Turk J Elec Eng & Comp Sci, Vol.19, No.6, 2011, © TÜBİTAK
doi:10.3906/elk-1004-2

Using learning automata for multi-objective generation dispatch considering cost, voltage stability and power losses

Arif KARAKAŞ[1,*], **Celal KOCATEPE**[1], **Fangxing LI**[2]
[1] *Department of Electrical Engineering, Yıldız Technical University, İstanbul, 34349, TURKEY*
e-mails: akarakas@yildiz.edu.tr, kocatepe@yildiz.edu.tr
[2] *Department of Electrical Engineering and Computer Science, University of Tennessee, Knoxville, TN 37996, USA*
e-mail: fli6@utk.edu

Received: 01.04.2010

Abstract

The economical and secure operation of power systems has significant importance. Due to technical limitations, the best economical operation point is not always the desired operating point for system stability or power losses. In this study, first, the most economical operating point is obtained by solving the non-linear, network-constrained economic dispatch problem using a genetic algorithm. Then, the system voltage stability is analyzed to compare the different possible operating points using V-Q sensitivity analysis. The power losses, obtained for various operating points, are considered the third objective function. Finally, these 3 aspects of cost, voltage stability, and power losses are combined, using the learning automata technique, to achieve a multi-objective optimization solution. The methodology was implemented in MATLAB 7.8 and applied to the IEEE 30-bus test system. The same technique of learning automata may be applied in the future to similar problems that need multi-objective consideration.

Key Words: *Economic dispatch, voltage stability, genetic algorithm, learning automata*

1. Introduction

The common formulation of economic dispatch (ED) is to find the optimal generation cost, subject to a number of equality and inequality constraints. For instance, ED needs to consider the generator's minimum and maximum capacity limits. Also, the transmission network constraints and losses must be considered in order to perform the economic dispatch accurately because power plants are generally not located near load centers. The consideration of these constraints, as well as the quadratic cost function of generators, typically makes ED a non-linear programming problem.

*Corresponding author: Department of Electrical Engineering, Yıldız Technical University, İstanbul, 34349, TURKEY

Various mathematical programming methods and optimization techniques have been applied to solve ED. The classical solutions to ED are surveyed in [1-2]. Several modern heuristic search techniques, including genetic algorithms, simulated annealing, tabu search, and particle swarm have been reported in [3]. An approach based on a genetic algorithm is developed in [4] to show its efficiency in solving highly non-linear economic dispatch problems. A genetic algorithm (GA) is a heuristic searching algorithm based on natural genetics and natural selection [5]. GA is inspired by the study of genetics [6]. In each new generation, a new set of chromosomes is produced using information from the old generations. GA is not a simple random process. It uses the historical information of old generations efficiently to consider new search points and improve performance [5, 7]. GA has several advantages when compared with classical search methods. One advantage is the use of stochastic operators instead of deterministic rules in the search for an ED solution. GA searches arbitrarily, from point to point, to avoid being trapped in the local optimal points that other methods may take.

On the other hand, voltage stability has also been recognized as an important problem of power system operation. An analysis of voltage stability can be achieved using a variety of techniques. These techniques can provide an indicator of voltage instability or collapse with an index or a margin. Voltage collapse is a severe form of voltage instability in any region of a system, where the system cannot maintain a sustainably acceptable voltage level. Additionally, in power system operation, voltage stability is strongly associated with power system losses. When the system is subjected to faults and becomes stressed, the line losses begin to grow rapidly.

Although ED usually addresses network issues such as thermal limits or contingency constraints, it is not a typical practice to explicitly consider the voltage stability. In systems with a severe threat of voltage stability, it is highly desirable to consider a trade-off between the economic dispatch solution and the stability criterion using techniques like the multi-objective optimization method.

The learning automata (LA) method is one such technique, which is derived from the theories of probability and Markov processes. In [8], M. L. Tsetlin began the study of learning automata. In his work, he presents a deterministic automaton running in random environments as a model of learning. In [9], Narendra and Thathachar have studied the theory and applications of learning automata, as well as learning algorithms and the behavior and hierarchical structure of the learning automata, and have performed simulation studies.

Learning is defined as the ability to improve the behavior of a system as a result of its past experience. The aim of a learning system is the optimization of a function not clearly known [10]. The stochastic automaton seeks a solution to the problem without any information on the optimal action (initially, all actions are assumed to have an equal probability). An initial action can be chosen randomly. Then, the response (output) from the system is obtained, and the action probabilities are updated based on that response. This process will be repeated until the optimal action is obtained. An automaton that behaves as described in order to improve its performance is called a learning automaton. A learning automaton is a solution that can be applied to complete the learning process.

Certainly, learning automata can be used to obtain an optimal point considering multiple objectives such as generation production cost, voltage stability, and power losses, which can be concerns in a power system dispatch problem. The generation production cost, the V-Q sensitivity index for stable operation, and the power system losses are considered the performance values for the multi-objective optimal operation of the power system, which can be used by learning automata.

In this research, first, a practical and efficient genetic algorithm is presented for solving the network-constrained economic dispatch problem for a power system. Then, the V-Q sensitivity index is used to compare different operating points of the system. Finally, an algorithm based on learning automata is presented to obtain an optimal operation point that satisfies the production cost, voltage stability, and power loss criteria. An hourly static analysis has been completed in this study. The methodology was implemented in MATLAB 7.8 and applied to the IEEE 30-bus test system.

