If current value is better than pbest, then sets pbest value equal to the current value and the pbest location equal to the current location in m-dimensional space;

4) Compare particle's fitness evaluation with population's overall previous best gbest. If current value is better than gbest, then resets gbest value equal to the current value, and the gbest location equal to the current location;

5) Change the particle's velocity and position according to equations (1) and (2) respectively:

$$v_{id}^{k+1} = wv_{id}^{k} + c_1 r_1 (p_{id} - x_{id}^{k}) + c_2 r_2 (p_{gd} - x_{id}^{k}) \quad (1)$$

$$x_{id}^{k+1} = x_{id}^{k} + v_{id}^{k+1} \quad (2)$$

$$(i = 1,2,\cdots n, d = 1,2,\cdots m)$$

Where: w is inertia weight; c_1 is cognition factor; c_2 is social-learning factor; r_1 and r_2 are random number between 0 and 1; the superscript is iterative generation; n is population size; m is particle's dimension; v_{id} and x_{id} are velocity and position of i^{th} particle on dimension index of d. p_{id} and p_{gd} are pbest and gbest positions on dimension index of d. The best position that every particle has achieved so far called pbest and the overall best position has achieved by all particles call gbest.

6) Loop to step 2) until a criterion is met.

3. MTNP Model

MTNP model is as follows:

$$\min \sum_{s=1}^{N} [\frac{1}{(1+r)^{y(s-1)}} u(x_s) + \frac{1}{(1+r)^{y(s)}} v(x_s)] \quad (3)$$

$$\begin{cases} P_i^s = V_i^s \sum_{j \neq i} V_j^s (G_{ij}^s \cos\theta_{ij}^s + B_{ij}^s \sin\theta_{ij}^s) \\ Q_i^s = V_i^s \sum_{j \neq i} V_j^s (G_{ij}^s \sin\theta_{ij}^s - B_{ij}^s \cos\theta_{ij}^s) \end{cases} \quad (4)$$

$$\begin{cases} -\overline{p}_i^s \le p_i \le \overline{p}_i^s \\ -\overline{q}_i^s \le q_i \le \overline{q}_i^s \end{cases} \quad i \in m + m^0 \quad (5)$$

$$0 < x_i^1 \cdots < x_i^s < x_i^{\max}, x \in Z, i \in m \quad (6)$$

In which: N is the number of planning stage; $u(x_s)$ is construction cost of stage s; x_s is expansion scheme of stage s; $v(s)$ is operation expense of stage s; r is interest rate; $y(s) = \sum_{l=1}^{s} g(l)$, $g(l)$ is years including in stage l, $y(0)=0$; formula (4) is power flow constraint of every stage; formula (5) is over load constraint of every stage; m is branch can add line; m^0 is branch cannot added line; formula (6) is upper and lower bound constraint of circuit number can be added on each branch; $x_i^{\max}$ is the upper bound of circuit number can be added on branch i; $0 \le x_i^1 \le \ldots \le x_i^s \le x_i^{\max}$ represents the planning of later stage is based on former planning scheme, and insures circuit having been built will not be removed afterwards. Power flow is represented by power flow calculation, and then represented by overload constraint; over load constraint of circuit can be represented by adding penalty item for overload of every circuit; formula (6) can be realized by following processes:

Update particle position and velocity in following equations:

$$\begin{aligned} v_{id}^{k+1} = Fix(&wv_{id}^{k} + c_1 r_1 (p_{id} - x_{id}^{k}) \\ &+ c_2 r_2 (p_{gd} - x_{id}^{k})) \end{aligned} \quad (7)$$

$$x_{id}^{k+1} = x_{id}^{k} + v_{id}^{k+1} \quad (8)$$

$$(i = 1,2,\cdots n, \quad d = 1,2,\cdots m)$$

In which: $Fix(f)$ is getting the integer part of f. When v_{id} is bigger than $v_{\max}$, make $v_{id} = v_{\max}$; When v_{id} is smaller than $v_{\min}$, make $v_{id} = v_{\min}$. $v_{\max}$ is often set at about 10-20% of the dynamic range of the variable on each dimension, and $v_{\min} = -v_{\max}$. When x_{id} is bigger than upper bound of circuits allowed to be added on a branch, then make x_{id} equal the upper bound. While $x_{id} < 0$, make $x_{id} = 0$.

Furthermore, the planning network must be connected, that is to say no isolated island is anticipated in planning scheme. Connectedness can be found out by topology search. When the network is unconnected, makes its fitness value equal to a very large penalty value.

