Application of S-model learning automata for multi-objective optimal operation of power systems

B.H. Lee and K.Y. Lee

Abstract: A learning automaton systematically updates a strategy to enhance the performance of a system output. The authors apply, a variable-structure learning automaton to achieve a best compromise solution between the economic operation and stable operation in a power system when the loads vary randomly. Both the generation cost for economic operation and the modal performance measure for stable operation of the power system are considered as performance indices for multi-objective optimal operation. In particular, it is shown that the S-model learning automata can be applied satisfactorily to the multi-objective optimisation problem to obtain the best trade-off between the conflicting objectives of economy and stability in the power system.

1 Introduction

An automaton acting so as to improve its performance in an unknown random environment is referred to as a learning automaton. Learning automata have attracted considerable interest in the last decades due to their potential usefulness in various engineering problems that are generally characterised by nonlinearity or a high level of uncertainty. Learning automata are based on the theories of probability and Markov processes. Tsetlin [1] first studied the behaviour of deterministic automata functioning in random environments. The fixed-structure learning automata learn to choose asymptotically better actions with a higher probability, while the state transition probabilities remain fixed. On the other hand, variable-structure stochastic automata update either the transition probabilities or the action probabilities on the basis of the input at each stage, leading to greater flexibility.

There is a body of literature on the fixed-structure and variable-structure learning automata. Cover and Hellman [2] studied a two-action fixed-structure scheme with finite memory, and Aso and Kimura [3] showed what kind of logical structure leads to expedient automata with multiple actions. Viswanathan and Narendra [4] studied the reinforcement scheme for variable-structure learning automata and its asymptotic characteristics. Narendra and Thathachar [5] carried out the outstanding work on a wide range of learning automata including learning algorithms, asymptotic behaviour and a hierarchical system. Thathachar and Phansalkar [6] proposed learning algorithms for feedforward connectionist systems in a reinforcement-learning environment. Najim and Poznyak [7] studied a multimodal searching technique in an environment with a changing number of actions of the automata. Howell and Gordon [8] proposed a genetic adaptation and population-based approach to increase the speed of convergence of the interconnected learning automata and to escape from local minima. Agache and Oommen [9] introduced a generalisation of the learning method of the pursuit algorithms that pursues the actions that have higher estimates than the currently chosen action, and hence minimising the probability of pursuing a wrong action. Papadimitriou et al. [10] proposed a new P-model absorbing learning automaton, which is based on the use of a stochastic estimator in order to achieve a rapid convergence.

A learning automaton generates a sequence of actions on the basis of its interaction with the random environment and the environment responds to the input action by producing its output (the input to the automaton) that is probabilistically related to the input action. There are three types of models for learning automata: (i) the P-model; (ii) the Q-model; and (iii) the S-model. In a P-model learning automaton, the output of the environment can take only one of two values, zero or one, with one corresponding to 'unfavourable' and zero corresponding to a 'favourable' response based on a suitably defined threshold. In a Q-model, the output set is composed of a finite number of discrete values in the interval [0, 1]. When the output of the environment is a continuous random variable that assumes values in the interval [0, 1], it is then referred to as a S-model. We will adopt the S-model in our work since it represents the most general version of the linear model environment.

A decision based on an engineering trade-off is made by simultaneously considering multiple quality criteria. In power systems, it is usually required to simultaneously optimise more than one of the power system attributes. Both producing power economically and maintaining the system stability are two important goals for the utility business, and yet these are in general conflicting objectives. This multi-objective problem requires a best compromise solution. There are several papers on the application of multi-objective optimisation methods in the power systems area. Jung *et al.* [11] studied optimal reactive power dispatch with the objectives of economy of operation and system security. Wadhwa and Jain [12] studied an optimal load flow problem to simultaneously minimise both the cost of

IEE Proceedings online no. 20040698

doi:10.1049/ip-gtd:20040698

Paper first received 25th February 2004. Originally published online: 5th November 2004

B.H. Lee is with the Department of Electrical Engineering, University of Incheon, 177, Dohwadong, Namgu, Incheon 402-749, Korea

K.Y. Lee is with the Department of Electrical Engineering, The Pennsylvania State University, 121, University Park, PA 16802, USA

generation and transmission loss. Gardunao-Ramirez and Lee [13, 14] presented a multi-objective optimisation technique in generating set points for power plants.