This paper is organized as follows. Section II provides a mathematical formulation for the network-constrained economic dispatch. Section III presents a genetic algorithm solution for economic dispatch. Section IV presents the V-Q sensitivity index. Section V proposes a learning automata algorithm for multi-objective generation dispatch. Section VI discusses the methodology of this study. The tests and results are shown in Section VII. Finally, the concluding remarks are presented in Section VIII.

2. Network-constrained economic dispatch for production cost minimization

The economic dispatch of a power system can be expressed as an optimization problem, minimizing the fuel cost of the generator units, under a number of constraints. Economic dispatch is a subproblem in generator scheduling and generally solved for 1 h, 12 h, 24 h or 48 h considering the constraints. Mathematically, this is given by

$$\min F = \sum_{i=1}^{n} F_i(P_i) \tag{1}$$

$$F_i(P_i) = a_i + b_i \cdot P_i + c_i \cdot P_i^2 \tag{2}$$

where a_i, b_i, andc_i are the cost coefficients of the i^{th} generation unit, n is the number of generation units connected to the power system, P_i is the active power generated from the i^{th} unit in MW, and $F_i(P_i)$ is the operating cost of the i^{th} unit in \$/h.

This optimization function is subjected to the following constraints;

The power balance constraint is given by

$$\sum_{i=1}^{n} P_i - P_D - P_L = 0 \tag{3}$$

where P_D is the active power loss of the transmission system and P_L is the load demand of the power system.

The generator's unit capacity limit is given by

$$P_i^{\min} \le P_i \le P_i^{\max}, \quad i = 1, 2, \ldots, n \tag{4}$$

where $P_i^{\min}$ and $P_i^{\max}$ are the lower and upper limits of the i^{th} unit, respectively.

The transmission line constraint is

$$P_{mn} \le P_{mn}^{\max} \tag{5}$$

where P_{mn} is the real power flow in the line 'm-n' (i.e., from bus m to bus n) and $P_{mn}^{\max}$ is the maximum power flow limit in the line 'm-n'.

3. Genetic algorithm to solve the economic dispatch for cost minimization

The primary purpose of GAs is to duplicate the evolutionary process of nature in a mathematical way. The methodology is based on the evaluation of a set of the population (solution of the objective function). The population is created randomly by individuals. The individuals can be encoded by either binary or real numbers. We use real numbers to encode the individuals in this research. Each individual is a vector of variables as real numbers. The fitness of an individual is obtained by the value of the objective function.

A typical GA performs several simple steps [5]:

i. Create an initial "population" as a generation;

ii. Evaluate the individuals in this generation using a fitness function;

iii. Select the best individuals and create a new generation.

The goal is to search for a minimum cost value in the economic dispatch problem with nonlinear constraints. Therefore, we begin the process by defining a chromosome as an array of variables to be optimized in the GA. If the chromosome has n variables given by $P_1, P_2, \ldots, P_n$, then the i^{th} chromosome, in real values, is written as

$$\text{chromosome}_i = [P_{1i}, P_{2i}, ..., P_{ni}], i = 1, 2, \ldots, N_p \tag{6}$$

where P_{ni} is the active power generated from the n^{th} unit at i^{th} chromosome.

The iteration of the GA process depends on a set of solutions (chromosomes), called the population. To initialize the GA process, generally we need to define an initial population, which consists of randomly generated N_p chromosomes.

The initial population for solving the ED is generated with satisfaction of equation (4). The transmission loss and line flows are then obtained by the load flow solution using the Newton-Raphson method. All constraints in (3-5) are rechecked to verify whether they have been violated or not. If the solution has at least one violated constraint, then it is subject to elimination using a penalty value.

The population size shows the number of individuals (chromosomes) in the population. If the population size is low, then the GA will explore a small solution space. On the other hand, if the population size is high, the GA will be time-consuming. The population size can be different for various problems and must be chosen using experience and expertise.

At this moment, we can decide which chromosomes in the initial population are good enough to survive and to reproduce offspring in the next generation. For this purpose, chromosomes are ranked from lowest fitness value to highest fitness value. Here, the objective function provided in equation (1) is used as the fitness function.

The best population members are defined by the fitness function. The best individuals with a minimum fitness value are chosen to be the new population based on the old population. This procedure is called elitism, which is used to successfully improve the optimization process. Not all chromosomes are fit enough to reproduce. In a given population, only the top N_k chromosomes are preserved for reproducing. The remaining chromosomes $(N_p - N_k)$ are eliminated to make available room for the new offspring.

Chromosomes are randomly chosen as the mothers and fathers in pairs. Each pair generates two offspring that contain information from each parent. Reproduction is simply an operation to contribute additional offspring to the next generation. Hence, the old population solutions are employed to generate a new population in this GA process. The justification is that the new generation is expected to be better than the old one. Old chromosomes (parents) are placed into a reproduction pool according to their fitness value.

Crossover is the primary genetic operator, which supports the search for new areas in the solution space. The genetically-crossed parents are chosen according to their fitness scores using the roulette wheel.

