Application of Ant Colony Optimization Algorithms to Optimal Reactive Power Dispatch

ALIREZA ABBASY
Ghods Niroo Consulting Engineers
Motahari Avenue, Tehran
IRAN

Abstract: - The optimal reactive power dispatch (ORPD) problem is formulated as a combinatorial optimization problem involving nonlinear objective function with multiple local minima. In this paper, as a new approach, different ant colony optimization (ACO) algorithms are applied to the reactive power dispatch problem. Ant system (AS), the firstly introduced ant colony optimization algorithm, and its direct successors, elitist ant system (EAS), and max-min ant system (MMAS), are employed to solve the reactive power dispatch problem. To analyze the efficiency and effectiveness of these modern search algorithms, the proposed methods are applied to the IEEE 30-bus system and the results are compared to those of conventional mathematical methods, genetic algorithm, evolutionary programming, and particle swarm optimization.

Key-Words: - Ant colony optimization, Ant system, Elitist ant system, Max-min ant system, Optimal reactive power dispatch

1 Introduction

Optimal reactive power dispatch (ORPD) plays a decisive role in economic and secure operation of power systems. ORPD is a special case of the optimal power flow (OPF) problem in which control parameters are those variables, which have a close relationship with the reactive power flow, such as generator bus voltages, output of static reactive power compensators, transformer tap-settings, shunt capacitors/ reactors, etc. The objective of ORPD is to minimize the network real power loss and improve the voltage profile, while satisfying a given set of operating and physical constraints. Because outputs of shunt capacitors/reactors and tap-settings of transformers are discrete variables while any other parameter in ORPD is continuous, the reactive power dispatch problem can be modeled as a large-scale mixed integer nonlinear programming (MINLP) problem.

Several classical approaches such as linear programming, nonlinear programming, quadratic programming, the mixed integer programming, the Newton method, etc., have been successfully applied to solve the ORPD problem [1]–[3]. Recently, methods based on interior point techniques, which present much faster convergence and noticeable convenience in handling inequality constraints in comparison with other methods, have been presenting encouraging results to handle the large-scale ORPD/OPF problems [4]–[5]. However, these techniques have severe restrictions in handling the integer problems and objective functions having multiple local minima.

In recent years, many new stochastic search methods have been developed for the global optimization problems. Many salient stochastic methods such as genetic algorithm (GA), simulated annealing (SA), tabu search (TS), evolutionary programming (EP), evolutionary strategy (ES), and particle swarm optimization (PSO), have been developed to solve the ORPD problem [6]–[10]. Such methods offer considerable superiority in finding the global optimum point and in handling discontinuous and non-convex objective functions. But many of these methods suffer from the inability to manage optimization problems of integer and discrete nature.

Ant colony optimization (ACO) algorithms are modern search methods, which are inspired by the foraging behavior of ant colonies, and target discrete optimization problems. The ACO methods belong to biologically inspired heuristic (meta-heuristics) methods. Ants are members of a family of social insects, which live in organized colonies. Real ants are blind creatures, which can stochastically build their path from their nest to the food source, without using visual cues, but by using a combination of heuristic information and a chemical secretion called pheromone. The first ACO algorithm was developed by Dorigo as his PhD thesis in 1992 [11], and published under the name ant system (AS) in [12]. AS obtained encouraging initial results, but found to have some drawbacks compared to other state-of-the-art meta-heuristic search methods [13]. So some improved extensions of the AS algorithm were developed to enhance the performance of the original version. These extensions include elitist ant system (EAS), and max-min ant system (MMAS). In power systems, the ACO has been applied to solve the optimum generation scheduling problems, and the optimum switch relocation problem, recently.

In this paper, we propose a novel ACO-based optimal reactive power dispatch approach and three different ACO algorithms, including the simple ant system and two of its direct successors, elitist AS, and max-min AS, have been used to solve the ORPD problem. Various aspects of performance of these modern search algorithms in solving the ORPD problem are analyzed using the IEEE 30-bus test system.