We intend to apply S-model learning automata in a multi-objective optimisation problem to obtain the best trade-off among the conflicting economy and stability objectives in a power system. The performance indices to be considered are the generation cost for economic operation and the modal performance measure for stable operation of a power system [15]. The multi-objective optimisation is performed in a simple six-bus test system, and the simulation results are compared for various S-model learning automata and are also compared with results obtained using Monte Carlo techniques. To date there has been no direct application of learning automata in power system problems or in solving multiple objective problems. We intend to propose a procedure to apply learning automata to solving multi-objective problems. This method presents the best compromise solution that satisfies the given criteria in the probabilistic sense when loads vary randomly.

2 S-model learning automata

A learning automaton generates a sequence of actions on the basis of its interaction with the random environment and the actions of the automaton are various alternatives to provide for the environment. The automaton approach to learning involves the determination of an optimal action out of a set of allowable actions. These actions are performed on a random environment and the environment responds to an input action by producing an output that is probabilistically related to the input action.

We can consider the action probability vector $\boldsymbol{p}(n)$ at an instant n whose ith component $p_i(n)$ is defined by

$$p_i(n) = \Pr[\alpha(n) = \alpha_i] \quad i = 1, 2, \ldots, r \tag{1}$$

where r is the number of different actions, $\alpha(n)$ is the action of the automaton at instant n and α_i is an action selected by the automaton.

A penalty probability s_i is the probability of obtaining response $\beta(n)$ corresponding to an action α_i, and may be defined by:

$$\Pr[\beta(n)|\alpha(n) = \alpha_i] = s_i \; i = 1, 2, \ldots, r \tag{2}$$

where $\beta(n)$ is the response of the environment at instant n. The S-model automaton is the automaton whose response can take continuous values over the unit interval [0, 1]. The response $\beta(n)$ in the S-model means the degree of unfavourableness, which approaches zero if the response is favourable and approaches one if the response is unfavourable.

When a stationary random environment with penalty probabilities $\{s_1, s_2, \ldots s_r\}$ is considered, a quantity $M(n)$ that is the average penalty for a given action probability vector is defined as follows:

$$\begin{aligned} M(n) &= E[\beta(n)|\boldsymbol{p}(n)] \\ &= \sum_{i=1}^{r} E[\beta(n)|\boldsymbol{p}(n), \alpha(n) = \alpha_i] \Pr[\alpha(n) = \alpha_i] \\ &= \sum_{i=1}^{r} s_i \, p_i(n) \end{aligned} \tag{3}$$

where $E[\cdot]$denotes the mathematical expectation. This average penalty $M(n)$ plays a useful role in comparing various automata.

The automaton is represented by the action probability sequence $\{\boldsymbol{p}(n)\}$, which is a discrete-time Markov process on a suitable state space. Let a variable-structure automaton with r actions operate in a stationary environment with $\beta(n) \in [0, 1]$. A general linear reinforcement scheme in the S-model for updating action probabilities can be represented as follows [5]:

$$\begin{aligned} &\text{If } \alpha(n) = \alpha_i \text{ and } i \in \{1, \cdots, r\} \text{ then} \\ &p_i(n+1) = p_i(n) - \beta(n) b p_i(n) + [1 - \beta(n)] a (1 - p_i(n)) \\ &\text{otherwise} \end{aligned} \tag{4}$$

$$\begin{aligned} p_j(n+1) = &p_j(n) + \beta(n)[b/(r-1) - b p_j(n)] - \\ &[1 - \beta(n)] a p_j(n) \quad \text{for all } j \neq i \end{aligned} \tag{5}$$

where a is a reward parameter in [0, 1] and b is a penalty parameter in [0, 1]. Equation (4) means that the probability of taking an action α_i increases if the response corresponding to an action α_i is favourable ($\beta(n)$ is close to zero) and decreases otherwise ($\beta(n)$is close to one). Equation (5) means that the probabilities of taking other actions increase if the response corresponding to an action α_i is unfavuorable and decreases otherwise. It can be seen that the property of probability, $\sum_{j=1}^{r} p_j(n) = 1$, is satisfied for all n whenever the action is selected randomly at all stages. The probability $\boldsymbol{p}(n+1)$ is determined completely by $\boldsymbol{p}(n)$, and $\{\boldsymbol{p}(n)\}$ is a discrete-time homogeneous Markov process, where the value at stage n depends only on the value at stage $n-1$.