The simplest method selects one or more points on the chromosomes to mark as the crossover points. Then, the variables between these points are simply exchanged between the two parents. Two chromosomes are selected randomly as parents for the crossover

$$\begin{aligned} parent_1 &= [P_{m1}, P_{m2}, ..., P_{m\alpha}, ..., P_{mn}] \\ parent_2 &= [P_{d1}, P_{d2}, ..., P_{d\alpha}, ..., P_{dn}] \end{aligned} \tag{7}$$

where the m and d indicate the mom and dad, respectively. Next, the crossover point (α) is chosen randomly and the selected variables (as shown in (2) with index α) are combined to create new variables that will appear in the offspring [7]

$$\begin{aligned} P_{new1} &= P_{m\alpha} - \gamma\,(P_{m\alpha} - P_{d\alpha}) \\ P_{new2} &= P_{d\alpha} + \gamma\,(P_{m\alpha} - P_{d\alpha}) \end{aligned} \tag{8}$$

where γ is also a random value between 0 and 1. With the last step in (2), the crossover is completed and the offspring are obtained as

$$\begin{aligned} offspring_1 &= [P_{m1}, P_{m2}, ..., P_{new1}, ..., P_{dn}] \\ offspring_2 &= [P_{d1}, P_{d2}, ..., P_{new2}, ..., P_{mn}] \end{aligned} \tag{9}$$

Mutation, as a secondary operator, is performed after the crossover is done. A mutation operator is used to avoid the GA from falling into a local solution. This should not occur often because chromosomes will be comprehensively changed and the information from the old generation will be lost. The mutation operator is defined by a random value.

A mutation changes the structure of a randomly selected gene in a chromosome, with a small probability. We consider that $C = [P_1, P_2, \ldots, P_i, \ldots, P_n]$ is a chromosome and $P_i \in [P_i^{\max}, P_i^{\min}]$ is a gene to be mutated. After the mutation operator is applied, the gene, P_i^{new} is obtained as

$$P_i^{new} = (P_i^{\max} - P_i^{\min}) \times \gamma + P_i \tag{10}$$

where γ is a random value between 0 and 1 [7]. The new gene needs to be set to $P_i^{\max}$ or $P_i^{\min}$ if P_i^{new} is over either limit.

After mutation, all constraints are again reviewed to determine whether they have been violated or not. This process is iterated until the best solution, or a satisfactory solution, is obtained.

4. V-Q sensitivity index

The V-Q sensitivity of a bus represents the gradient of the QV curve for a given operating point. V-Q sensitivity studies evaluate the impact of reactive power injections on voltages, with the following guidelines [11]:

- A positive V-Q sensitivity shows a stable operation;
- A negative V-Q sensitivity shows an unstable operation;
- With a smaller sensitivity value, the system will be more stable;
- A sensitivity of 0 represents a completely stable system;
- An infinite sensitivity means that the system is operating at the threshold of the stability limit.

A Jacobian matrix of the system is required to perform the V-Q sensitivity analysis. The Jacobian matrix can be divided into 4 smaller sub-matrices or "sub-Jacobians", such that

$$J = \begin{bmatrix} J_{P\theta} & J_{PV} \\ J_{Q\theta} & J_{QV} \end{bmatrix} \quad (11)$$

where
$J_{P\theta}$ = Jacobian of partial derivatives of P relating to θ;
J_{PV} = Jacobian of partial derivatives of P relating to V;
$J_{Q\theta}$ = Jacobian of partial derivatives of Q relating to θ;
J_{QV} = Jacobian of partial derivatives of Q relating to V.

Using these expressions, a sensitivity value can be calculated by the formula

$$J_R = \left(J_{QV} - J_{PV} \cdot J_{P\theta}^{-1} \cdot J_{Q\theta}\right) \quad (12)$$

where the inverse of J_R (J_R^{-1}) is the reduced Jacobian matrix of the system [11,12,13]. Its i^{th} diagonal element is the V-Q sensitivity at bus i. By using this value, it is possible to decide the sensitivity value at any bus, with regard to reactive power injections at another bus or the same bus.

5. Learning automata

A learning automaton mathematically represents a control strategy to improve the performance of a system's output results using responses from the environment. The variable structure of learning automata has great elasticity because it updates the state transitions, or action probabilities, at every step with a reinforcement plan.

A learning automaton creates a series of actions based on a random environment. The learning automata method extracts the optimal action from a set of acceptable actions. These actions are performed in a random environment. Given an input action, the random environment will produce an output (response), which is probabilistically related to the input action.

There are 3 primary models. The first is called the P-model, in which the output can be only one of two values, 0 or 1. Here, "1" indicates an unfavorable response and "0" indicates a favorable response, by defining a proper threshold. The second model is called the Q-model, in which the output is composed of more than 2 values that can take any value in the interval [0,1]. The last model is the S-model, in which the output of the environment is a continuous random variable in the interval [0,1]. In this study, a P-model learning automaton is used.

The action probability vector $p(n)$ at an instant n is given by

$$P_i(n) = Pr\{\alpha(n) = \alpha_i\}, i = 1, 2, \ldots, r \tag{13}$$

where r is the number of different actions, α(n) is the action of the automaton at instant n, and α_i is the action chosen by the learning automaton.

If no prior information regarding the actions is available, all actions are chosen with an equal probability, called pure chance. In such a case, the action probability vector $p(n)$ is given by [9]

$$p_i(n) = \frac{1}{r}, \quad i = 1, 2, \ldots, r \tag{14}$$

The probability of an unfavorable output from the environment, related to an action α_i, is called the penalty probability c_i, and may be defined by [9]

$$c_i = Pr\{\beta(n) = 1 | \alpha(n) = \alpha_i\}, \quad i = 1, 2, \ldots, r \tag{15}$$

where $\beta(n)$ is the response of the environment at instant n. If a stationary random environment is considered with penalty probabilities $\{c_1, c_2, \ldots, c_r\}$, an average penalty $M(n)$ as a measure for a given action probability can be described as follows [9]

$$M(n) = \sum_{i=1}^{r} c_i p_i(n) \tag{16}$$

The average penalty is a useful component in the evaluation of the different automata. $M(n)$ is a constant for the pure-chance automaton and symbolized by M_0 [9]

$$M_0 = \frac{1}{r} \sum_{i=1}^{r} c_i \tag{17}$$

When we consider a variable-structure automaton with r different actions that work in a stationary environment with $\beta(n) = \{0,1\}$ outputs, then the action probabilities can be updated using a reinforcement scheme described mathematically as follows [9]:

If $\alpha(n) = \alpha_i$ $(i = 1, 2, \ldots, r)$ then we have

$$\begin{aligned} p_j(n+1) &= (1-a) \cdot p_j(n) \\ p_i(n+1) &= p_i(n) + a \cdot [1 - p_i(n)] \end{aligned} \tag{18}$$

for all j $\neq$ i when $\beta(n) = 0$; and

$$\begin{aligned} p_j(n+1) &= \tfrac{b}{r-1} + (1-b) \cdot p_j(n) \\ p_i(n+1) &= (1-b) \cdot p_i(n) \end{aligned} \tag{19}$$

for all $j \neq$ i when $\beta(n) = 1$ where a is a reward parameter and b is a penalty parameter ($0 < a < 1$ and $0 < b < 1$). According to equations (18) and (19), the probability of performing action α_i increases if the response related to action α_i is favorable ($\beta(n) = 0$), otherwise it decreases ($\beta(n) = 1$). The probability of performing other actions increases if the response related to action α_i is unfavorable and decreases if the response is favorable.

It can be seen from equations (18) and (19) that the equation of $\sum_{i=1}^{r} p_i(n) = 1$ is satisfied at any n^{th} stage. The probability $p(n+1)$ is decided entirely by $p(n)$, which is a discrete-time homogeneous Markov process.

The linear schemes of the learning automata are defined based on the relationship between the reward and penalty parameters. These are called the linear reward penalty (L_{R-P}) scheme (a = b), the linear reward ε-penalty ($L_{R-\varepsilon P}$) scheme (a > b), and the linear reward-inaction (L_{R-I}) scheme (b = 0). In this study, the $L_{R-\varepsilon P}$, L_{R-I} and L_{R-P} schemes with different actions are applied to the multi-objective optimization of the power system operation.

6. Proposed multi-objective methodology

The methodology of this study seeks a multi-objective solution of economic dispatch considering the generation cost and explicit voltage stability, expressed as V-Q sensitivity. This is of particular interest in systems with a serious concern regarding voltage stability.

The methodology consists of 3 primary steps. In the 1st step, a power generation schedule is selected as the most economical one in terms of production cost as the only objective. Secondly, the stability of the system is considered and the most suitable operating point for stability is determined. At the same time, power losses for different operating points are considered as the 3rd objective function. Finally, 2 operating points related to cost and stability, and the power loss values that are obtained from the previous steps, are taken into account with certain thresholds. The goal is to find the optimal operating point for the system using these 3 criteria as the multiple objective problems of economic dispatch.

The starting point of the study is to obtain the operating point with the minimum production cost for a given power system. This process is achieved by using a practical, viable genetic algorithm. The power balance constraint, the generator's unit capacity limit, transmission losses, and transmission line constraints are all considered in obtaining the economic dispatch solution.

In the 2nd step, we analyze the V-Q sensitivity under different operating conditions. After obtaining the active power values of the generator units, a simple yet effective calibration is performed. We choose a generator unit and assign it a constant value within its minimum and maximum limits. For simplification, we may simply divide the interval between the minimum and maximum values into n equal ranges and take the values as possible actions for the learning automata. One of the other units is selected as the system swing bus, and the rest remain at the values obtained in the first step. For each case (action) we run the power flow and find the Jacobian matrix (J_R) of the system, as mentioned in section IV. The inverse of this matrix provides the V-Q sensitivity value. This process is performed for every unit to obtain different operating points as actions for the learning automata algorithm. In this process, we also consider the constraints mentioned previously. The V-Q sensitivity values provide information about the system voltage stability. A sensitivity of 0 represents a completely stable system. The V-Q sensitivity values are ordered from high to low values to compare each operating point obtained in this step. The best operating point, which has the minimum V-Q sensitivity, can be obtained and is considered the best stable operating point for the system. Also, we have a power loss value for every action or operating points. The power system losses can be directly calculated from the power flow. The power loss values are ordered from the highest to lowest values to compare each operating point obtained in this step.

In the 3rd step, we define 3 performance values, $M_C^{\min}$ for the minimum cost value obtained from the 1st step, $M_S^{\min}$ for the best stable operating point, and $M_L^{\min}$ for the minimum power loss value obtained from the 2nd step. We find an optimal operating point which satisfies these 3 criteria, production cost, voltage stability and power loss, with the appropriate thresholds, chosen by the decision makers. Hence, we can describe 3 performance rates related to cost, stability, and power loss as follows [14]:

1) The ratio of the minimum generation cost value to the generation cost value of the i^{th} operating case ($M_C^{\min}/M_C^i$) is assumed to be 90% for acceptable economic operation,

2) The ratio of the minimum V-Q sensitivity value to the actual V-Q sensitivity value of the i^{th} operating case ($M_S^{\min}/M_S^i$) is assumed to be 50% for acceptable stable operation,

3) The ratio of the minimum power loss value to the power loss value of the i^{th} operating case ($M_L^{\min}/M_L^i$) is assumed to be 50% for acceptable operation, considering the power loss.

Each performance ratio is normalized to provide a sigmoid output between 0 and 1 around the threshold value. The sum of the performance ratios, for every action, is entered into the environment to create an output $\beta(n)$ for the learning automata algorithm. Since the performance ratios are greater than the thresholds, the power system operation will be favorable ($\beta(n) = 0$). Likewise, if the performance ratios are smaller than the thresholds, the power system operation will be unfavorable ($\beta(n) = 1$). Finally, the learning automata algorithm will find the optimal action which provides the optimal operation point of the power system, considering cost, losses and stability.