2 Problem Formulation

The objective of the reactive power dispatch consists of minimising the real power loss in the transmission network, which can be described mathematically as follows:

$$f = \sum_{k \in N_l} P_{loss,k} = \sum_{k \in N_l} g_k \, (V_i^2 + V_j^2 - 2V_i V_j \cos\theta_{ij}) \quad (1)$$

where $k = (i, j)$; $i \in N_B$ and j belongs to the set of buses adjacent to bus i. In the above formulation, N_l and N_B represent the set of network branches and the set of total buses, respectively, g_k is the conductance of branch k, V_i is the voltage magnitude of bus i ,and θ_{ij} is the voltage angle difference between buses i and j.

The minimization of the above function is subject to a number of equality and inequality constraints:

$$0 = P_{gi} - P_{di} - V_i \sum_{j \in N_0} V_j (G_{ij} \cos\theta_{ij} + B_{ij} \sin\theta_{ij}), i \in N_0 \quad (2)$$

$$0 = Q_{gi} - Q_{di} - V_i \sum_{j \in N_0} V_j (G_{ij} \sin\theta_{ij} + B_{ij} \cos\theta_{ij}), i \in N_{PQ} \quad (3)$$

where N_0, N_{PQ} are the set of total buses excluding slack bus, the set of PQ buses, respectively; P_{di} and Q_{di} are the specified active and reactive power demand at bus i, respectively; V_i is the voltage magnitude of bus i; G_{ij} and B_{ij} are transfer conductance and susceptance between buses i and j.

The inequality constraints of this optimization problem are the lower and upper bounds of bus voltages, transformer tap positions, reactive power output of generators, and reactive power output of the shunt VAR sources.

The decision variables of the ORPD problem are the voltages at generation buses, reactive power output of shunt compensators and transformer tap positions.

3 Basic Concepts of Ant Colony Optimization Algorithms

ACO algorithms simulate the behavior of real ants [12], and are based on the principle that a group (colony) of ants is able to find the shortest path between two points, without any sense of sight but by using simple communication mechanisms among the members of the colony. While walking, each ant deposits a chemical substance called pheromone on the ground. The pheromone guides other ants toward the target point. Each ant uses a probabilistic choice rule, based on the quantity of pheromone and some heuristic information to choose its path.

First of all, to devise any ACO algorithm to be used in solving any optimization problem, a graph demonstrating the whole search space is constructed. This graph can be shown as: $G(C, L)$, where C represents the set of all nodes and L is the set of all arcs of the graph. Each ant begins its trip from a specified start node (nest) and moves toward the target node (destination), crossing some nodes of the search graph.

After the graph construction, all ants are positioned on the start node and initial values for pheromone trails are set on all arcs of the graph. Then each ant starts its tour from the starting node to the target node and chooses the next node to visit, taking into account the amount of pheromone trail and some heuristic information. Ants prefer to move to nodes, which have a high amount of pheromone. This process will repeat until all ants complete a tour.

After all ants have completed their tours, the fitness value of each ant must be calculated. Some fitness functions of the optimization problem can be used to evaluate the performance of the ants. After the fitness evaluation of all ants, these fitness values are used to update the amount of pheromone on the arcs of the search graph. Update of pheromones in each iteration of the algorithm is performed in two stages. First, pheromone evaporation is performed on all arcs of the graph. The pheromone amount of each arc will evaporate over time, and it loses intensity if other ants lay down no more pheromone. After the evaporation, a specific amount of pheromone is added on some arcs. This pheromone update process differs for different ACO algorithms and will be described for each algorithm separately in the next section. The algorithm is terminated if the number of iterations reaches the predefined maximum value or the best solution of each iteration has not improved for a specific number of iterations.

4 Ant System and its Extensions

As mentioned before, the first ACO algorithm, ant system (AS) was introduced using the traveling salesman problem (TSP) as a sample application. AS presented interesting initial results, but had some deficiency compared to powerful algorithms available for the TSP. The main difference between AS and its extensions is the way of pheromone update, as well as some additional details in the management of the pheromone trails. This section is devoted to presentation of AS, EAS and MMAS algorithms, and analyzing their characteristics.

