SERBIAN JOURNAL OF ELECTRICAL ENGINEERING
Vol. 3, No. 1, June 2006, 77 - 88

Ant Colony Search Algorithm for Optimal Reactive Power Optimization

K. Lenin, M. R. Mohan[1]

Abstract: The paper presents an (ACSA) Ant colony search Algorithm for Optimal Reactive Power Optimization and voltage control of power systems. ACSA is a new co-operative agents' approach, which is inspired by the observation of the behavior of real ant colonies on the topic of ant trial formation and foraging methods. Hence, in the ACSA a set of co-operative agents called "Ants" co-operates to find good solution for Reactive Power Optimization problem. The ACSA is applied for optimal reactive power optimization is evaluated on standard IEEE, 30, 57, 191 (practical) test bus system. The proposed approach is tested and compared to genetic algorithm (GA), Adaptive Genetic Algorithm (AGA).

Keywords: Reactive power optimization, Ant colony search algorithm, Global optimization.

1 Introduction

The reactive power optimization problem has a significant influence on secure and economic operation of power systems. The reactive power generation, although itself having no production cost, does however affect the overall generation cost by the way of the transmission loss. A procedure, which allocates the reactive power generation so as to minimize the transmission loss, will consequently result on the lowest production cost for which the operation constraints are satisfied. The operation constraints may include reactive power optimization problem. The conventional gradient based optimization algorithm has been widely used to solve this problem for decades. Obviously, this problem is in nature a global optimization problem which may have several local minima and the conventional optimization methods easily lead to local optimum.

On the other hand, in the conventional optimization algorithms, many mathematical assumptions, such as analytic and differential properties of the objective functions and unique minima existing in problem domains, have to be given to simplify the problem. Otherwise it is very difficult to calculate the gradient variables in the conventional methods. Further, in practical power

[1]College of Engineering, Anna University, Chennai, India - 600 025. Email:Lenin_gk@yahoo.com

system operation, the data acquired by the SCADA (Supervisory Control and Data Acquisition) system are contaminated by noise. Such data may cause difficulties in computation of gradients. Consequently, the optimization could not be carried out in many occasions. In the last decade, many new stochastic search methods have been developed for the global optimization problems such as simulated annealing, genetic algorithms and evolutionary programming. Here a new search algorithm ACSA is proposed to find the global solution for reactive power optimization problem.

For the last few years, the algorithms inspired by the observation of natural phenomena to help solving complex combinatorial problems have been increasing interest. In analyzing the behaviors of real ants, it was found that the ants are capable of finding shortest path from food sources to the nest without using visual cues. In the application of this method to our reactive power optimization problem, the initial population of colony can be first randomly generated within the search space of problem. Then, the fitness of ants is individually assessed based on their corresponding objective function. With the selection of trail, the ant dispatch can be activated based on the level of pheromone and distance of the selected trail in order to find the best tour or the shortest path [1].

2 Problem Formulation

The objective of the reactive power optimization problem is to minimize the active power loss in the transmission Network as well as to improve the voltage profile of the system.

Adjusting reactive power controllers like Generator bus voltages, reactive Power of VAR sources and transformer taps performs reactive Power scheduling. We are starting the method by assuming that real power optimization has been already done.

Main objective function for RPO is

$$\text{minimization } P_L = \sum_{i=1}^{NB} P_i(X, Y, \delta) \tag{1}$$

Subject to three constraints:

i) The control vector constraints

$$X_{\min} \le X \le X_{\max} \,. \tag{2}$$

ii) The dependent vector constraints

$$Y_{\min} \le Y \le Y_{\max} \tag{3}$$

and

iii) The power flow constraint

$$F(X, Y, \delta) = 0, \tag{4}$$

where:

$$X = [V_G, T, Q_C]; \tag{5}$$

$$Y = [Q_g, V_L, I]; \tag{6}$$

NB - Number of buses in the system;
δ - Vector of bus phase angles;
P_i - Real Power injection into the i^{th} bus;
V_G - Vector of Generator Voltage Magnitudes;
T - Vector of Tap settings of on load Transformer Tap changer;
Q_C - Vector of reactive Power of switchable VAR sources;
V_L - Vector of load bus Voltage magnitude;
I - Vector of current in the lines;
P_L - Vector of real power flows; and
Q_g - Vector of reactive power generations of the generator buses.