7. Test results

The proposed methodology is applied to the IEEE 30-bus, 6-generators test system which is modified from reference [15]. The system demand is 190 MW. The fuel-cost coefficients and generation limits for each unit are shown in Table 1. The system one-line diagram is shown in Figure 1 for illustration.

Table 1. Generating unit capacity and coefficients for 6 generators in a 30-bus system.

Gen.	Bus No.	a_i $\left(\frac{\$}{h}\right)$	b_i $\left(\frac{\$}{MW\cdot h}\right)$	c_i $\left(\frac{\$}{MW^2\cdot h}\right)$	P_{max} (MW)	P_{min} (MW)
1	1	7.5	2	0.02	80	10
2	2	5	1.75	0.018	80	10
3	13	12	3	0.025	40	5
4	22	12.5	1	0.063	50	5
5	23	12	3	0.05	30	5
6	27	10	3.25	0.08	55	10

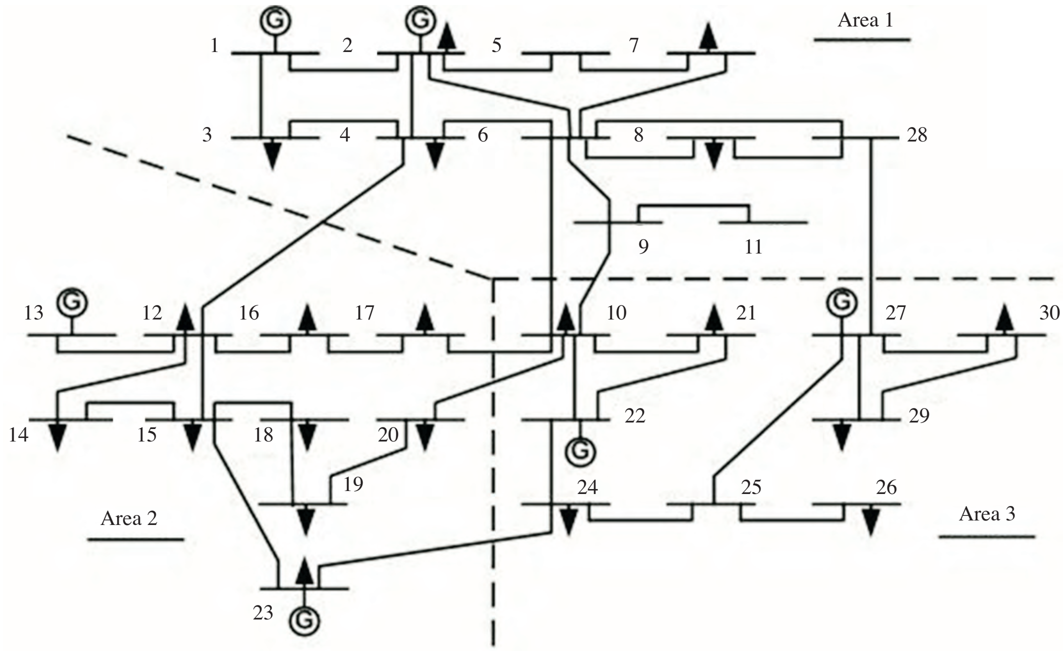

Figure 1. One-line diagram of the IEEE 30-bus test system.

The current work uses MATLAB for simulation. First, the network-constrained economic dispatch solution with the proposed genetic algorithm is implemented.

The parameter settings are taken as follows:

- Population size = 20
- Probability of crossover = 0.5
- Probability of mutation = 0.01

Figure 2 illustrates the generation cost evolution during the iterative process of the genetic algorithm.

Table 2 shows the solutions for the economic dispatch, considering the transmission line constraints and line losses. The results obtained from the proposed approach are compared with the AC optimal power flow (ACOPF) solution in MATPOWER [15], which shows that the GA method proposed in this paper provides even better results. The possible reason is that the ACOPF solution in MATPOWER may be trapped in a local optimum, which can be avoided with the proposed GA method.

Table 2. Results for the economic dispatch.

	Proposed GA Solution	MATPOWER
$f(P_i)$ \$/h	638.48	638.92
P_1(MW)	44	44.98
P_2(MW)	56.53	57.83
P_3(MW)	23	18.44
P_4(MW)	37	23.05
P_5(MW)	16	16.60
P_6(MW)	16	31.70
P_L(MW)	2.53	2.60

After obtaining the economic dispatch solution for cost minimization, we made a small calibration to find the different actions, as mentioned in section VI. Table 3 shows the active power values (P_i) of 10 actions for the learning automata as a demonstration, although the actual number of actions can be larger. Action 7 gives the minimum economical operating point whereas action 6 provides the best stable operating point. Action 3 is the minimum active power loss value.

Table 3. Active power values of actions.