As can be seen from the TNEP model, the relationship and difference between MTNP and STNP are mainly embodied in formula (6), and compared to STNP of the same network, the decision variables and constraints of MTNP are N times to that of STNP. Traditional optimization methods, such as linear programming and heuristic algorithm, are difficult to solve this problem. Meta-heuristic Algorithms (MH) developed in recent years, such as GA and ACA, et. al, shows great advantage in solving this problem. However, “two steps forward, one step back” phenomenon occurs [12]. A simple example to illustrate this concept follows. Consider a three-dimensional vector $X(x_1, x_2, x_3)$, and the fitness function is:

$$\min f(x) = x_1^2 + x_2^2 + x_3^2 \tag{9}$$

Its optimal point is (0, 0, 0). If the current best point searched is (10, 0, 10), its fitness value is 200, if the next iteration find point (5, 5, 5), its fitness value is 75. MH algorithm thinks the latter point is better than the former. It is caused by the fact that the fitness function is computed only after all the components in the vector have been updated to their new values. This means an improvement in two components (two steps forward) will overrule a potentially good value for a single component (one step back). In order to overcome this flaw, direct method is to optimize the three variables separately. However, if the three variables are associated, how to harmonize their relations is a critical problem. This phenomenon is more serious in high dimension optimization problem such as MTNP. A cooperative hierarchy model according to the independence of different planning stage and corresponding PCPSO to overcome this flaw is introduced in this paper.

4. PCPSO Based Cooperative Hierarchy MTNP

The interrelation of different planning stage is in formula (6). If taking this constraint out of consideration, every planning stage itself is a relatively independent optimal problem. So the following optimization model is put forward.

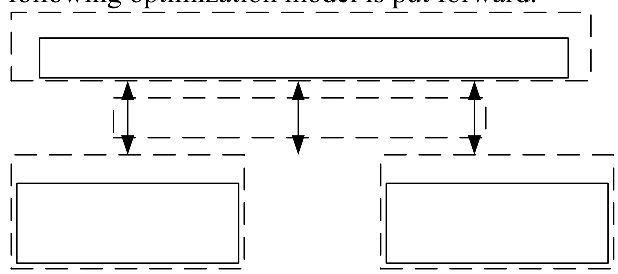

Fig.1 PCPSO based MTNP

The Optimization process is as follows:

a) Particle position and velocity initialization:

The initialization of PSO is usually random. However, we find out that many (about 1/2) network represented by the initial particles is unconnected; this makes many particles wasted. So, for the purpose of making the initial network connected, random topology-tree search is adopted: start from a random bus, randomly select a branch, add random number of circuit on this branch, perform the former random search process from the end bus of the branch, and so on, until all the buses having been searched. All initial particles formed by this random topology-tree search are connected. Furthermore, to ensure the planning scheme satisfy the mono-increasing constraint of formula (6), the added line number of every branch should be arranged from little to big according to the time order of planning stage.

b) The optimization process of server

1) Initialize f_s, f_1,… f_N and k to 0, f_s is new optimum found symbol of main PSO, f_1,… f_N are new sub-optima found symbol of every planning stages respectively; initialize particle position and velocity :

$$X_i^s = (x_{i1}^1, x_{i2}^1, \cdots x_{im}^1) \cdots (x_{i1}^N, x_{i2}^N, \cdots x_{im}^N);$$
$$V_i^s = (v_{i1}^1, v_{i2}^1, \cdots v_{im}^1) \cdots (v_{i1}^N, v_{i2}^N, \cdots v_{im}^N),$$
$$i = 1, 2, \cdots n,$$

n is particle’s population number; m is number of branch that can add line; send initial particle position and velocity of different planning stage to corresponding sub-PSO client;

$$X_i^s = (x_{i1}^s, x_{i2}^s, \cdots x_{im}^s);V_i^s = (v_{i1}^s, v_{i2}^s, \cdots v_{im}^s);$$
$$i = 1,2,\cdots n \ s = 1,2,\cdots N;$$

2) Power flow calculation of particles for every stage.
3) Calculate the total investment, net loss and penalty value for overload according to power flow results of corresponding planning stage, and then fitness value in all.
4) Update gbest and pbest of every particle. If new gbest is found, make k=0, f_s=1;else make k=k+1;
5) If k is bigger than given iteration times, go to step 7); else continue;
6) If any new sub-optimum found symbol f_i is 1(i=1,2…N), then read corresponding new sub-optimum to the same stage position of 5~8 particle who have the worst fitness value and set f_i to 0; update the other particles using equations (7) &(8); go to step 2);
7) Send end message to every client, and output optimization result.