Selection of the reward and penalty parameters, a and b, dictates the property of the learning automata. The linear reinforcement schemes in the S-model for $a=b$, $a>b$ and $b=0$ are called, respectively, the linear reward-penalty ($SL_{\mathrm{R-P}}$) scheme, the linear reward ε-penalty ($SL_{\mathrm{R-\varepsilon P}}$) scheme and the linear reward-inaction ($SL_{\mathrm{R-I}}$) scheme. The linear reward-penalty ($SL_{\mathrm{R-P}}$) scheme and the linear reward ε-penalty ($SL_{\mathrm{R-\varepsilon P}}$) scheme are not dependent on initial conditions since they converge to the optimal value irrespective of the initial conditions. However, the linear reward-inaction ($SL_{\mathrm{R-I}}$) scheme can be dependent on the initial conditions because it can have an absorbing state, that is, the state can be trapped with probability one [5]. These three linear reinforcement schemes with multiple actions will now be applied to the multi-objective optimisation problem for power system operation.

3 Multi-objective optimal operation of a power system

3.1 Problem formulation

A multi-objective optimisation problem in a power system can be represented in a compact form:

$$\begin{aligned} &\text{minimise} : \boldsymbol{J}(\boldsymbol{x}) = \{J_1(\boldsymbol{x}), J_2(\boldsymbol{x}), \cdots, J_n(\boldsymbol{x})\}^{\boldsymbol{t}} \\ &\text{Subject to} : x \in \boldsymbol{g}_{\mathrm{c}} \end{aligned} \tag{6}$$

where $\boldsymbol{J}(\boldsymbol{x})$ is a vector of multi-objective functions, $\boldsymbol{x}$ is a vector of decision variables, $\boldsymbol{t}$ represents transpose and $\boldsymbol{g}_{\mathrm{c}}$ represents the feasible region in the decision space which is spanned by the decision variables and it includes various constraints.

Under a random environment the load varies randomly and the objective functions become random variables. Then, the multi-objective optimisation problem can be represented in the following form:

$$\begin{aligned} &\text{maximise} : \Pr\{\boldsymbol{J}(\boldsymbol{x}) \leq \boldsymbol{J}_s\} \\ &\text{Subject to} : x \in \boldsymbol{g}_{\mathrm{r}} \end{aligned} \tag{7}$$

where $\boldsymbol{J}_{\mathrm{s}}$ is a vector of specified values of objective functions that are considered to be satisfactory. Equation (7) means to maximise the probability that $\boldsymbol{J}(\boldsymbol{x})$ may be less than J_s, satisfying the given constraints.

This optimisation problem can be solved by using learning automata. The concept of learning automata application to the multi-objective problem is shown in Fig. 1, where the random environment represents a target system, which may have uncertainties and random noises.

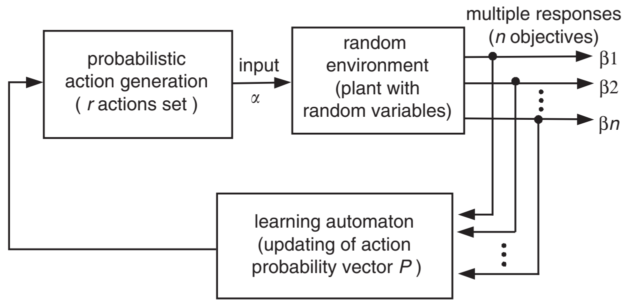

Fig. 1 *Application of a learning automaton to the multi-objective optimisation problem*

The procedure in the learning automata is summarised as follows:

1. Extremum points are determined by independently, minimising each single objective function and r actions are appropriately selected over the space enclosed by the extremum points.
2. The probabilities of r actions (probability vector: $\boldsymbol{p}(n) = \{p_1(n), p_2(n), \cdots, p_r(n)\}^t$) are updated according to the degree of favourableness of the response in the random environment (In general, the initial probabilities are of equal probability in (4) and (5)).
3. An action is randomly selected under the probability vector of r actions and applied to the environment.
4. The random environment responds to the input and the performance of $\boldsymbol{J}(\boldsymbol{x})$ is assessed by evaluating the output according to the given criteria.
5. The above steps 2, 3 and 4 are repeated until the probability vector converges to a fixed value.
6. After convergence, the action that has the highest probability in a learning automaton is the best compromise solution that simultaneously satisfies the multi-criteria.

To illustrate this concept, two objective functions are considered namely: (i) the generation cost for economic operation; and (ii) a modal performance measure for stable operation of the power system.