Actions	P_1 (MW)	P_2 (MW)	P_3 (MW)	P_4 (MW)	P_5 (MW)	P_6 (MW)
1	38	62.47	23	37	16	16
2	44	52	27.44	37	16	16
3	20.38	80	23	37	16	16
4	44	56.53	33	37	5.88	16
5	72.63	56.53	23	9.5	16	16
6	44	56.53	37.02	23	16	16
7	44	56.53	23	37	16	16
8	44	56.53	31.39	37	7.5	16
9	44	56.53	28.91	37	16	10
10	44	56.53	23	18.63	16	35

Figure 3 shows the simulation results of the learning automata for $L_{R-\varepsilon P}$ scheme with a = 0.015 and b = 0.0015. The probability of action 8 converges to 1 as the trial numbers increase. This demonstrates that action 8 is the optimal solution. Figure 4 shows that since the probability of action 8 increases to 1, the probabilities of the other actions decrease to 0 at the same time. As shown in Figure 5, the average penalty value decreases to its minimum by the trial number. In the same way, the simulation results, when the L_{R-I} scheme is applied with a = 0.015 and b = 0.0, are given in Figures 6-8. The simulation results, when the L_{R-P} scheme is applied with a = 0.015 and b = 0.015, are given in Figures 9-11. When we compare the 3 learning schemes, the L_{R-I}

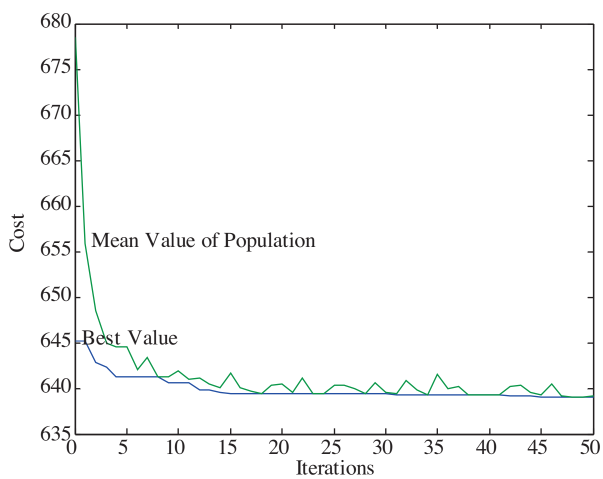

Figure 2. Generation cost evolution during the iterative process.

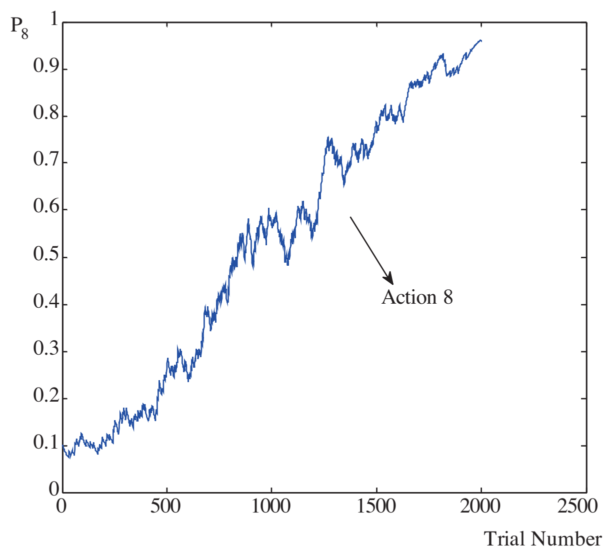

Figure 3. Probability of action 8 w.r.t. trial numbers in the $L_{R-\varepsilon P}$ scheme.

scheme is the fastest, but it may have an undesirable absorbing state where a probability reaches 0 and remains in that state; the L_{R-P} scheme has the slowest convergence speed with rather large oscillations, and the $L_{R-\varepsilon P}$ scheme has no absorbing state and a reasonably fast convergence speed [9].

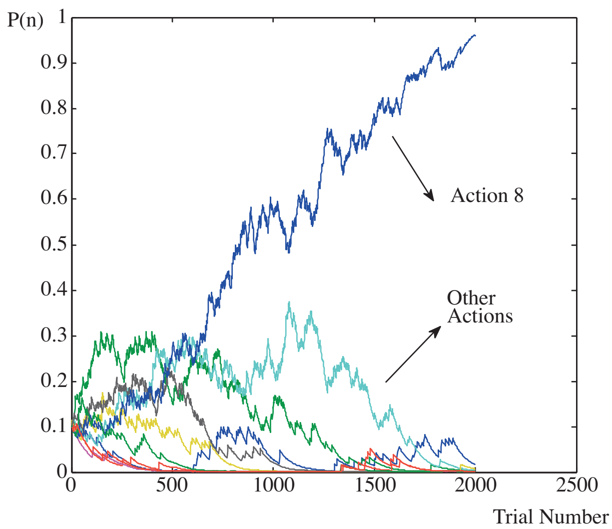

Figure 4. Probabilities of actions in the $L_{R-\varepsilon}$ scheme.

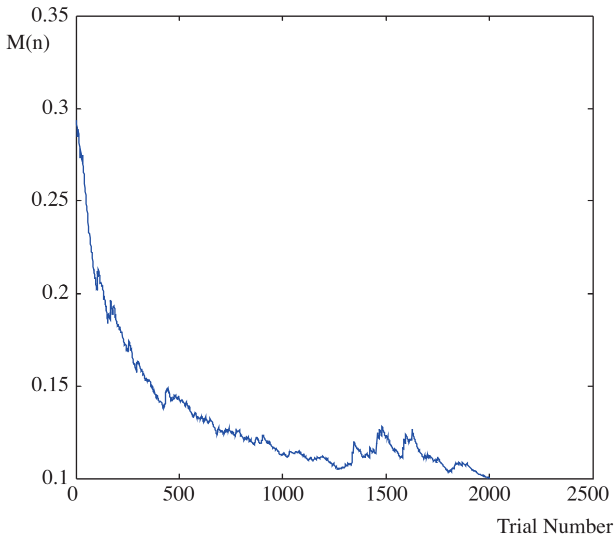

Figure 5. Average penalty in the $L_{R-\varepsilon}$ scheme.