3 ACSA Paradigm

3.1 Behavior of Real Ants

Ant colony search (ACS) studies are inspired from the behavior of real ant colonies that are used to solve function or combinatorial optimization problems. Currently, most work has been done in the direction of applying ACS to combinatorial optimization. The first ACS system was introduced by Marco Dorigo [2], and was called "ant system". Ant colony search algorithms, to some extent; mimic the behavior of real ants. As it is well known, real ants are capable of finding the shortest path from food sources to the nest without using visual cues. They are also capable of adapting to changes in the environment; for example, finding a new shortest path once the old one is no longer feasible due to a new obstacle. The studies by ethnologists reveal that such capabilities are essentially due to what is called "pheromone trails", which ants use to communicate information among individuals regarding path and to decide where to go. Ants deposit a certain amount of pheromone while walking, and each ant probabilistically prefers to follow a direction rich in pheromone rather than a poorer one [3].

The process can he clearly illustrated by Fig. 1a ants are moving on a straight line that connects a food source to their nest. An ant:

- Ants deposit pheromone while walking.

- Probabilistically prefers to follow a direction rich in pheromone.

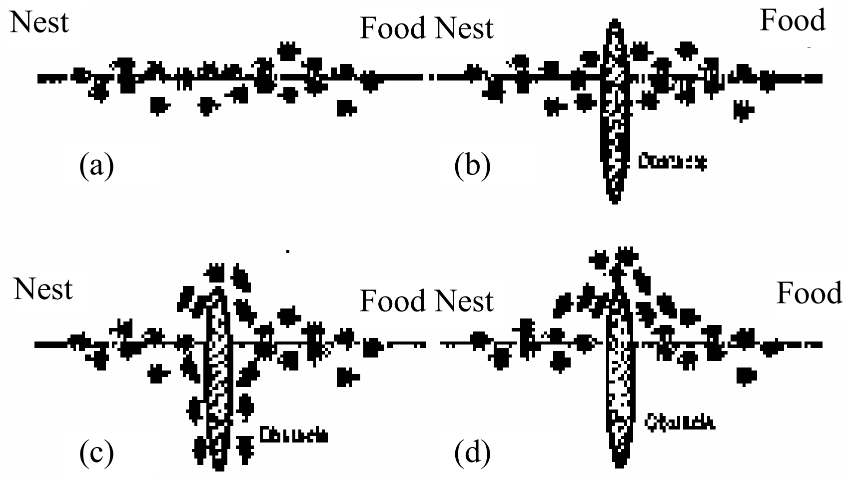

Fig. 1 – Behavior of ants.

(a) Real ants follow a path between nest and food source. (b) An obstacle appears on the path: ants choose whether to turn left or right with equal probability. (c) Pheromone is deposited more quickly on the shorter path. (d) All ants have chosen the shorter path.

This behavior can be explained how ants can find the shortest path that reconnects a line that is broken by an obstacle in Fig. 1b. On introducing, those ants are just in front of the obstacle and they cannot to continue to go. Therefore they have to choose between turning right or left. Half the ants choose to turn right and the other half choose to turn left. A similar situation arises on the other side of the obstacle Fig. 1c. Ants choosing the shorter path will more rapidly reconstitute the interrupted pheromone trail compared with those choosing the longer path. Thus, the shorter path will receive a greater amount of pheromone per time unit and, in turn, a larger number of ants will choose the shorter path. Due to this positive feedback, all the ants will rapidly choose the shorter path Fig. 1d. All ants move at approximately the same speed and deposit a pheromone trail at approximately the same rate. The time to go round the longer side of an obstacle is greater than the shorter. This makes the pheromone trail accumulate more quickly on the shorter side. Ants prefer higher pheromone trail levels causing this accumulation to build up still faster on the shorter path. This behavior of ants can be used to solve optimization problems and in particular the Traveling Salesman Problem (TSP). This is the problem of finding a shortest closed tour, which visits all cities in a given set once. This was the first problem solved by using the ant colony metaphor [4].