3.2 Multi-objective functions

3.2.1 Economic operation of a power system:

The first goal of power system operation is to minimise the operation cost. For simplicity of analysis, it is assumed that the cost function for economic operation is given by the total summation of the generation fuel costs, which can be expressed as a quadratic function of the generating powers:

$$J_1(\boldsymbol{P}_{sg}) = \sum_{k \in G} (a_k + b_k P_k + c_k P_k^2) \tag{8}$$

where G is a set of indices of generator buses including the swing bus, J_1 is the generation cost function, $\boldsymbol{P}_{sg}$ is the real powers of the generator buses including the swing bus, and a_k,b_k and c_k are generation cost parameters. The steepest descent method [16] is used to minimise this function.

3.2.2 Stable operation of a power system: -

The second important goal is to enhance the dynamic stability of a power system. The following modal performance measure is used for a rapid decay of the mode envelopes derived from the mathematical model for dynamic stability [13]:

$$J_2(\boldsymbol{P}_{sg}) = \sum_{j=0}^{n} J_{Sj} \tag{9}$$

with

$$J_{Sj} = \int_0^T \sum_{i=1}^{n} z_{j,i}^t W_j z_{j,i} * \mathrm{dt} \text{ for the } j\text{th state} \tag{10}$$

where T is an integration time interval, $\boldsymbol{z}(t)$ is the output error from a reference, $\boldsymbol{z}_{j,i}$ is the ith mode of the component of $\boldsymbol{z}(t)$ that depends on the jth state of the state vector $\underline{\boldsymbol{x}}(t)$ in the system dynamics, $\boldsymbol{W}_j$ is a weighting matrix for the jth state, and superscript (*) denotes complex conjugate. A detailed description of the mathematical model for the dynamic stability can be referred to in [13]. The gradient of the modal performance measure has been evaluated in [13]. We will use the steepest descent method [16] to obtain the minimum of the performance measure J_2.

4 Simulation

4.1 Power system model

The system used for simulation is the six-bus power system shown in Fig. 2. Buses 1 and 2 are generator buses and others are load buses. The line data and the initial data for generation and load of the power system are given in Tables 1 and 2, respectively. Bus 1 is a swing bus. It is assumed that loads contain uncertainty and are Gaussian

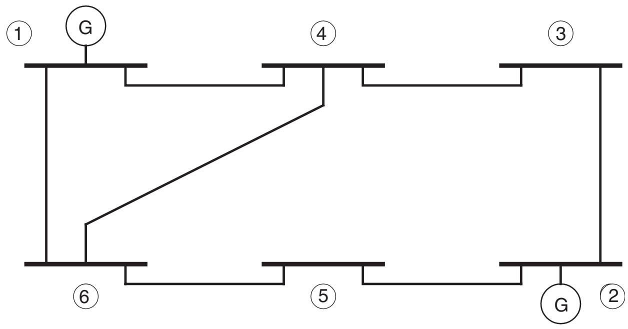

Fig. 2 *The six-bus power system*

Table 1: Line data for the six-bus power system

Line	From bus	To bus	Line impedance p.u. R	X
1	1	6	0.135	0.962
2	1	4	0.100	0.756
3	4	6	0.120	0.795
4	5	6	0.110	0.873
5	2	5	0.142	0.983
6	2	3	0.181	1.210
7	3	4	0.050	0.410

Table 2: Initial data of generation and load of the system (unit: p.u.)

Bus	Voltage magnitude	Voltage angle	P	Q
1	1.0	0.0		
2	1.0		0.32	
3			−0.27	−0.06
4			0.00	0.00
5			−0.19	−0.05
6			−0.28	−0.03

with a 3% standard deviation and that the power factor at each bus remains constant. The model for the analysis of the modal performance measure includes the nonlinear machine model with a two-axis representation of the generator and the IEEE type-1 excitation system [13]. The data for the generation cost parameters are given in Table 3.

Table 3: Generation cost data

Cost coefficients of generator 1		Cost coefficients of generator 2	
a_1	52.0	a_2	88.0
b_1	1.12	b_2	1.91
c_1	0.0021	c_2	0.0035

the threshold point. As the ratios of the performance measures become greater than the given thresholds, the power system operation becomes more favourable and the value of $\beta(n)$ approaches zero. Similarly, as the performance ratios become less than the given thresholds, the power system operation becomes less favourable and the value of $\beta(n)$ approaches one.