c) The communication process of server

8) If f_s is 1, send connection-request message to every client; if confirm message is received from client, then transmit planning scheme of stage i-1, i and i+1 in gbest particle to i^{th} client (i=1…N). and set f_s to 0；
9) If connection-request message of client is received, send confirm message to client, receive sub-optimum from corresponding client, and set corresponding f_i to 1;
10) Go to step 8);

d) The optimization process of client

11) Set f_i to 0; read planning scheme of stage i-1 and i+1 as lower and upper bound of particles.
12) Power flow calculation of particles for i^{th} stage.
13) Calculate the i^{th} planning scheme's investment, net loss and penalty value for overload according to power flow results, and then fitness value.
14) Update gbest and pbest of every particle. If new sub-optimum is found, make f_i=1;
15) If end message is received, stop calculation; else continue;
16) Update particles' position and velocity using equations (5) &(6); go to step 12);

e) The communication process of client

17) If f_i is 1, send connection-request message to server; if confirm message is received from server, then transmit sub-optimum planning scheme to server, and set f_i to 0；
18) If connection-request message of server is received, send confirm message to server, and receive planning scheme of stage i-1 and i+1 as new lower and upper bound of particles;
19) Go to step 17)

There still have a problem in this optimization process, which is how to ensure the mono-increasing constraint of formula (6). Few researches have concerned about this problem. Circuit number added on every branch is arranged from little to big according to the time order of planning stage in paper [13] just as the particle initialization of this paper; however, this factitious ordering method will disturb the search process of PSO and decrease the search efficient. This paper adopts a repeatedly penalty relaxation method. Penalty function is as follow:

$$C\sum_{d=1}^{m}\sum_{s=1}^{N}\min\{0,[\sum_{j=1}^{s-1}(x_{id}^{i}-x_{id}^{j})+\sum_{k=s+1}^{N}(x_{id}^{k}-x_{id}^{i})]\}^{2} \tag{10}$$
$$(i = 1,2,\cdots n)$$

In which: C is penalty factor.

Adding the penalty (10) to the fitness function of MTNP can ensure the mono-increasing constraint. Moreover, repeatedly mask off penalty (10) in given iteration times to make particle able to search the border area of joint stages.

As can be seen from the above optimization process, PCPSO based MTNP cooperative hierarchy optimization algorithm take advantage of the relative independence of different planning stage, and also give attention to the interrelation of different stage by the server's optimization and assignment. "read corresponding new sub-optimum to the same stage position of 5~8 particle who have the worst fitness value" in step 6) instead of to the particle who have the best fitness value can not only provide optimal search direction for main PSO and prevent the "two step forward, one step back" phenomenon, but also prevent the mode loss of optimization. Since the sub-PSO provide optimal search direction, population size needed of main PSO is decreased, so quicker calculation speed and less computer memory consumption is achieved.

5. Simulation

Adopting three P4 1.7G processor (a server and two client) in 100M ether LAN, 256M DDR, VC6.0 programming, Socket communication, optimize the north-northwest Brazil system [14], which has 87 buses, 197 branches, two planning stages: 1998-2001; 2002-2008; particle dimension is 394; population size of main PSO is 100; mono-increasing constraint of circuit mask off 2 times in 40 times iterations; interesting rate is 0.1; optimization result is as follows:

Table.1 two stage transmission network planning result of north-northwest Brazil system

Stage 1(number in bracket is circuit number)	Stage 2(number in bracket is circuit number)
2-4,2-60,2-87(2),3-83,3-87,4-5,4-81,5-58(2),5-60,8-73,12-15,13-15(2),14-45,15-16(2),16-44 (3),16-61,18-50(6),18-74(3),20-21(2),20-38,22-58,24-43,25-55(2),27-53,30-31,30-63(2),32-33,33-67,35-51,36-46(2),40-45(2),41-64(2),43-55,43-58,48-49,49-50(2),54-58,54-63,61-64,61-85(2),67-68,67-69,67-71(3),71-72,72-73,73-74,75-82,75-83,76-82,78-80,79-82,80-83	1-2,4-5(2),4-81(3),5-38(1),5-58(2),13-15(2),15-16(2),15-46,16-44(3),16-61,18-50(5),18-74(3),20-21,20-38,20-66,22-58,25-55,26-54,29-30,30-31,35-51,36-39,36-46,43-55,43-58,48-50,49-50(2),52-59,61-85,65-66,65-87,73-74,75-81
Investment: $2228.707millon	Investment: $1031846*0.683=704.74419 million
Total investment:：$2933.451 million	

Compared to $2204.162 million in paper [15], the investment is increased, that is because some zero power injection nodes haven’t included in the result of paper [15], while this paper excluded the unconnected scheme. Compared to $3078.529 million calculated by parallel ACA in paper [4], the investment of this paper is decreased. Paper [4] excluded the unconnected planning scheme also, so this paper found better optimal result than paper [4] did.