3.2 Ant colony search algorithm

3.2.1 ACS state transition rule

In ACS the state transition rule is as follows: an ant positioned on node r chooses the city s to move to by applying the rule given by (7).

$$S = \begin{cases} \text{Arg} \max\limits_{W \in J_{k(r)}} \{[\tau(r,u)][\eta(r,u)]^{\beta}\}, \text{if } q \le q_0, & \text{(exploitation)} \\ S, \quad \text{otherwise} & \text{(biased exploration)} \end{cases} \tag{7}$$

where:

- q is a random number uniformly distributed in $[0\ldots1]$;
- q_0 is a parameter $(0 \le q_0 \le 1)$;and
- S is a random variable selected according to the probability distribution given in equation (8).

The state transition rule used by ant system, called a random-proportional rule, is given by (8), which gives the probability with which ant k in city r chooses to move to the city s.

$$P_k(r,s) = \begin{cases} \dfrac{[\tau(r,s)][\eta(r,s)]^{\beta}}{\sum\limits_{\mu \in J_k(r)} [\tau(r,u)][\eta(r,u)]^{\beta}}, \text{ if } s \in J_k(r) \\ 0, \text{ otherwise} \end{cases} \tag{8}$$

where:

- τ is the pheromone;
- $J_k(r)$ is the set of cities that remain to be visited by ant k positioned on city r (to make the solution feasible);
- β is a parameter, which determines the relative importance of pheromone versus distance $(\beta > 0)$;and
- $\eta = I / \delta$ is the inverse of the distance $\delta(r,s)$.

3.2.2 ACS Global Updating Rule

Global updating is performed after all ants have completed their tours. The pheromone level is updated by applying the global updating rule of (9).

$$\tau(r,s) \leftarrow (1-a)\,\tau(r,s) + \alpha\,\Delta\tau(r,s)\,, \tag{9}$$

where:

$$\Delta\tau(r,s) = \begin{cases} (L_{gh})^{-1}, & \text{if } (r,s) \in \text{global-best-four} \\ 0, & \text{otherwise} \end{cases}$$

and α is the pheromone decay parameter $(0 < \alpha < 1)$.

L_{gh} is the length of the globally best tour from the beginning of the trial.

3.2.3 ACS local updating rule

While building a solution of the UC, ants visit edges and change their pheromone level by applying the local updating rule of (10).

$$\tau(r,s) \leftarrow (1-p)\tau(r,s) + p\Delta\tau(r,s), \tag{10}$$

where:

- p is a heuristically defined coefficient $(0 < p < 1)$;
- $\Delta\tau(r,s) = \tau_0$; and
- τ_0 is the initial pheromone level.

3.2.4 ACS parameter setting

In this program of the following sections the numeric parameters, except when indicated differently, are set to the following values: $\beta = 2$, $q_0 = 0.9$, $\alpha = p = 0.1$ and $\tau_0 = (nL_{nn})^{-1}$, where L_{nn} is the tour length produced by the nearest neighbor heuristic and n is the number of cities [1].

4 Algorithm for Reactive Power Dispatch

It is a combinational optimization problems, at back step the ants make a probabilistic decision according to some discrete probability distribution. Since, our problem is a continuous optimization problem the domain changes from discrete to continuous. The adaptation of moving discrete to continuous by using probability density function.