Since the power system operation in this model depends on the generation power of generator 2, the generation power of generator 2 can be considered as the action of the learning automaton. The interval between $P_{G2E}^{\min}$ and $P_{G2S}^{\min}$ is considered as the range of actions that can take place and it is discretised with equal increments to give ten candidate compromise solutions. The values (P_{G2}) of the actions are shown in Table 4. The problem of solving for a best compromise solution that simultaneously satisfies both the economic operation criterion and the stable operation criterion is reduced to that of determining the best action to satisfy the given criteria in the probabilistic sense. The system is under a random environment where the uncertainly of the load is represented by a normally distributed random variable with a 3% standard deviation. The random loads are generated repeatedly and the power system is analysed for the given values of the loads. Then the learning automata learn to choose better actions to improve the performance, making the solution converge to a best compromise solution. The reward parameter a and the penalty parameter b are selected suitably by trial and error. If the values of a and b are large, the trajectories converge with a large oscillation. On the other hand, if the values are small, the trajectories are smooth, but converge slowly.

Table 4 P_{G2} values of actions. (unit: p.u.)

Actions	Action 1	Action 2	Action 3	Action 4	Action 5	Action 6
P_{G2}	0.072	0.106	0.141	0.175	0.210	0.244
Actions	Action 7	Action 8	Action 9	Action 10	Action 11	
P_{G2}	0.279	0.313	0.348	0.382	0.417	

4.2 Performance evaluation

If the power of generator 2 is specified, the power of generator 1 can be determined through load flow. Therefore, the generation dispatch in this system is to determine the real power of generator 2. For the case considered, the real power ($P_{G2E}^{\min}$) of generator 2 to safely minimise the generation cost is 0.072 p.u., whereas the real power ($P_{G2S}^{\min}$) to safely minimise the modal performance measure for the enhancement of power system stability is 0.417 p.u. Suitable values of the thresholds used for determining the extent of satisfaction are decided by the decision maker, depending on various conditions. Here, they are selected so that the performance ratio, i.e. the ratio of the minimum value of the generation cost to the actual value of the generation cost, $J_1(P_{G2E}^{\min})/J_1(P_{G2E})$, to be 90% for economic operation and, similarly, the performance ratio of the minimum value of the modal performance measure to the actual value of the modal performance measure, $J_2(P_{G2S}^{\min})/J_2(P_{G2S})$, to be 50% for stable operation. Each performance ratio is normalised to give a sigmoid output between zero and one around the threshold value. The sum of the performance ratios are added as an input to generate the response $\beta(n)$, which is a monotonically decreasing sigmoid function with one-half at the threshold point.

4.3 Simulation results

The simulation results for the SL_{R-P} scheme with $a=b=0.02$ are shown in Fig. 3. The probability of action 5 converges to one as the trial number n increases and this

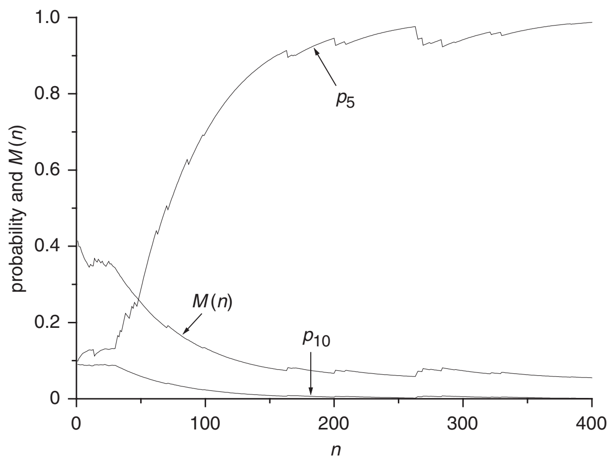

Fig. 3 *Probabilities of actions and average penalty in the SL_{R-P} scheme*

implies that action 5 is optimal. As the probability of action 5 increases to one, the probabilities of the other actions asymptotically decrease to zero. It is shown that the trajectory of the probability of action 10 asymptotically decreases to zero and the average penalty M decreases to its minimum value as the trial number n increases. The value of M converges from the initial value of 0.414 to 0.05. Similarly, the simulation results for the $SL_{R-\varepsilon P}$ scheme with $a = 0.02$ and $b = 0.002$ are shown in Fig. 4, and the results for the SL_{R-I} scheme with $a = 0.02$ and $b = 0.0$ are shown in Fig. 5. Comparing the three cases, the SL_{R-P} scheme is the slowest in terms of the speed of convergence with somewhat large fluctuations. Among the three schemes the SL_{R-I} scheme is the fastest, but it may have an absorbing state from which the state cannot escape [5]. On the other hand, the $SL_{R-\varepsilon P}$ scheme has no absorbing state and is relatively fast in speed. The simulation results of the $SL_{R-\varepsilon P}$ and SL_{R-I} schemes are very close to one an other. The performances of the SL_{R-P} and SL_{R-I} schemes are compared in Fig. 6. From the simulation results, we can see that action 5, i.e. $P_{G2} = 0.210$ p.u. is the best solution that satisfies both the criteria simultaneously. Considering the fact that the threshold values are somewhat flexible, this solution seems to be satisfactory enough for practical application. We have shown that S-model learning automata can be applied in power systems, which are under a random environment and are difficult to solve analytically.