4.1 Ant System

After the graph construction, a constant amount of pheromone is put on all the arcs. In AS an appropriate value for the initial pheromone is a value slightly higher than the expected amount of pheromone deposited by the ants in one iteration. In AS, *m* ants build their tour from the start point to the target. At each construction step, each ant applies a probabilistic choice rule to decide which point to visit next. The probability with which ant *k*, currently at node *r*, chooses to go to node *s* is as follows:

$$p_{rs}^{k} = \frac{[\tau(r,s)]^{\alpha}[\eta(r,s)]^{\beta}}{\sum_{l}[\tau(r,l)]^{\alpha}[\eta(r,l)]^{\beta}}, \quad s,l \in N_r^k \qquad (4)$$

where $\eta(r,s)$ is a heuristic value that is available a priori, α, β are two parameters which determine the relative influence of the pheromone trail and the heuristic information, and N_r^k is the feasible neighborhood of ant *k* when being at node *r*. By this probabilistic rule, the probability of choosing a particular arc (*r,s*) increases with the value of the associated pheromone trail $\tau(r,s)$ and the heuristic information value $\eta(r,s)$.

After all ants have constructed their tours, the pheromone trails are updated. This is done by first lowering the pheromone value on all arcs by a constant factor, and then adding pheromone on the arcs the ants have crossed in their tours. Pheromone evaporation is implemented by:

$$\tau(r,s) = (1-\rho)\,\tau(r,s)\ , \quad \forall (r,s) \in L \qquad (5)$$

where $0 < \rho \le 1$ is the pheromone evaporation rate. This parameter is used to avoid unlimited accumulation of the pheromone trails and it enables the algorithm to forget bad decisions made before.

After evaporation, all ants deposit pheromone on the arcs they have crossed in their tour:

$$\tau(r,s) = \tau(r,s) + \sum_{k=1}^{m} \Delta\tau^k(r,s)\ , \quad \forall (r,s) \in L \qquad (6)$$

where $\Delta\tau^k(r,s)$ is the amount of pheromone deposited by ant k on the arc (*r,s*) it has crossed. It can be shown as follows:

$$\Delta\tau^k(r,s) = \begin{cases} 1/f^k\ , & if\ (r,s) \in T^k \\ 0\,, & otherwise \end{cases} \qquad (7)$$

where f^k is the objective function value corresponding to the tour T^k built by ant *k*. According to (12) the better an ant's tour is, the more pheromone the arcs belonging to this tour receive. Generally, arcs that are used by many ants which are part of short tours (tours leading to small values of the objective function) receive more pheromone and are more likely to be chosen by ants in future iterations of the algorithm.

4.2 Elitist Ant System

One of the first improvements performed on the initial AS is the elitiset ant system (EAS) [11]. The idea is to provide strong additional pheromone reinforcement to the arcs belonging to the best tour found since the start of the algorithm. This additional reinforcement to the best-so-far tour, T^{bs}, can be viewed as additional pheromone deposited by an additional ant called 'best-so-far' ant. The additional reinforcement of tour T^{bs} is achieved by adding a quantity e/f^{bs} to its arcs, where *e* is a parameter that defines the weight given to the best-so-far tour, T^{bs}, and f^{bs} is the value of objective function corresponding to T^{bs}. Therefore the pheromone update is done according to:

$$\tau(r,s) = \tau(r,s) + \sum_{k=1}^{m} \Delta\tau^k(r,s) + e\Delta\tau^{bs}(r,s)\,,\ \forall (r,s) \in L \qquad (8)$$

where $\Delta\tau^k(r,s) = 1/f^k$ and $\Delta\tau^{bs}(r,s) = 1/f^{bs}$.

Note that in EAS, pheromone evaporation is implemented as in the initial AS.

4.3 Max-Min Ant System

Max-min ant system [13] introduces four main modifications with respect to AS. First, it strongly exploits the best tours found during the algorithms. In MMAS, either the iteration-best ant, that is, the ant that produced the best tour in the current iteration, or the best-so-far ant, that is, the ant that obtained the best tour since the start of the algorithm, is allowed to deposit pheromone. But such a strategy may lead to a stagnation situation in which all the ants follow the same tour, because of the excessive accumulation of pheromone trails on arcs of a good, although sub-optimal, tour. To counteract this effect, a second modification introduced by MMAS is that it limits the possible range of pheromone trail values to the interval $[\tau^{min}, \tau^{max}]$. Third, the pheromone trails are initialized to the upper trail limit, which, together with a small pheromone evaporation rate, increases the exploration of the search space at the start of the search. Finally, in MMAS, pheromone trails are reinitialized each time the system approaches stagnation or when no improved tour has been generated for a certain number of iterations.

