Chapter 11

Evolutionary Computation Methods for Fuzzy Decision Making on Load Dispatch Problems

Guoli Zhang [1], Guangquan Zhang [2], Ya Gao [2], and Jie Lu [2]

[1] *School of Mathematics and Physics, North China Electric Power University, Baoding 071003, China*

E-mail: zhangguoli@ncepu.edu.cn

[2] *Faculty of Engineering and Information Technology, University of Technology, Sydney, PO Box 123, Broadway NSW 2007, Australia*

{*zhangg, yagao,jielu*}*@it.uts.edu.au*

This chapter introduces basic concepts relating to a day-ahead market in a power system. A load dispatch model considers a ramp rate and valve-point-loading effects. An environment/economic load dispatch model is presented to handle uncertainty factors. The model provides theoretical foundations for the research on operations and decision making in the electric power market. To solve load dispatch problems from day-ahead markets in power systems, a hybrid evolutionary computation method with a quasi-simplex technique, a weight point method for multi-objective programming, and a fuzzy-number-ranking-based optimization method for fuzzy multi-objective non-linear programming are developed.

11.1 Models for Day-ahead Markets

The load dispatch in a spot market is one of the kernel problems in an electric power market. It not only relates to the benefits of every participant in the market, but is also a key issue to assure safety, reliability of the power system and order operation of the electric power market. Although a lot of achievements have been obtained, there are still many problems to be solved for the power market operation. This section introduces the basic concepts of electric power markets and builds up two load dispatch models for a day-ahead market.

D. Ruan, *Computational Intelligence in Complex Decision Systems*, Atlantis Computational Intelligence Systems 2, DOI 10.2991/978-94-91216-29-9_11, © 2010 Atlantis Press/World Scientific

11.1.1 *Introduction*

In a traditional generation electricity plan, the electricity price is determined by the generated electricity cost. In general, one price corresponds to one unit, and the price is fixed for a long time. Under an electric power market environment, since the previous pricing mechanism is unreasonable to represent fair trading and reflect the market status of supply and demand, many new price methods have been proposed. There are two typical electricity prices widely used in electric power markets. One is the so-called uniform market clearing price (MCP) or system marginal price (SMP) which can be obtained by the highest bidding of the unit committed. The other is pay-as-bid price (PAB). SMP represents the fairness of merchandise price, i.e., the same quality electric energy should have the same electricity price in the same power grid. It represents the fairness of market competition by using PAB to compute the fee of purchasing electricity and dispatching load, which is consistent with the purpose of an opening electric generation market. Both SMP and PAB are reasonable, but they still have insufficiencies. A reasonable price should consider the fairness of both the merchandise pricing and the market competition. Therefore we propose the principle of the market clearing price determined by SMP and load dispatch calculated by PAB. This mechanism combines the merits of SMP and PAB, solves simultaneous fairness of the price of merchant and market competition, and is feasible and simple. This mechanism encourages generation enterprise to uncover the inner potential, decrease generation cost, increase competition capability, realize lower bid, and finally benefit consumers.

The basic structure of a power market [1, 19, 21] consists of power exchange (PX) and independent system operator (ISO). In this market structure, PX takes charge of the spot trading in the day-ahead market, with the main task to solve the dynamic economic load dispatch problem. In the practical process of electric energy exchange, ISO takes responsibility for both network security and the auxiliary service. In other words, the congestion management and spinning reserve are controlled by ISO. In this study, we use this market structure and build two load dispatch models.

11.1.2 *A Load Dispatch Model for a Day-ahead Market*

Economic dispatch (ED) is very important in power systems, with the basic objective of scheduling the committed generating unit outputs to meet the load demand at minimum operating cost, while satisfying all units and system constraints. Different models and techniques have been proposed in the literature [3, 5, 6].

