Transmission network expansion planning based on schema recording parallel Ant Colony Algorithm

ZHAI HAI-BAO, CHENG HAO-ZHONG
Department of Electrical Engineering
Shanghai Jiaotong University
1954, Huashan Road, Shanghai 200030
CHINA

Abstract In this paper, the insufficiencies of current optimization algorithms in solving large-scale transmission network expansion planning (TNEP) problems are discussed. Then, the mathematical model of TNEP is formulated, and the schema of its solution is studied detailedly. After the main disadvantages of traditional ant colony algorithm (ACA) are discussed, a schema recording parallel ant colony algorithm (SRPACA) is proposed. It can partition the solution space through schema recording, and can identify, record and jump away from the local optimal solution. The simulation results of two practical systems show that this algorithm has high computation efficiency and good local & global convergence.

Key-Words Transmission network expansion planning; Solution space partition; Schema recording parallel ant colony algorithm; Local search; Message Passing Interface (MPI)

1 Introduction

TNEP is a nonlinear, discrete, multi-peak, multi-objective and nondifferentiable problem. The main problem of solving this problem is that when the scale of the planning network is very large, the searching process always can’t explore the solution space thoroughly but converges at the local optimal solution (LOS).

Metaheuristic algorithms is a kind of popular methods to solve TNEP problems. Simulated annealing (SA)[1] algorithm, Tabu search (TS)[2] algorithm, Genetic algorithm (GA)[3] and Ant Colony Algorithm[4] have all been used to solve TNEP problems and gotten good effects. The core idea of all these methods is to control the direction of stochastic search by certain heuristic optimization strategies to achieve the optimization computation. Unfortunately, since none of these algorithms can guarantee the global convergence in limited time theoretically, they will always converge at a LOS, namely premature convergence. In fact, when dealing with the off-line TNEP problems, the ability of global search, or the ability of identifying and jumping away from the LOS and finding the global optimal solution (GOS), is a predominant factor to evaluate an optimization algorithm. SA algorithm does not identify the LOS, but it can jump away from the LOS by relaxing the searching limitations. But because it doesn’t have memory ability, the latter search may return to this LOS, which causes repetitive search. TS algorithm has excellent local search ability and can recognize the LOS and jump away form it by compulsively shifting the searching direction. It can also prevent the iterant search by recording some LOS in Tabu list. But because each element in the Tabu list is just a single spot, the recording efficiency of Tabu list is very low. To improve the recording capability, the scale of Tabu list must be increased, but this will consequently reduce the computation efficiency. GA uses the selection operator to control the searching direction. Because the selection operator usually chooses solutions in direct proportion to their fitness, those individuals which have very large fitness will be propagated repeatedly, while others will be eliminated. This would cause the schemata loss[5], which is inclined to premature convergence. The crossover operator can only strengthen the search of inner-schema, while can’t solve the problem of schemata
losing. It is only the mutation operator which can get back the lost schemata. However, too small mutation rate can’t prevent the schemata losing effectively, while too large mutation rate will cause eyeless stochastic search. ACA jumps away from the LOS on the probability determined by the state transition rule and controls the searching direction by locally and globally updating the pheromone intensities. Whereas, there are many factors that affect the convergence of ACA and also exists the confliction between the global convergence and the local search.

From the aforementioned discussion, it can be deduced that to design an excellent optimization algorithm, there are 5 aspects to be considered. 1) The partition of the solution space; 2) The identification of LOS and the thoroughly searching of its neighborhood; 3) The recording and effective jumping out of the LOS; 4) The reduction of iterate search; 5) The enhancement of searching efficiency. In this paper, after deeply investigating the mechanism of traditional ACA and analyzing the structure of the solution of TNEP problem, the author proposes a SRPACA for solving TNEP problem. This algorithm can memorize the LOS, partition the solution space through schema recording, strengthen the exploring of local space through a special local search and enhance the searching efficiency through a reasonable Parallelization strategy. The simulation results of two practical sample systems show the feasibility, validity and advancement.

2 The mathematical model of TNEP problem

2.1 Mathematical model

The objective of TNEP problem is to find the most economical planning scheme to meet the load demand in the horizon year subject to the security and reliability constraints [2]. The TNEP problem can be formulated as follows:

\[
\min f(\mathbf{Z}) = \begin{cases} 
\sum_{i=1}^{N} C_{zi} + C_{o}P_{o}(\mathbf{Z}) & \text{when } B = 1 \\
C_{o} & \text{when } B > 1 
\end{cases}
\]

s.t. \quad KCL

\[KVL\]

\[0 \leq g_{j} \leq \bar{g}_{j} \]

\[0 \leq z_{i} \leq \bar{z}_{i} \]

Where, \(f(\mathbf{Z})\) is the total expense. \(\mathbf{Z}\) is a N-dimension solution vector to the optimal planning problem. \(N\) is the number of candidate branches. \(z_{i}\) is the \(i\)-th element of \(\mathbf{Z}\), which represents the number of actually expanded circuits of branch \(i\) and has the value from \{0, 1, ..., \(\bar{z}_{i}\)\}.