5 Implementation of ACO Algorithms for the ORPD Problem

The vector of decision variables for the ORPD problem consists of all the generator bus voltages, transformer tap ratios, and the reactive power output of shunt capacitors/reactors. In our approach, the search space of the problem that represents the settings of all these control parameters is mapped on a search graph, which is the space in which the artificial ants walk. Fig. 1 shows such a graph for the ORPD problem. Each stage of the graph corresponds to a specific control variable and all the possible discrete settings for that variable are represented by the states of the corresponding stage. The number of stages is equal to the number of control parameters of the optimization problem, and the number of states of a specific stage is equal to the number of possible discrete values for the corresponding variable. So the number of states are not essentially the same in all stages.

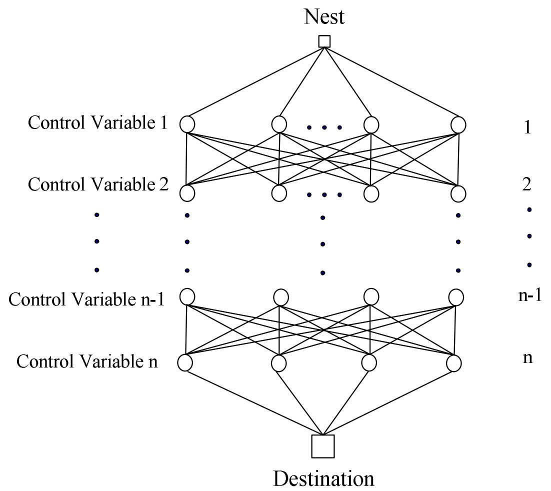

Fig. 1. Search graph for the ORPD problem

Fig. 2 shows the general algorithm of the ACO algorithms for solving the optimal reactive power dispatch problem.