A conventional economic dispatch (CED) considers only the output power limits and the balance between the supply and the demand. If ramp rate constraints are included, the model becomes the dynamic economic dispatch (DED). Great efforts have been devoted to economic dispatch problems and various models have been proposed [1, 4, 6, 8, 15, 13, 14, 17, 19, 22, 23, 24]. In general, since the CED model does not take into account the ramp rate constraints, its solution may not be real optimal. In order to assure the optimization of solutions, the load dispatch model must consider the ramp rate limit. Therefore, the DED model is needed. Due to the inclusiveness of ramp rate constraints, the number of decision variables involved in the problem will increase dramatically compared with the corresponding CED problem. The sharp increase of the number of variables implies the increase of searching dimensions, which furthermore results in the difficulty of solving the problems of DED. On the other hand, CED problems usually formulate the objective function as smooth, which are solved by using equal λ rules [22], which, however, are not always adequate for real ED or DED problems. A non-smooth function sometimes has to be used to account for special factors, such as the voltage rippling [23, 24]. A more accurate ED model that can account for special cost factors leading to a non-smooth objective, and also including the ramp rate constraints would be highly desired. In addition, there are different constructions and operation modes in power markets, such as the England and Wales power market, California power market, Norway power market, Chile power market, and the Australia and New Zealand power markets [1, 3, 6, 7, 19, 21]. Among some of these power markets, a power exchange-independent system operator model (PX-ISO model) has been adopted in the Chilean power market [19], and the California power market [1, 21]. In this model, PX administrates the day-ahead market and the ED is the major task for PX. ISO will verify the dispatch schedule against a set of criteria, including network security, transmission congestion and spinning reserve. Hence, constraints on the spinning reservation can be ignored in the DED model. Based on the analysis above, a dynamic economic load dispatch (DELD) model for a PX-ISO power market can be described as follows:

$$\text{(M1)} \qquad \begin{cases} \min f(P_j(t)) = \sum_{t=1}^{T} \sum_{j=1}^{N} F(P_j(t)) \\ \sum_{j=1}^{N} P_j(t) = P_D(t) + P_L(t) \\ P_{j\,\min} \leqslant P_j(t) \leqslant P_{j\,\max} \\ -D_j \leqslant P_j(t) - P_j(t-1) \leqslant R_j \end{cases} \qquad (1)$$

where $P_j(t)$ is the output power of the j-th unit during the t-th time interval, T is the number of time intervals per dispatch cycle, N represents the number of committed units, and $F(P_j(t))$ is the generation cost function and can be formulated as

$$F(P_j(t)) = a_{0j} + a_{1j}P_j(t) + a_{2j}P_j^2(t) + \left|d_j \sin[e_j(P_{j\min} - P_j(t))]\right| \tag{2}$$

where a_{0j}, a_{1j}, a_{2j} are constants, $\left|d_j \sin[e_j(P_{j\min} - P_j(t))]\right|$ represents the rippling effects caused by the steam admission valve openings, d_j and e_j are coefficients of the j-th unit, $P_D(t)$ and $P_L(t)$ are the load demand and network loss in the t-th time interval respectively, $P_{j\min}$ and $P_{j\max}$ are the minimum and maximum output power of the j-th unit respectively, D_j and R_j are the maximum downwards and the maximum upwards ramp rate of the j-th unit respectively.

The objective function in the above model (M1) can also be the expense of purchasing electricity.

The model (M1) describes a non-linear programming problem with multiple local optimal points. The prospective algorithms for solving this model must have a stronger global searching capability. The new algorithm to solve this problem will be given later in this chapter.

11.1.3 *An Uncertain Environment/Economic Load Dispatch Model*

A conventional economic dispatch problem is mainly concerned with the minimization of operating costs or purchasing electricity fee, subject to the diverse constraints in terms of units and systems. However, an environmental pollution problem caused by generation has been presented in recent years. A variety of feasible strategies [1, 17, 19] have been proposed to reduce atmospheric emissions. These include installation of pollutant cleaning equipment, switching to low emission fuels, replacing the aged fuel-burners and generator units, and emission dispatching. Petrowski referred the first three options as the long-term ones, and the emission dispatching option as an attractive short-term alternative [17]. In fact, the first three options should be determined by the generation companies, not by the regulatory authorities, especially in the circumstances of the electric power market. The desired long-term target is to reduce the emission of harmful gases. In other words, the emission of harmful gases required to generate electricity should be curtailed in accordance with laws and regulations. Therefore, the environmental/ economic load dispatch problem considering emission of harmful gases is a kernel issue in electric power markets.

Some researchers pointed out that the environmental/economic load dispatch problem is to simultaneously minimize two conflicting objective functions, i.e., minimization of fuel cost and emission, while satisfying load demand and system constraints. The emission of thermal units mainly includes SO_2, NO_x and CO_2, which are not distinguished in this chapter for reasons of simplicity. In general, these harmful gases mentioned above are all functions of output power P_j, and their emission (*ton/h*), written as $E(P_j)$, can be described as

$$E(P_j) = \alpha_j + \beta_j P_j + \gamma_j P_j^2 \tag{3}$$

where α_j, β_j and γ_j are coefficients of the j-th generator emission characteristics.