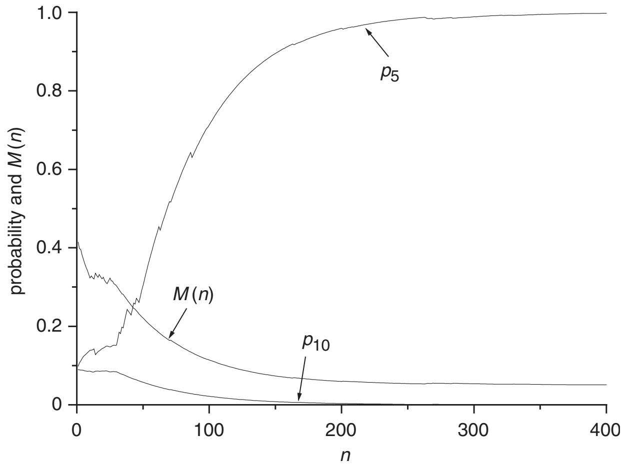

Fig. 4 *Probabilities of actions and average penalty in the* $SL_{R-\varepsilon P}$ *scheme*

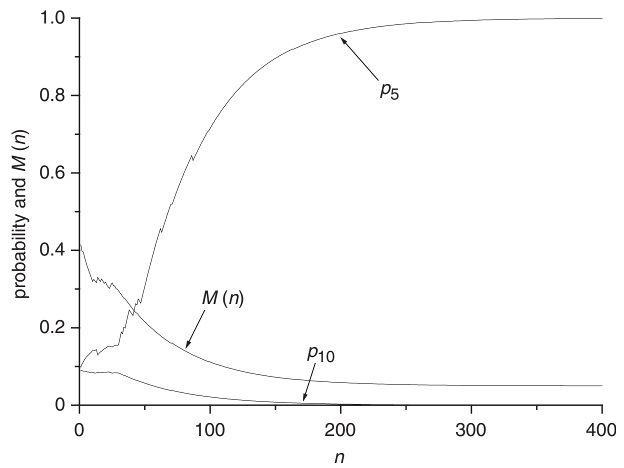

Fig. 5 *Probabilities of actions and average penalty in the* SL_{R-I} *scheme*

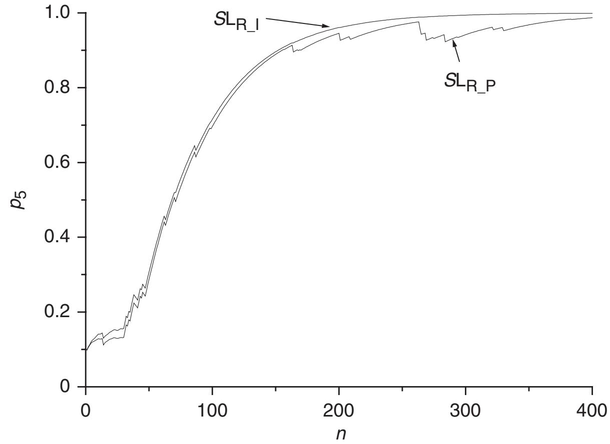

Fig. 6 *Comparison of probabilities of action 5 in the* SL_{R-P} *and* SL_{R-I} *schemes*

In a large-scale power system where a number of generators participate in actions, a base case is first analysed and the corresponding operating point($\boldsymbol{P}_{GB}$) is determined. Then, for the case considered, the real power vector ($\boldsymbol{P}_{\mathrm{GE}}^{\min}$) of the generators to solely minimise the generation cost is determined. Also, the real power vector ($\boldsymbol{P}_{\mathrm{GS}}^{\min}$) to solely minimise the modal performance measure for the enhancement of power system stability is determined. The vector space enclosed by the two extrema, $\boldsymbol{P}_{\mathrm{GE}}^{\min}$ and $\boldsymbol{P}_{\mathrm{GS}}^{\min}$, forms the range of actions. From the selected actions in the range the best compromise solution can be determined using our proposed procedure.