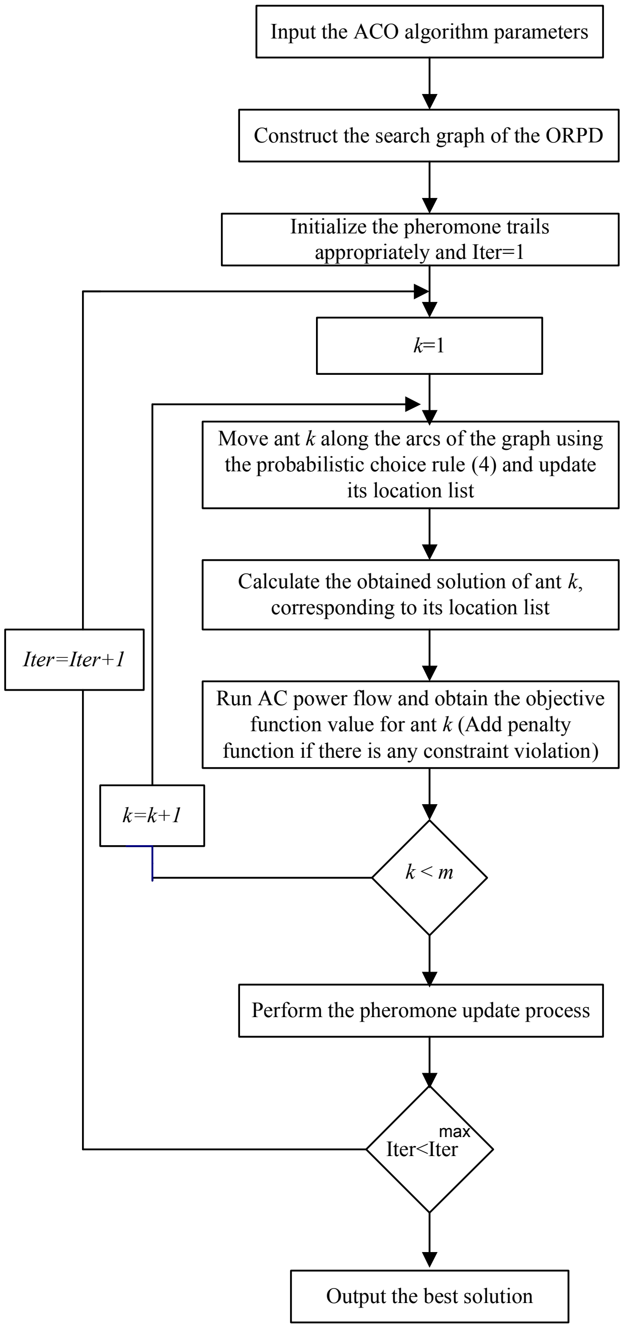

Fig.2. The general flowchart of ACO algorithms for the ORPD problem

Each ant chooses the next states to go to in accordance with the probabilistic choice rule given in (4). When ant k moves from one stage to the next, the state of each stage will be recorded in its location list, J^k. After its tour is complete, its location list is used to compute the ant's current solution. When all ants in the colony complete their path, and the solution of each ant is achieved, the fitness of each ant is computed. This is done by performing an AC load flow on the system using the control variables' values obtained by each ant. As we mentioned before, the main difference between AS and its extensions, proposed in this paper to solve the ORPD problem, is the way of pheromone update. So the following algorithm is the same for all three ACO-based algorithms introduced in the previous sections, except for the pheromone update process, that was exactly described for each algorithm before.

The important issue is the way of handling constraints of the ORPD problem. All ants are free to choose infeasible paths and constraint violations are penalized and added to the objective function as penalty terms. In the ORPD problem, generator bus voltages, tap position of transformers, and the amount of the reactive power source installations are control variables, which are self-constrained. Voltages of PQ-buses and injected reactive power of PV-buses are constrained by adding them as quadratic penalty terms to the objective function.

6 Numerical Results

The IEEE 30-bus system [14] is used as the test system to apply the presented ACO algorithms to the optimal reactive power dispatch problem. The reactive power source installation buses are 3, 10 and 24. The transformer taps are in 21 steps (0.01 p.u. for each step), while the reactive power compensations are in 13 steps (0.03 p.u. for each step). We have selected 41 steps for generator-bus voltages (0.005 p.u. for each step) in this study. The graph representing the whole search space of the problem consists of 13 stages (four transformer taps, six generator voltages, and three shunt capacitors), and the total number of states is equal to 369.

To analyze different characteristics of the presented ACO algorithms in solving the ORPD problem, simulation results have been compared with various techniques available in the literature, namely, Broyden's nonlinear programming method, standard genetic algorithm (SGA), adaptive genetic algorithm (AGA), and particle swarm optimization (PSO), all in [10], and the EP method in [8].

Table 1 summarizes the minimum active power loss obtained by different methods. As can be seen, the EAS algorithm has achieved acceptable results and outperforms the EP, Broyden, and SGA methods. The AS algorithm has shown poor performance and its optimal solution is worse than that of all stochastic search methods in the table. One can see from Table 1 that the optimal dispatch solutions determined by the MMAS algorithm lead to lower power losses than found by other methods, which confirms that MMAS is well capable of determining the global or near global optimum dispatch solution. It should be noted that all the presented ACO algorithms have succeeded in keeping all the dependant variables, load-bus voltages and reactive power output of generators, within the their specified limits.

Table 1. Comparison of optimal result obtained by different methods

Method	Minimum P_{loss} (p.u.)
Broyden	0.055860
SGA	0.049800
AGA	0.049260
EP	0.049630
PSO	0.049262
AS	0.049945
EAS	0.049298
MMAS	0.049035

Because of the randomness of all stochastic search algorithms, AS, EAS, MMAS, and SGA are executed 50 times when applied to the test system. The best, worst and average solutions obtained by these algorithms are presented in Table 2. From this table, the MMAS method shows noticeable consistency by keeping the difference between the best and worst solutions within 0.8%. The outstanding issue in the results presented in Table 2 is that the average solution obtained by the MMAS method is better than the best solutions obtained by any other method in Table 1.

To analyze the results obtained by the presented ACO algorithms in a statistical manner, the relative frequency of convergence is provided for each range of power losses among 50 trials in Table 3. The numbers presented in Table 3 show the percentage of solutions found in each specified range by each method. One can observe the robustness and superiority of the MMAS method, which has achieved 100% of its solutions in such a range, which SGA and basic AS have not been able to reach even once.

Table 2. Comparison of performance of different methods within 50 trials (P.U.)