In a typical environmental/economic load dispatch model, the coefficients of both cost function and emission function are constants, and generally can be obtained by experiments. However, there exist many factors which affect these coefficients, such as: experiment errors, different operation situations, the quality of coal and the aging of facilities. Therefore, it is not precise to describe these coefficients as fixed values. Aimed at characterizing the cost and emission more precisely, we present these coefficients described by fuzzy numbers. A new load dispatch model with uncertainty, called the fuzzy dynamic environmental/economic load dispatch model (FDEELD), is built as follows:

$$\text{(M2)} \quad \begin{cases} \min f = \sum\limits_{t=1}^{T}\sum\limits_{j=1}^{N} F(P_j(t)) = \sum\limits_{t=1}^{T}\sum\limits_{j=1}^{N} (\widetilde{a}_j + \widetilde{b}_j P_j(t) + \widetilde{c}_j P_j^2(t)) \\ \min e = \sum\limits_{t=1}^{T}\sum\limits_{j=1}^{N} (\widetilde{\alpha}_j + \widetilde{\beta}_j P_j(t) + \widetilde{\gamma}_j P_j^2(t)) \\ \sum\limits_{j=1}^{N} \mathrm{P}_j(t) = P_D(t) + P_L(\mathrm{t}) \\ \mathrm{P}_{j\,\min} \leqslant \mathrm{P}_j(t) \leqslant \mathrm{P}_{j\,\max} \\ -\mathrm{D}_j \leqslant \mathrm{P}_j(t) - P_j(t-1) \leqslant R_j \end{cases} \tag{4}$$

where $\widetilde{a}_j$, $\widetilde{b}_j$, $\widetilde{c}_j$ are fuzzy cost coefficients of the j-th unit, e is an emission function, $\widetilde{\alpha}_j$, $\widetilde{\beta}_j$, $\widetilde{\gamma}_j$ are fuzzy emission coefficients of the j-th unit. The meanings of the other symbols are the same as the symbols in the model (M1).

The model (M2) describes a fuzzy multi-objective non-linear programming problem, from which it is very hard to obtain an optimal solution. In Section11.2, we will propose a weighted ideal point method, a hybrid evolutionary method and a fuzzy number ranking method to solve FDEELD.

11.2 Evolutionary Computation Methods and Fuzzy Decision Making

The model (M1) built in the above section is a non-linear programming problem with multiple local optimal points, and (M2) is a fuzzy multi-objective non-linear programming problem. These optimization problems are hard to solve; we will develop some new algorithms to solve these problems.

11.2.1 *Evolutionary Computation*

Conventional optimization methods suffer from local optimality problems and some of them require a function with good characteristics, such as differentiability, continuity, which, to a certain extent, limit their application. In recent years, stochastic optimization techniques, such as simulated annealing (SA), genetic algorithms (GA), and evolutionary algorithms (EA), have drawn many researchers' attention because the stochastic optimization techniques are capable of finding the near global optimal solutions without putting restrictions on the characteristics of the objective functions, although they require significant computing burdens and generally take a fairly long time to reach a solution. A great amount of effort has been devoted to improving these methods and some of them have been successfully used in a variety of real world problems [17, 26].

GA was initially introduced by John Holland in the seventies as a special technique for function optimization [9]. Hereafter, we refer to it as the classical GA (CGA). A typical CGA has three phases, i.e., initialization, evaluation and genetic operation, which consist of reproduction, crossover and mutation. The performance of CGA precedes the traditional optimization methods in aspects of global search and robustness on handling an arbitrary non-linear function. However, it suffers from premature convergence problems and usually consumes enormous computing time.

In the CGA, the ability of local search mainly relies on the reproduction and crossover operations, which can be referred to as exploitation operations, while the capability of global search is assured by the mutation operation, which can be regarded as the exploration operation. Generally speaking, the velocity of local search increases when the probability of crossover increases. Similarly, the level of capability of global search will increase when the probability of mutation increases. Since the sum of probabilities of all the generic operations must be the unity, the mutation probability has to be reduced to increase the crossover probability for a reasonable level of capability of local search. This contributes to the fact that the probability of mutation in CGA is very low, with a range of 0.1-5%. On the other hand, to achieve a satisfactory level of capability of global search, the probabilities of reproduction and crossover have to be decreased to increase the mutation probability. This will weaken the capability of local search dramatically, slow down the convergence rate and make the global search ability unachievable eventually. In the process of balancing exploration and exploitation based on reproduction/crossover and mutation operations for a fixed population, it is hardly possible to achieve a win-win situation for both sides simultaneously. Therefore, how to create a balance between exploration and exploitation in

GA-type algorithms has long been a challenge and retained its attractiveness to many researchers [10, 17].