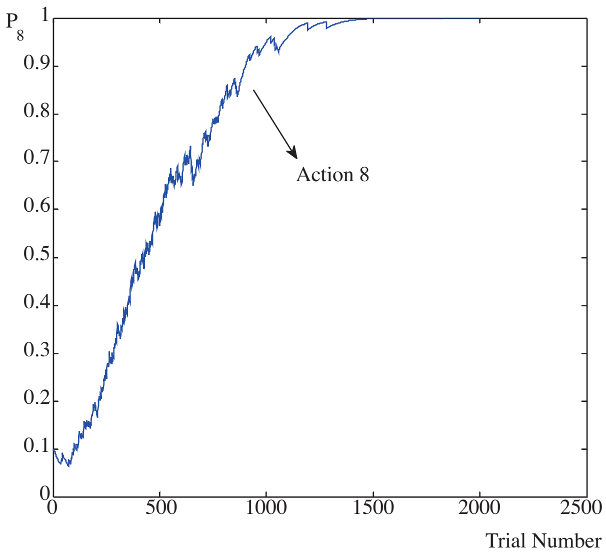

Figure 6. Probability of action 8 w.r.t. trial numbers in the L_{R-I} scheme.

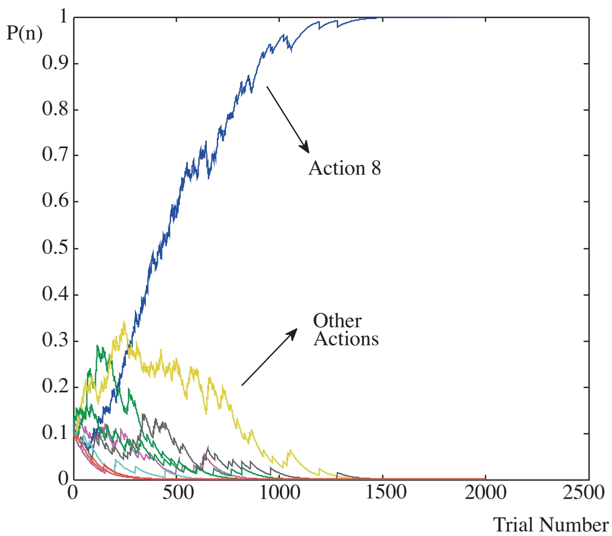

Figure 7. Probabilities of actions in the L_{R-I} scheme.

From the simulation results of the learning automata, we can see that action 8 is the best solution. It satisfies both the economical operation and voltage stable criteria simultaneously. The cost is calculated as \$645.12 for this operating point.

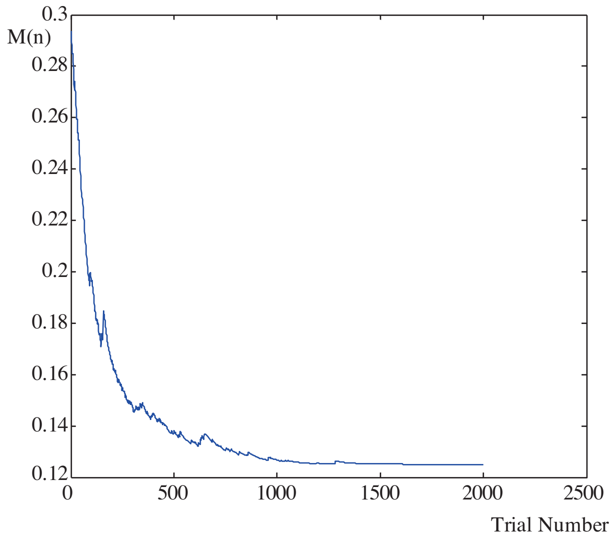

Figure 8. Average penalty in the L_{R-I} scheme.

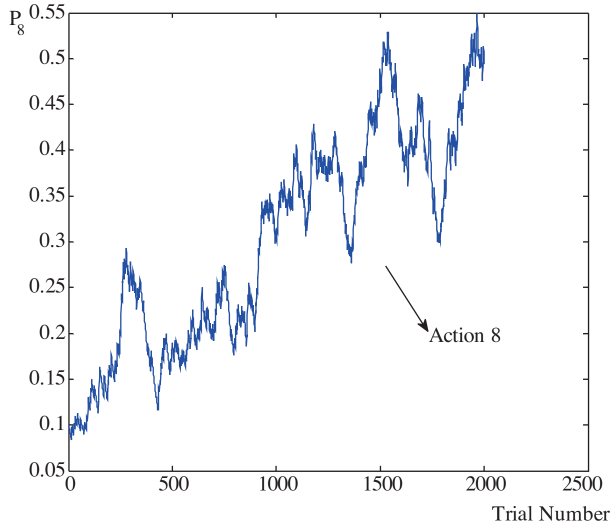

Figure 9. Probability of action 8 w.r.t. trial numbers in the L_{R-P} scheme.

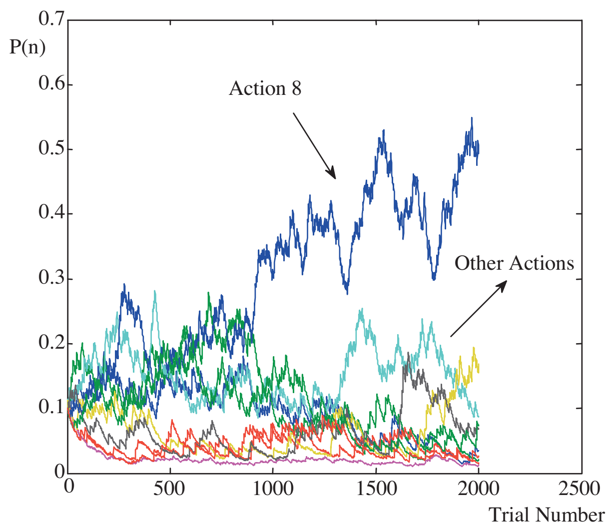

Figure 10. Probabilities of actions in the L_{R-P} scheme.