The function in the normal form is given by

$$g(X,\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(X-\mu)^2}{2\sigma 2}} \tag{11}$$

and the pheromone distribution is based on a mixture of normal kernels. It is defined as a weighted sum of several normal PDFs, and denoted as G:

$$P(X) = G(X,\omega,\mu,\sigma) = \sum_{j=1}^{k} \omega_j g(X,\mu_j,\sigma_j), \tag{12}$$

where:

- ω is the vector of weights associated with the components of the mixture;
- μ is the vector of means; and
- σ is the vector of standard deviations.

The dimensions of all those vectors are equal to the number of normal PDFs constituting the mixture. For convenience we will use parameter k to describe this number dimension of $\omega =$ dimension of $\mu =$ dimension of $\sigma = k$.

Such a distribution allows for reasonably easy generation of random numbers according to it, and yet it provides a much increased flexibility in the possible shape.

4.1 Initialization of ants

In the first step, the colonies of ants are first generated. Ants are positioned on initial state while the initial pheromone value is also given at this step. Fig. 2 plots a multi-stage search space. All the possible permutations constitute this search space. Each stage contains several states, while the order of state selected at each stage can be combined as an achievable tour that is deemed a feasible solution to the problem.

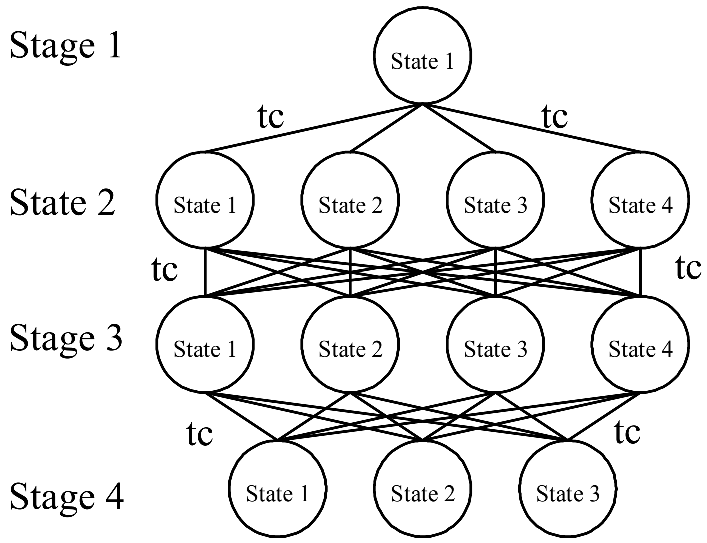

Fig. 2 – *The Multi-state Scarch Space.*

4.2 Fitness evaluation

In the step the fitness of all ants is accessed based on the corresponding objective function.

$$f_n = P_L^n + \alpha \sum_{j=1}^{NG} Q_{G,j}^{\lim,n} + \beta \sum_{j=1}^{NL} \mathrm{V}_{L,j}^{\lim,n}, \quad n = 1,2,\ldots,N_n, \tag{13}$$

where:

- α,β is penalty factors for the constraint violations;
- P_L is total real power loss;

$$Q_{G,j}^{\lim,n} = \begin{cases} Q_{G,\min} - Q_{G,j}^n & \text{if} \quad Q_{G,j}^n < Q_{G,\min} \\ Q_{G,j}^n - Q_{G,\max} & \text{if} \quad Q_{G,j}^n > Q_{G,\max} \end{cases} \tag{14}$$

and

$$\mathrm{V}_{L,j}^{\lim,n} = \begin{cases} |\,\mathrm{V}_{L,j}^n\,| - \mathrm{V}_{L,\max}, & \text{if} \quad |\,\mathrm{V}_{L,j}^n\,| > \mathrm{V}_{L,\max} \\ 0, & \text{otherwise} \end{cases}. \tag{15}$$

The values of penalty factors α and β are chosen such that if there are any constraints violations the fitness function value corresponding to that ant will be ineffective. The best solution is computed for given ants size for the first iteration ($S_{\mathrm{I\,best}}$). The global best solution $S_{\mathrm{G\,best}}$ (best of $S_{\mathrm{I\,best}}$ and previous $S_{\mathrm{G\,best}}$) is initially as taken as first iterations best solution.