Basic parallel PSO (PPSO) using the same optimization function and decision variables in different processor also simulated in this paper, in which three same computers are adopted. Comparison of PPSO and PCPSO is shown in tab.2

Table.2 Results comparison of basic PSO and PCPSO

	PPSO	PCPSO
Calculation times	50	50
Probability of success times	86%	96%
Average	3094.1	2988.2
Population size in every computer	100	100
Optimization time expend/s	347~413	189~27

As can be seen from tab.2, probability of converged times of PPSO is lower than that of PCPSO, average investment of basic PPSO in 50 times calculations is bigger than that of PCPSO, and calculation time of PPSO is about 2 times that of PCPSO in total 50 times calculation. Better performance of PCPSO is attributes to its sub-PSO, in which less decision variables are optimized, “two step forward, one step back” phenomenon is weakened and better search direction in current network structure context is supplied to main PSO.

6. Conclusion

This paper analyzed the merits and flaws of current optimization methods for MTNP, and put forward cooperative hierarchy model for MTNP. Both the pertinence and independence of every stage are taken into account and the localized deterioration phenomenon of meta-heuristics algorithms is prevented in this model. A parallel cooperative particle swarm optimization is also put forward, whose optimization functions and decision variables are different in every processor, which makes the algorithm can deal with the contradiction of different planning stage, so better search directions and calculation speed are obtained. Repeatedly penalty relaxation method makes particle able to search the border area of joint stages, and then widen the space can be searched.

References:

[1] Yuan Bin, Ma Wei-xin, Zheng Mei-Te. Composition and decomposition algorithm for large-scale system and its application in multistage transmission planning [J].

System Engineering-Theory Methodology Applications. Vol.3, No.4,1994, pp: 53-58.

[2] Cheng Hao_zhong, Gao Ci_wei, Ma Ze_liang, et. al. The stratified method for multi-object optimal electric power network planning [J]. *Proceedings of the CSEE*. Vol.23, No.10, 2003, pp: 11-16.

[3] Wang Xiu-li; Wang Xi-fan; Mao Yu-bin. Improved genetic algorithm for optimal multistage transmission system planning [C]. *Proceedings of the IEEE Power Engineering Society Transmission and Distribution Conference*. v 3, n SUMMER, 2001, pp: 1737-1742.

[4] Zhai Hai-bao, Cheng Hao-zhong, Lu Gan-yun, et. al. Parallel ant colony algorithm for the multistage transmission network optimum planning [J]. *Automation of Electric Power System*. Vol.28, No.20, 2004, pp: 37~42.

[5] Kennedy J, Eberhart R. Particle swarm optimization [C]. *Proceedings of IEEE International Conference on Neural Networks*, 1995(4), pp: 1942-1948.

[6] Y. Shi, R.C. Eberhart. A modified particle swarm optimizer [C]. *Proceedings of the IEEE International Conference on Evolutionary computation*. 1998; 69-73.

[7] Yuhui Shi, Russel C. Eberhart. Parameter selection in particle swarm optimization [C]. *Evolutionary Programming* VII. Springer: Berlin, 1998,pp: 591-600.

[8] R. C. Eberhart, Y. Shi. Comparing inertia weights and constriction factors in particle swarm optimization [C]. *Proceedings of the 2000 World Congress on Evolutionary Computation* 2000, pp: 84-88.

[9] Maurice Clerc. The swarm and the queen: Towards a deterministic and adaptive particle swarm optimization [C]. *Proceedings of the Congress of Evolutionary Computation* 1999.3, pp: 1951-1957.

[10] Morten Løvbjerg, Thomas Kiel Rasmussen, Thiemo Krink. Hybrid particle swarm optimizer with breeding and subpopulations [C]. *Proceedings of the third Genetic and Evolutionary Computation Conference* 2001.

[11] Jin Yi-xiong, Cheng Hao-zhong, Yan Jian-yong, et.el. Improved particle swarm optimization method and its application in power transmission network planning [J]. Proceedings of the CSEE. Vol.25, No.4, 2005, pp:46-50, 70.

[12] Frans van den Bergh, Andries P. Engelbrecht, A Cooperative approach to particle swarm optimization [J], *IEEE transactions on evolutionary computation*. Vol.8, No.3, pp: 225-239.

[13] Cheng Hao-zhong, *Zhang Yan. Power network planning methods and applications* [M]. Shanghai, science and technology publishing company of Shanghai, 2002.

[14] Romero R, Monticelli A, Garcia A, *et al*. Test systems and mathematical models for transmission network expansion planning [J]. *IEE Proc., Gener. Transm. And Distrib*. Vol.148, No.5, 2002, pp: 482-488.

[15] Escobar, A.H.; Gallego, R.A.; Romero, R. Multistage and coordinated planning of the expansion of transmission systems. *IEEE Transactions on Power Systems*. Vol.19, No.2, 2004 pp:735-744.

Supported by 2007 scientific research project of Shanghai Educational Commission (07ZZ145); The special fund for outstanding young teacher's scientific research of Higher Education of Shanghai; Shanghai leading academic discipline project (P1301); and keystone project of Shanghai Science and Technology Commission(061612040)