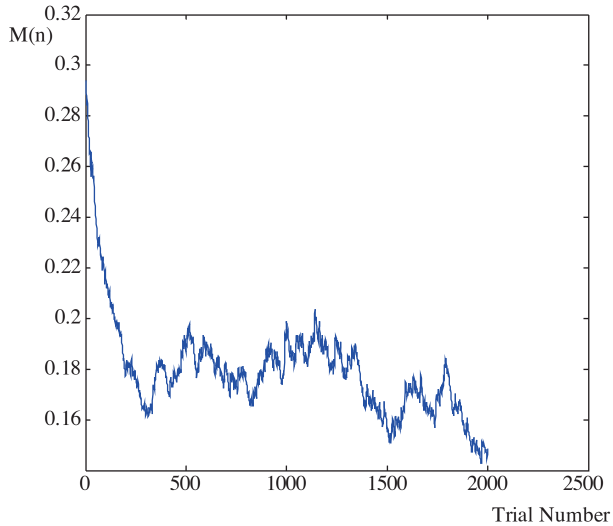

Figure 11. Average penalty in the L_{R-P} scheme.

The features and contributions of the comprehensive methodology presented in this paper can be summarized as follows:

- This work presents a learning automata algorithm to obtain a multi-objective optimal operation point considering cost, voltage stability, and power losses simultaneously.
- The proposed learning automata algorithm essentially considers network constraints because one of its fundamental modules, the proposed genetic algorithm for initial generation dispatch, considers network constraints. This is an improvement from the popular practice in many previous works that ignore the network model in unit commitment or economic dispatch when economics and stability are simultaneously considered using multi-objective optimization.

- This work addresses voltage stability rather than other forms of instability, because voltage stability is an increasing concern in the present power system operation, in particular, for systems with insufficient voltage support or with voltage instability more dominant than other stability problems.
- The simulation result based on a standard test system, the IEEE 30-bus system with 6 generators, shows the robustness and feasibility of the application of learning automata to solve power system dispatch considering multiple objectives like cost, voltage stability, and system losses.

8. Conclusions

In this study, a practical and efficient genetic algorithm is presented for solving the network constrained economic dispatch problem. The V-Q sensitivity index and power loss values are used to compare the different operating points of power systems. Finally, the learning automata algorithm is presented and used to obtain a multi-objective optimal operation point which satisfies the cost, stability, and power loss criteria. The methodology was implemented in MATLAB 7.8 and applied to the IEEE 30-bus test system. The simulation results demonstrate that the methodology is satisfactory.

The proposed method has good potential to solve the economic dispatch in a weak system with a strong need for considering voltage stability during generation scheduling.

Acknowledgements

In this study, Dr. Arif Karakas was supported by TÜBİTAK 2219 under a postdoctoral research grant for his studies at the University of Tennessee, Knoxville, TN, USA.

References

[1] B.H. Chowdhury, S. Rahman, "A review of recent advances in economic dispatch", IEEE Transactions on Power Systems, Vol. 5, pp. 1248-1259, 1990.

[2] A.J. Wood, B.F. Wollenberg, Power Generation, Operation and Control, New York, Wiley, 1996.

[3] R.C. Bansal, "Optimization methods for electric power systems: an overview", International Journal of Emerging Electric Power Systems, Vol. 2, Article 1021, 2005.

[4] A. Bakirtzis, V. Petridis, S. Kazarlis, "Genetic algorithm solution to the economic dispatch problem", IEE Proceedings on Generation, Transmission and Distribution, Vol. 141, pp. 377-382, 1994.

[5] D.E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning, Boston, Addison-Wesley, 1989.

[6] J.H. Holland, Adaptation in Natural and Artificial Systems, Ann Arbor, MI, Michigan Univ. Press, 1975.

[7] R.L. Haupt, S.E. Haupt, Practical Genetic Algorithms, New York, Wiley, 2004.

[8] M.L. Tsetlin, "Automaton Theory and Modeling of Biological Systems", Mathematics in Science and Engineering, Vol. 102, New York, Academic Press, 1973.

[9] K.S. Narendra, M.L. Thathachar, Learning Automata: An Introduction, Englewood Cliffs, NJ, Prentice Hall, 1989.

[10] K.S. Narendra, M.A.L. Thathachar, "Learning automata: a survey", IEEE Transactions in Systems, Man and Cybernetics, Vol. SMC-4, pp. 323-334, 1974.

[11] P. Kundur, Power System Stability and Control, McGraw-Hill, Inc., 1994.

[12] Y.H. Hong, C.T. Pan, W. Lin, "Fast calculation of a voltage stability index of power systems", IEEE Transactions on Power Systems, Vol. 12, pp. 1555-1560, 1997.

[13] J.A. Momoh, M.E. El-Hawary, Electric Systems, Dynamics and Stability with Artificial Intelligence Applications, New York, Marcel Dekker, 2000.

[14] B.H. Lee, K.Y. Lee, "Application of S-model learning automata for multi-objective optimal operation of power systems", IEE Proceedings on Generation, Transmission and Distribution, Vol. 152, pp. 295-300, 2005.

[15] R.D. Zimmerman, C.E. Murillo-Sanchez, D. Gan, MATPOWER: A Matlab Power System Simulation Package, 2006.