\(\bar{z}_{i}\) is the upper bound of \(z_{i}\), i.e. the maximum allowable number of candidate circuits in branch \(i\). \(C_{i}\) is the investment cost of each circuit in the \(i\)-th candidate branch. \(B\) is the number of isolated islands if the network planning scheme identified by \(\mathbf{Z}\) is a disjoint one, otherwise \(B = 1\). \(P_{o}(\mathbf{Z})\) is the total amount of overload in all overloaded circuits. \(C_{o}\) is used to penalize the planning schemes with overloaded circuits, and it should be large enough. \(C_{o}\) is a big constant, which represents the penalty coefficient when the network is disconnected. KCL and KVL represents the Kirchhoff’s current law and Kirchhoff’s voltage law respectively. \(g_{j}\) is the generation of the \(j\)-th bus and \(\bar{g}_{j}\) is its upper bound.

2.2 The schema of the TNEP problem’s solution

In TNEP problem, the decision variables are the number of expanded circuits in each candidate branch, and the following code string can denote the planning scheme.

\[
\mathbf{Z} \quad (z_{1}, z_{2}, ..., z_{N})
\]

A schema is a syntactic pattern that describes parts of solutions and formula (3) shows the format of the planning solution’s schema.

\[
\mathbf{Z'} = \ast, \ldots, z'_{S1}, \ast, \ldots, z'_{S2}, \ast, \ldots, z'_{Sm}, \ast, \ldots, \ast
\]

Where, \(\mathbf{Z'}\) is a schema, “\(\ast\)” is a wildcard which represents either of a number between 0 to \(\bar{z}_{i}\); \(z'_{Sk}\) \(1, 2, \ldots, m\) is a certain number which represents the number of expanded circuits of \(S_{k}\)-th branch in this schema is \(z'_{Sk}\); \(S_{1} \ldots S_{m}\) are called the schema digits.

It can be inferred that a schema actually denotes a local solution space with the elements which have the same value in all the schema digits. Obviously, to do a “global search” in this local space is much easier than in
the whole solution space. What’s more, the recording capability of a schema is much larger than that of a single solution. Hence, it can be deduced that in the searching process, firstly, the solution space can be partitioned into several subspaces in terms of the schemata. Then, in each subspace, a thorough “global search” will be achieved.

3 Traditional ACA and its main disadvantages

3.1 Introduction of traditional ACA

The core idea of ACA is to search the solution space stochastically under the state transition rule, to adjust the direction of local search by the local updating of pheromone intensities according to the objective value of the current solution and to adjust the direction of global search by the global updating of pheromone intensities in the light of the objective value of the current GOS. Generally, there are 6 steps to do, and three of them are the key steps:

1) The definition of the state transition rule

\[ P_j(i,k) = \begin{cases} 1, & \text{if } q \leq q_0, \\ \max_{k \in L_j(i)} \{ \tau_j(i,k) \}, & \text{otherwise} \end{cases} \]  

2) The local updating of the pheromone intensities

\[ P_j(i,k) = \frac{\tau_j(i,k)}{\sum_{k \in L_j(i)} \tau_j(i,k)}, \quad \text{if } k \in L_j(i) \]  

or

\[ P_j(i,k) = 0, \quad \text{otherwise} \]  

Where \( q \) is a random variable uniformly distributed in [0,1], \( q_0 \) is a parameter(0 \( \leq q_0 \leq 1 \)). \( P_j(i,k) \) is the probability with which ant \( j \) in step \( i \) chooses path \( k \). \( \tau_j(i,k) \) is, when ant \( j \) begins to choose a path, the pheromone intensity of path \( k \) in step \( i \). \( L_j(i) \) represents the set of paths that can be chosen by ant \( j \) in step \( i \).