We present a new method to enhance the capability of global search by increasing the probability of mutation operation while assuring a satisfactory level of capability of local search by employing the idea of simplex method, the so called quasi-simplex technique: a new hybrid real-coded genetic algorithm with quasi-simplex technique (HRGAQT) is used. HRGAQT has the following aims: (1) we assure the capability of global search by increasing the probability of mutation; (2) mutation is implemented by using an effective real-value mutation operator instead of traditional binary mutation; and (3) we enhance the capability of local search by introducing the so-called quasi-simplex techniques into the CGA since the capability of local search will be significantly weakened by the probability of reproduction/crossover decrease as a result of increasing the probability of mutation. In each iteration, HRGAQT first divides the population into a number of sub-populations and each sub-population is treated as a classical simplex. Then for every simplex, HRGAQT applies four operations in parallel to produce offspring. The first operation is the quasi-simplex evolution in which two prospective individuals will be chosen as the offspring. The other three operations are reproduction, crossover and mutation respectively, which are very similar to the traditional genetic operation, except that the probability of mutation is fairly high. All four operations together will produce a new sub-population with the same size as the corresponding parent sub-group. The new generation is the collection of all the newly produced sub-groups. In short, HRGAQT maintains the diversity of a population to enhance global search capability eventually because a higher diversity of population leads to a higher level of capability to explore the search space, while the local search is mainly implemented by the quasi-simplex technique and reproduction including the elitist strategy and crossover operations.

11.2.1.1 *Function optimization and quasi-simplex technique*

We consider the global minimization problem described by Yao and Liu [25] for the purpose of development of new search algorithms. According to Yao and Liu, the problem can be formalized as a pair of real valued vectors (s, f), where $S \subseteq \mathbb{R}^n$ is a bounded set on $\mathbb{R}^n$ and $f : S \to R$ is an n-dimensional real-valued function. f needs not be continuous but must be bounded. The problem is to find a point where $f(x_{\min})$ is a global minimum on S. More specially, it is required to find an $x_{\min} \in S$ such that

$$\forall x \in S, \quad f(x_{\min}) \leqslant f(x) \tag{5}$$

On solving the above optimization problem by genetic algorithms, an effective method, which can speed up the local convergence rate, is to combine the CGA with conventional optimization methods. Since it has been highly recognized that GA has no special request on the characteristics of the objective functions, the conventional optimization methods that go with GA should not require that the objective functions have special characteristics. In this light, Simplex method is promising because it demands less function characteristics. Therefore, we choose to combine the conventional GA with simplex technique to form a hybrid generic algorithm in which a real-value scheme and a dynamic sub-grouping are used. To understand the HRGAQT algorithm, we briefly introduce basic ideas of the simplex technique. Simplex is a type of direct search method, which is a widely accepted search technique. A simplex in an n-dimensional space is defined by a convex polyhedron consisting of $n+1$ vertices, which are not in the same hyper-plane. Assuming there are $n+1$ individuals, denoted by x^i, with function values denoted as f_i, $i = 1, 2, \ldots, n+1$, the worst and the best points in terms of function values are denoted by x^H and x^B, respectively, and can be determined by

$$f(x^H) = f_H = \max_i f_i, \quad i = 1, 2, \ldots, n+1 \tag{6}$$

$$f(x^B) = f_B = \min_i f_i, \quad i = 1, 2, \ldots, n+1 \tag{7}$$

where f_H and f_B denote the worst and the best function values, respectively.

To determine a better new point than the worst point x^H, the centroid x^C of the polyhedron are all the points but the worst one needs to be calculated by

$$x^C = \frac{\left(\left(\sum_{i=1}^{n+1} x^i\right) - x^H\right)}{n} \tag{8}$$

A better point predicted by simplex techniques lies on the line starting from the worst point, towards the centroid, which can be referred to as the worst-opposite direction. The actual location can be determined by the following formula:

$$x = x^C + \alpha(x^C - x^H) \tag{9}$$

where α is a constant and can be a different value for different points lying on the worst-opposite direction, such as the reflection point, expansion points, and the compression points. The actual value ranges of α for different points are shown in Table 11.1.