The penalty factors are chosen such that if there are any constraints violations the fitness function value corresponding to that ant will be ineffective.

4.3 Pheromone update

Pheromone update is a process of modifying the probability distribution used by the ants during the construction process, so that the ant moves towards the global best solution. At each iteration, the iteration best solution is used for pheromone update for each dimension the update is given by incorporating probability density function and kernel probability density function. This process traditionally consists of two actions:

(i) Positive update: reinforcing the probability of the choices that lead to good solutions and

$$\begin{cases} P^i(X^i) = P^i(X^i) + \sum_{j=1}^{n_S} P^i(X^i) \\ k^i = k^i + n_s \end{cases} \tag{16}$$

where:

- $P^i = (X^i)$ for positive update,
- K^i is kernel probability distributive function.

(ii) Negative update: decreasing probability of other choices i.e. forgetting bad solutions.

$$\begin{cases} P^i(X^i) = P^i(X^i) + \sum_{j=1}^{n_s} P^i(X^i) \\ k^i \;=\; k^i \;+ n_s \end{cases} \tag{17}$$

where:

- $P^i = (X^i)$ for negative update; and
- K^i is kernel probability distributive function.

4.4 Stop criteria

The computation process continues until the number of iterations reaches the predefined maximum threshold, or the iteration counter without improving the best objective function reaches the maximum allowable value. All the tour visited by ants in each iteration should be evaluated. If a better path found in the process, it will be saved for later reference. The best path selected among all iterations implies the optimal scheduling solution to the problem.

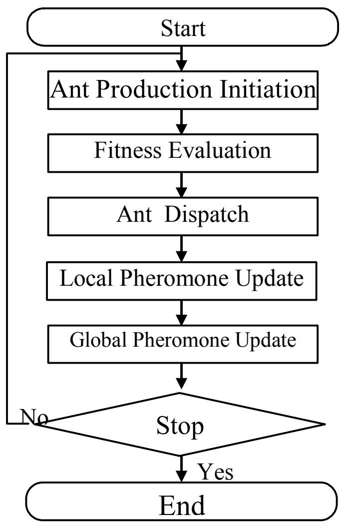

Fig. 3 – *The flowchart of the ACSA unit commitment program.*

5 Numerical Results

The proposed method is tested on standard IEEE 30, 57, 191 (practical) test bus systems and the results shows that ACSA algorithm gives a best solution when compared to adaptive genetic algorithm. **Table 1**, **Table 2** and **Table 3** shows the comparative results and Figs. 4, 5 and 6 shows comparison between the losses and iterations.

For *IEEE* 30 Bus
NG = 6, NL = 41, NB = 30, NTR = 4

Table 1
Comparison of algorithms for 30 bus system.

	GA	AGA	ACSA
No. of iteration	75	65	25
Population size	5	5	5
Time taken (sec.)	20.2	15.7	4.5
Loss	9.6770	9.5680	9.478

For *IEEE* 57 Bus
NG = 7, NB = 57, NTR = 17 NQ = 5

Table 2
Comparison of algorithms for 57 bus system.

	GA	AGA	ACSA
No. of iteration	125	100	76
Population size	10	10	10
Time taken (sec.)	22.7	18.9	4.62
Loss	26.7890	25.0012	24.7752

For *Practical* 191 Bus
NG = 20, NL = 200, NB = 199 NTR = 55

Table 3
Comparison of algorithms for 191 bus system.