2) The local updating of the pheromone intensities

When finishes a tour, ant \( j \) changes the pheromone intensity of each chosen path by applying the local updating rule of (6):

\[ \tau(i,k) = (1 - \alpha_0) \times \tau(i,k) + \alpha_0 \times \Delta \tau(i,k) \]  

Where, \( 0 < \alpha_0 < 1 \) is the local volatilization coefficient. \( \Delta \tau(i,k) \) is chosen diversely in different cases and usually in inverse proportion to the objective value of the current tour.[4]

3) The global updating of the pheromone intensities

When all ants have finished their tours, the pheromone intensities on all edges are updated according to the global updating rule of (7):

\[ \tau(i,k) = \begin{cases} (1 - \alpha_0) \times \tau(i,k) + \alpha_1 \times \Delta \tau(i,k), & \text{if } k \in Z_{opt} \\ (1 - \alpha_1) \times \tau(i,k), & \text{otherwise} \end{cases} \]  

Where, \( Z_{opt} \) is the GOS, \( 0 < \alpha_1 < 1 \) is the global volatilization coefficient. \( \Delta \tau(i,k) \) is chosen diversely in different cases and usually in inverse proportion to the objective value of the GOS.[4]

3.2 The main disadvantages of traditional ACA

ACA is sometimes inclined to premature convergence in limited computation time. The main reasons cause these problems are listed below:

1) The convergence of ACA is sensitive to the parameters \( q_0 \), \( \alpha_0 \) and \( \alpha_1 \), and the different setting of them may lead to different optimizing results.

2) When a LOS that has satisfying objective value is found very early during the searching process, the later search will be attracted by this solution. This would lead to early stagnation.

3) Since there is no special identification or recording for the LOS, the reduplicate searches will occur.

4) After a period time of searching, the pheromone intensities on some edges may become too high or too low due to the updating of pheromone, which will limit the choosing of edges in certain positions and cause early stagnation.

5) The scale of the colony also affects the performance of ACA a lot. On one hand, if the colony scale is too small, the schemes found by the colony will be too unitary. On the other hand, if the colony scale is too large, it will cause large computational effort and has low efficiency.

4 SRPACA for the TNEP problem

4.1 The process of how an ant finds a scheme
Figure 1 shows how an ant finds a scheme.

The whole searching process is divided into $N$ steps. On the first step, the ant chooses a number from 0 to $Z_i$ as the number of actually expanded circuits of branch $L_i$. Then, the ant orderly chooses the number of actually expanded circuits of other branches. Thus, after the ant finishes choosing the expansion plan of all the branches, the planning scheme $((z_1,z_2,\ldots,z_N))$ of TNEP problem is found.

4.2 The decomposition of the solution space and the confirmation, recording and taboo of the LOS

This is the key step of deciding whether the algorithm can jump away from the LOS or not.

1 The decomposition of the solution space and the confirmation of the LOS.

As analyzed before, after a period time of searching, the pheromone intensities of certain expanded numbers (the values of the expanded circuits’ number) of some branches (the branches on the schemata digits) will be very high, and the algorithm actually gets into the local search in the subspace identified by the schema $*,*,*,\ldots,z_S1\',*\ldots,z_Sm\',*,*,*$. This used to be a disadvantage of the traditional ACA. But, when given enough time, this process can guarantee the sufficient search of this subspace. Functionally, it indicates that ACA can partition the solution space by schema. Further more, this process is completely spontaneous and stochastic so that it can avoid the increased work and possible space loss when decomposing the solution space manually.

2 The schema recording and taboo of the LOS

The following rule is adopted to recognize the schemata digits:

In branch $L_i$, if the pheromone intensity of a certain expanded number $z_k$ satisfies $\frac{τ(i,z_k)}{\sum_{i=0}^{N} τ(i,z_{i})} > \overline{P}$, $i$ is recognized as the schema digit, and $z_k$ is the value of the schema digit. Where, $\overline{P}$ should be large enough. In this paper, $\overline{P}$ is set to 0.95.

When the LOS is confirmed, the values of its schemata digits will be recorded in a Tabu list. For the latter solutions, the algorithm firstly compares the values of the schemata digits to those of the Tabu list. If the latter solutions are already found by the former search, they would be regarded as noneffective ones and their objective values would not be calculated. If they are not existing ones, their objective values will then be calculated. Thus, the taboo from searched solution space against the latter search is realized effectively. What’s more, the schema recording actually memorizes the whole subspace identified by the schema.

3 The combination of schemata in the Tabu list

Supposed there are two schemata:

$Z_1=(*,*,z_{S1}\',*,\ldots,z_{S1}\'.,z_{S2}\',*,\ldots,z_{S2}\'.,\ldots)$

$Z_2=(*,*,z_{S1}\'',*,\ldots,z_{S1}\'',*,\ldots,z_{S2}\'',*,\ldots)$

If $\{S' = \{S_1\',\ldots,S_m\'} \subset S = \{S_1,\ldots,S_m\}$, we define $Z_1 \subset Z_2$ and combine $Z_1$ and $Z_2$ to $Z_2$.