Compared item	AS	EAS	MMAS	SGA
Best solution	0.049945	0.049298	0.049035	0.049800
Average solution	0.050877	0.049527	0.049166	0.050810
Worst solution	0.051843	0.050316	0.049428	0.052140

Table 3. Comparison of relative frequency of convergence for different methods

Methods	Range of power loss (p.u.)						
	0.0520 - 0.0525	0.0515 - 0.0520	0.0510 - 0.0515	0.0505 - 0.0510	0.0500 - 0.0505	0.0495 - 0.0500	0.0490 - 0.0495
SGA	4	10	22	30	26	8	0
AS	0	12	22	32	28	6	0
EAS	0	0	0	0	18	38	44
MMAS	0	0	0	0	0	0	100

7 Conclusion

In this paper, an ACO-based approach for the optimal reactive power dispatch problem was proposed and three different ACO algorithms, including AS, EAS and MMAS, were applied to the problem. Our approach consists of mapping the solution space on a search graph, where artificial ants walk. According to the results presented, the AS method has severe limitations in finding global or near global optimum solutions compared to other powerful stochastic techniques. According to the computational results, applying the elitist strategy to the basic AS improves the algorithm's performance in every respect. The EAS algorithm is well capable of finding acceptably high-quality solutions, and outperforms some stochastic methods in the literature. The MMAS algorithm, on the other hand, due of its special emphasis on the best solutions found during the search process, together with its ability to explore the search space in early iterations and also its preventive strategy against stagnation, is quite effective and robust in solving the optimal reactive power dispatch problem as a complex optimization problem of integer and discrete nature.

References:

[1] N. Grudinin, Reactive power optimization using successive quadratic programming method, *IEEE Transactions on Power Sysems,* vol. 13, pp. 1219-1225, Nov. 1998

[2] K. Aoki, M. Fan, and A. Nishikori, Optimal VAR planning by approximation method for recursive mixed integer linear programming, *IEEE Transactions on Power Systems*, vol. 3, pp. 1741-1747, Nov.

[3] D. I. Sun, B. Ashley, B. Brewer, A. Hughes, and W. F. Tinney, Optimal power flow by Newton approach, *IEEE Transactions on Power Systems*, vol. PAS-103, pp. 2864-2880, Oct. 1984.

[4] S. Granville, Optimal reactive dispatch through interior point methods, *IEEE Transactions on Power Sysems*, vol. 9, pp. 136-146, Feb. 1994.

[5] J. A. Momoh, S. X. Guo, E. C. Ogbuobiri, and R. Adapa, The quadratic interior point method solving power system optimization problems, *IEEE Transactions on Power Systems*, vol. 9, pp.1327-1336, Aug. 1994.

[6] K. Iba, Reactive power optimization by genetic algorithm, *IEEE Transactions on Power Systems*, vol. 9, pp. 685-692, May 1992.

[7] Y. Liu, L. Ma, and Zhang, GA/SA/TS hybrid algorithms for reactive power optimization, in *Proceedings of IEEE Power Engineering Society Summer Meeting*, vol. 1, pp. 245-249, July 2000.

[8] L. L. Lai, J. T. Ma, Application of evolutionary programming to reactive power planning—Comparison with nonlinear programming approach, *IEEE Transactions on Power Systems*, vol. 12, pp. 198-206, Feb.

[9] K. Y. Lee and F. F. Yang, Optimal reactive power planning using evolutionary algorithms: A comparative study for evolutionary programming, evolutionary strategy, genetic algorithm and linear programming, *IEEE Transactions on Power Systems*, vol. 13, pp. 101-108, Feb. 1998.

[10] B. Zhao, C. X. Guo, and Y. J. Cao, A multiagent-based particle swarm optimization approach for optimal reactive power dispatch, *IEEE Transactions on Power Systems*, vol. 20, pp. 1070-1078, May 2005.

[11] M. Dorigo, *Optimization, learning and natural algorithms*, Ph.D. dissertation, Politecnico de Milano, Milan, Italy, 1992.

[12] M. Dorigo, V. Maniezzo, and A. Colorni, Ant system: Optimization by a colony of cooperating agents, *IEEE Transactions on Systems, Man, Cybernetics B*, vol. 26, pp. 29-41, Feb. 1996.

[13] M. Dorigo, T. Stutzle, *Ant Colony Optimization*, New Delhi: Prentice-Hall of India, 2005.

[14] K. Y. Lee, Y. M. Park, and J. L. Ortiz, A united approach to optimal real and reactive power dispatch, *IEEE Transactions on Power Systems*, vol. 1, pp. 1147-1153, May 1985.