5 Comparison with the Monte Carlo method

The least-squares method is usually used to solve the multiple objective problems. In this method, weightings are given to each objective and the weighted objective functions are added up. Consequently, the problem is reduced to minimising the sum of all the weighted objectives. On the other hand, the Monte Carlo method is used to solve problems with random variables. In a Monte Carlo analysis, the simulation is repeatedly performed in order to obtain as many solutions as possible and hence determine the probabilistic distribution of the solutions in the random environment. By performing many simulations, the best solution in a probabilistic sense can be obtained. Under a random environment the load varies randomly and the objective functions become random variables. Then, the multi-objective optimisation problem can be represented in the following form:

$$\begin{aligned} &\text{maximise} : J(\boldsymbol{x}) = \sum_{i=1}^{n} W_i J_i(\boldsymbol{x})/J_{\mathrm{B}i} \\ &\text{subject to} : \boldsymbol{x} \in \boldsymbol{g}_{\mathrm{c}} \end{aligned} \tag{11}$$

where W_i is the weighting of the ith performance measure and J_{Bi} is the value of the ith performance measure of the base case, which is used to normalise each performance measure. In the case of the two performance measures, namely the generation cost for economic operation and the modal performance measure for stable operation, the objective function becomes: $J(\boldsymbol{x}) = W_1 J_1(\boldsymbol{x})/J_{\mathrm{B}1} + W_2 J_2(\boldsymbol{x})/J_{\mathrm{B}2}$.

The simulation results obtained using the Monte Carlo method under the same random environment as in the case of the learning automata are shown in Fig. 7. Here, the weightings of the economic operation and the stable operation performance measures are the same ($W_1 = 0.5$ and $W_2 = 0.5$). The small circles in the Figure indicate the optimum values obtained to each to iteration. Considering

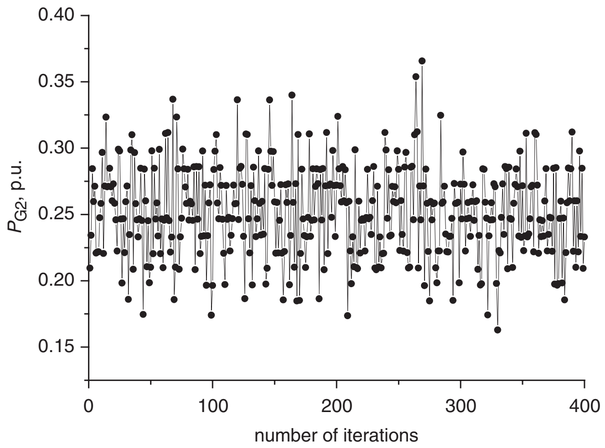

Fig. 7 *Monte Carlo analysis ($W_1 = 0.5$ and $W_2 = 0.5$)*

the results after 400 iterations, the best generation power to satisfy the given performance measure in a probabilistic sense is $P_{G2} = 0.252$ p.u. Similarly, the simulation results with $W_1 = 0.35$ and $W_2 = 0.65$ are shown in Fig. 8, in which case the best generation power to satisfy the given performance measure in a probabilistic sense is $P_{G2} = 0.230$ p.u.

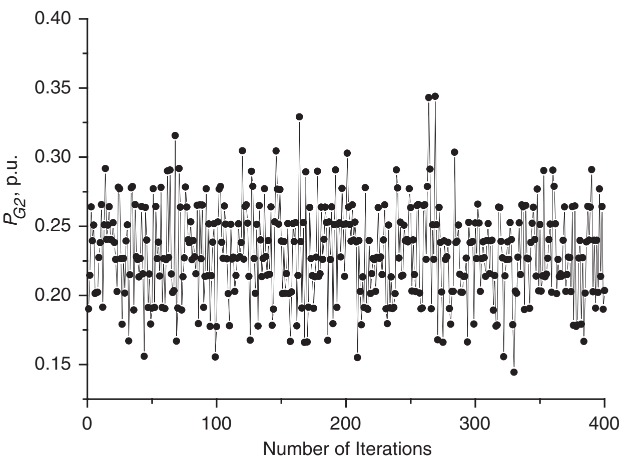

Fig. 8 *Monte Carlo analysis ($W_1 = 0.65$ and $W_2 = 0.35$)*

The Monte Carlo method can give a reasonable solution under a random environment. However, determining the weighting factors is very difficult in the Monte Carlo method. In the learning automata, the value of the threshold for each objective can be easily selected by considering the optimum value of each objective. It rates the action as favourable if the value of the objective function corresponding to the response is less than the threshold value and it increases the probability of selecting that action among all the possible actions. The processes are repeatedly performed to learn which action is the best in a random environment and the learning automata converges to the best action as the learning automata learn more about which action causes good response with the highest probability according to the repeated procedures.