4.3 Parallelization strategy

Because ACA is a multi-agent algorithm, parallel computation is easy to be realized. It can not only improve the computation efficiency, but also solve the premature convergence caused by the improper setting of parameters and colony scales. The parallelization strategies of this paper are shown as follows:

1 Master/slave processors approach - one of the most successful parallelization strategies$[7]$ is used here.

2 The master processor fulfills the functions of recording the current GOS, setting Tabu list, dynamically combing the schemata in the Tabu list and judging whether to stop searching or not.

3 In each slave processor, a colony of artificial ants is set to execute the process of ACA and local search separately, where the parameters of $q_0$, $a_0$ and $a_1$ are set diversely.

5 Simulation results

The sample system is a 77-bus network of East China. The original topology and the detailed data are available from the author.

In this paper, the parallel computation is
implemented by 6 P4 microcomputers interconnected via 100Mbps Ethernet on a MPI platform.

In the latter computation, $D_0$ is set from 8.3 to 8.1, $D_1$ is set from 8.5 to 8.3 and $D$ is set to 400*10$. 

5.1 Simulation results of traditional ACA

1) The influence of different parameter combinations on simulation results

Fig. 2 shows the optimal and average optimal objective values found by ACA under different parameter combinations. Since ACA is a randomized optimization method, it is difficult to find the GOS every time. Here, in each case, the algorithm is tried 50 times.

In this sample system, a preferable combination is $q_0 = 0.5$, $a_0 = 0$, $a_1 = 0.1$. From the two figures, it can be concluded that the combinations of parameters influent the convergence of ACA a lot. The objective value of the GOS is 483.0*10$.

2) The influence of different colony scales on simulation results

Fig. 3 shows the optimal objective values found by ACA at different computation time under variable colony scales. The parameters combination is $q_0=0.5$, $a_0=a_1=0.1$.

From Fig.3, it can be concluded that, when the colony scale is too small, ACA always can’t find the global optimal scheme; while when the colony scale is too large, the computational efforts grow a lot so that although the searching capability of each colony is enhanced, still the computation speed of the whole ACA is decreased. For the 77-bus system, the optimal number of ants in each colony is 30 and the convergence time is about 180 s.

5.2 Simulation results of SRPACA and some comparisons

By using the proposed SRPACA of this paper, the advantages of different parameter combinations are combined and the colony scale is broken into parts so that the computation efficiency is enhanced. The same optimal scheme as that of ACA is found, which is shown in figure 4.
The optimal planning scheme of the 77-bus system

The author also used the coarse-grain parallel genetic algorithm[8] (CPGA) and the SRPACA without local search to solve the TNEP problem, and the comparisons between SRPACA, ACA ($q_0=0.5$, $a_0=0.1$ and $a_1=0.1$) and CPGA are shown in Table 2.

Table 2 The comparisons among different algorithms

<table>
<thead>
<tr>
<th>Target</th>
<th>SRPACA</th>
<th>ACA</th>
<th>CPGA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimal objective value($10^6$)</td>
<td>483.0</td>
<td>483.0</td>
<td>483.0</td>
</tr>
<tr>
<td>Average value of the optimal schemes ($10^6$)</td>
<td>483.0</td>
<td>485.42</td>
<td>484.12</td>
</tr>
<tr>
<td>Probability of finding the globally optimal scheme (%)</td>
<td>100</td>
<td>78</td>
<td>90</td>
</tr>
<tr>
<td>Computation time (s)</td>
<td>50–54</td>
<td>180–188</td>
<td>62–65</td>
</tr>
</tbody>
</table>

From Table 2, it can be concluded that: for this 77-bus system, all the four algorithms can find the optimal scheme with a high probability, but the proposed SRPACA has distinct advantages in both computation speed and global convergence probability. The acceleration ratio $S_p=3.33–3.76$ and the parallel efficiency $E_p=0.56–0.63$.

6 Conclusions

In this paper, on the basis of analyzing the mechanism and main disadvantages of traditional ACA, an improved SRPACA is proposed which is proved to be successful in solving TNEP problems and has the following advantages:
1. It can partition the solution space through schemata, which would disassemble the problem scale without space losing.
2. It can recognize the LOS effectively.
3. It realizes the shield among all the subspaces by schemata recording, which can avoid the redundant search effectually.
4. Through local search, it strengthens the exploring in the neighborhood of the LOS, which enhances the computation speed ulteriorly.
5. It reduces the premature convergence caused by the improper setting of the parameters and the colony scale by parallel computation.

The simulation results of the 77-bus system have shown that the developed SRPACA based system is feasible and efficient, with great potential for practical applications.

Reference:

Biographies:
ZHAI Hai-bao received his B.Sc. degree from Shanghai Jiao Tong University, China in 2000. He is now a doctoral student in Institute of Power System and Automation, Shanghai Jiao Tong University, China. His current research interest is power system planning.