	GA	AGA	ACSA
No. of iteration	149	125	102
Population size	20	20	20
Time taken (sec.)	59.7	45.7	6.920
Loss	149.772	149.001	148.241

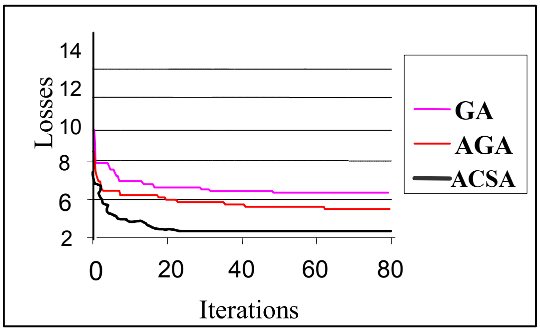

Fig. 4 - *Convergence rate of algorithms for 30 bus system.*

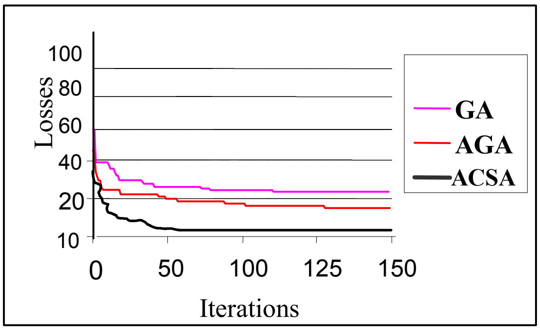

Fig. 5 - *Convergence rate of algorithms for 57 bus system.*

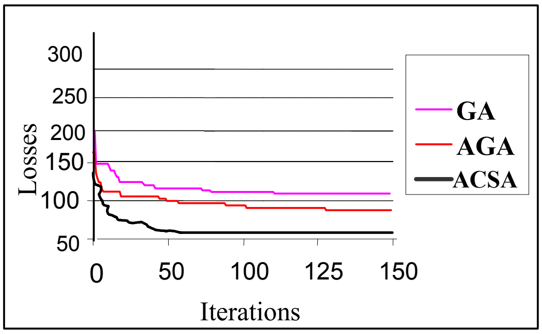

Fig. 6 - *Convergence rate of algorithms for 191 bus system.*
V_{min} 0.95 , V_{max} 1.05, T_{min} 0.9 , T_{max} 1.1, sus_{max} 0.15, sus_{min} 0.0.

6 Nomenclature

NB - total number of buses

NL - total number of load buses

NG - total number of generator buses

NTR - total number of transformers

V_G - generator voltage

Q_s - vector of switchable VAR sources

T - vector of tap settings of on-load tap changing (OLTC) of transformers

Q_g - vector of reactive power generations of the generator buses

V_L - vector of load bus voltages

N_Q - number of capacitances.

7 Conclusion

In this paper ACSA algorithm has been developed for determination of global optimum solution for reactive power optimization problem. The performance of the proposed algorithm demonstrated through its evaluation on IEEE 30, 57, 191 (practical 191) bus power system shows that ACSA is able to undertake global search with a fast converges rate and a future of robust computation. From the simulation study it has been found that ACSA converges to the global optimum.

8 References

[1] M. Dorigo, L.M. Gambardella: Ant colony system: a cooperative learning approach to the traveling salesman problem, IEEE Trans. Evolutionary Computation, Vol. 1, pp. 53-66, Apr. 1997.

[2] T. Sum-im, W. Ongsakul: Ant Colony Search Algorithm for Unit Commitment, IEEE, 2003.

[3] Y.H. Song,Irving, M.R.: Optimisation Techniques for Electrical Power Systems: Part 2 Heuristic Optimisation Methods, IEE Power Engineering Journal, pp. 151-160, June 2001.

[4] Q.H. Wu, J. T. Ma: Genetic Search for optimal reactive power dispatch of power systems, Control '94, Proceedings of the IEE International Conference on, Vol. 1 pp.717–722, 1994.

[5] Q.H. Wu, Y J Cao, Y.J. Wen: Optimal Reactive power dispatch using an genetic algorithm, International Journal of Electrical Power and Energy Systems, Vol. 20 pp. 564–569, 1998.