6 Conclusions

The concept of learning automata has applied to a multi-objective optimisation problem to obtain the best trade-off between the conflicting objectives of economy and stability in a power system. The generator power is considered as the action for the learning automaton. The problem of solving for a best compromise solution that simultaneously satisfies both the economic operation and stable operation criteria was reduced to that of determining the best action to satisfy the given criteria in a probabilistic sense. The procedure of applying the learning automata to a multi–objective optimisation problem was proposed. We have demonstrated that learning automata can be applied effectively to solve multi-objective power system problems that are under a random environment and are difficult to solve analytically.

7 References

1 Tsetlin, M.L.: 'On the behaviour of finite automata in random media', *Avtom. Telemekh.*, 1961, **22**, pp. 1345–1354
2 Cover, T.M., and Hellman, M.E.: 'The two-armed bandit problem with time invariant finite memory', *IEEE Trans. Inf. Theory*, 1970, **16**, pp. 185–195
3 Aso, H., and Kimura, M.: 'The structures of automata to adapt to an unknown environment', *IEEE Trans. Syst., Man Cybern.*, 1976, **6**, pp. 494–504
4 Viswanathan, R., and Narendra, K.S.: 'A note on the linear reinforcement scheme for variable structure stochastic automata', *IEEE Trans. Syst. Man Cybern.*, 1972, **2**, pp. 292–294
5 Narendra, K.S., and Thathachar, M.A.L.: 'Learning Automata', (Prentice Hall, New Jersey, 1989)
6 Thathachar, M.A.L., and Phansalkar, V.V.: 'Convergence of teams and hierarchies of learning automata in connectionist systems', *IEEE Trans. Syst. Man Cybern.*, 1995, **25**, pp. 1459–1469
7 Najim, K., and Poznyak, A.S.: 'Multimodal searching technique based on learning automata with continuous input and changing number of actions', *IEEE Trans. Syst. Man Cybern.*, 1996, **26**, pp. 666–673
8 Howell, M.N., and Gordon, T.J.: 'On the genetic adaptation of stochastic learning automata'. Proc. Congress on Evolutionary Computation, 2000, Vol. **2**, pp. 1079–1086
9 Agache, M., and Oommen, B.J.: 'Generalized pursuit learning schemes: New families of continuous and discretized learning automata', *IEEE Trans. Syst. Man Cybern.*, 2002, **32**, pp. 738–749
10 Papadimitriou, G.I., Sklira, M., and Pomportsis, A.S.: 'A new class of ε-optimal learning automata', *IEEE Trans. Syst. Man Cybern.*, 2004, **34**, (1), pp. 246–254
11 Jung, T.H., Alvarado, F.L., and Demarco, C.L.: 'Optimal reactive power dispatch based on a multiobjective optimization technique'. Proc. IFAC Symp. on Power Systems and Power Plant Control, Seoul, Korea, 1989, pp. 106–111
12 Wadhwa, C.L., and Jain, N.K.: 'Multiple objective optimal load flow: a new perspective', *IEE Proc.*, 1990, **137**, (1), pp. 13–18
13 Garduno-Ramirez, R., and Lee, K.Y.: 'Multiobjective Optimal Power Plant Operation Through Coordinate Control with Pressure Set Point Scheduling', *IEEE Transactions on Energy Convers.*, 2001, **16**, (2), pp. 115–122
14 Garduno-Ramirez, R., and Lee, K.Y.: 'A Multiobjective-Optimal Neuro-Fuzzy Extension to Power Plant Coordinated Control', *Trans. Inst. Meas. Control*, 2002, **24**, (2), pp. 159–192
15 Lee, B.H. LEE, and K.Y.:: 'Dynamic and static voltage stability enhancement of power systems', *IEEE Trans. on Power Syst.*, 1993, **8**, pp. 231–238
16 Chong, E.K.P., and Zak, S.H.: 'An introduction to optimization', (Wiley, New York, 1996)