Conventional simplex techniques mainly consist of four operations, i.e., reflection, expansion, compression, and contraction. The simplex algorithm produces a new simplex by either replacing the worst point by a better point produced using the simplex technique or contracting current simplex towards the best point in each iteration step. The process will

Table 11.1 Points obtained using the simplex techniques with different α

$x = x^C + \alpha(x^C - x^H)$	$\alpha = 1$	reflection point	Reflection point of x^H respect to x^C
	$\alpha > 1$	Expansion point	A point farther than the reflection point from x^C
	$0 < \alpha < 1$	Compression point	Points between x^C and reflection point
	$-1 < \alpha < 0$	Compression point	Points between x^H and x^C

be continuous until the termination criterion is satisfied. The crucial idea of the classical simplex techniques is to track the local optimal following the worst-opposite direction of each simplex, which can be regarded as guidance in the search landscape. Therefore, the simplex algorithm has a higher level of ability of local search.

11.2.1.2 *Hybrid real-coded GA with Quasi-Simplex techniques*

HRGAQT is established by combining a technique evolved from the traditional simplex technique, which is referred to as a quasi-simplex technique with the CGA. In doing so, HRGAQT can achieve a substantially high level of global exploration by increasing the probability of mutation, while its capability of local exploitation can also be reasonably high by using both reproduction/crossover and quasi-simplex techniques.

The process of HRGAQT can be described as follows: First, HRGAQT initializes a random-generated population with μ individuals (real-coded chromosomes) and each individual has n components. The population starts to evolve. At the beginning of each iteration, the generation is divided into a number of sub-populations (or sub-groups) with each sub-group having n+1 individuals. Each sub-group will then evolve into a new sub-population of the same size by four operations in parallel, which are quasi-simplex operation, reproduction, crossover and mutation. The quasi-simplex operation (QS) will generate two new individuals, and the reproduction will retain the best individual by applying the elitist strategy and also produce some individuals based on the probability of reproduction (R). The crossover operation will also generate a number of pairs of individuals according to the probability of crossover (C) and the left-over individuals will be produced by mutation (M). At the end of each evolution iteration, all the new individuals from the sub-populations will merge together and evolution enters new generation. If the termination criterion is not met, evolution starts a new iteration. This process continues until the termination criterion is satisfied. The best individuals of population in the final generation will be taken as the optimal solutions.

HRGAQT has a number of outstanding features which enable both local exploitation and global exploration. HRGAQT adopts the dynamic sub-grouping idea to ensure each simplex consists of reasonably correlated individuals in the entire evolution process to enhance the convergence rate. HRGAQT implements population partition different to the strategies proposed in the literature by two methods. One is to take into account the dimension of individuals on deciding the number of sub-groups. HRGAQT divides a population into a number of sub-groups with each sub-group consisting of $n+1$ individuals to ensure the search validity and efficiency in terms of computing times. The detailed discussion about the size of a sub-population and the number of sub-populations to be used will be presented in another paper. The other method is to make a partition for each iteration. Although the computation time for each iteration may increase due to the partition process, the enhancement in the convergence rate could decrease the number of iterations needed.

Secondly, HRGAQT employs the quasi-simplex technique with ancillary reproduction and crossover operation to assure the local exploitation. The quasi-simplex technique absorbs the idea of classical simplex techniques to perform a guided search. It produces four prospective individuals using the reflection, expansion and compression operations along with the worst-opposite direction. The quasi-simplex technique also expands the conventional simplex technique by looking at the prospective individuals lying on a line starting from the centroid towards the best point of the simplex. We refer to this direction as the best-forward direction, in contrast with the worst-opposite direction. Three prospective individuals x^e, x^m and x^n will be produced along the best-forward direction by the expansion and compression operations using the following formula:

$$x = x^B + \beta(x^B - x^D) \tag{10}$$

where x^D denotes the centroid of the remaining points except for the best point x^B and can be calculated by

$$x^D = \left(\left(\sum_{i=1}^{n+1} x^i \right) - x^B \right) \Big/ n \tag{11}$$

The points x^e, x^m and x^n can be determined by the value of β in (10) and the range of β is shown in Table 11.2.

To avoid a situation in which too many individuals are similar so that the diversity of the population decreases dramatically, HRGAQT selects the best one from the two prospective individual groups along the worst-opposite and the best-forward directions to produce two new individuals as a part of offspring.

11.2.1.3 *A new mutation operator*

To guarantee the local search effect, GA usually uses a very small mutation probability. A typical mutation probability ranges from 0.001 to 0.05. The mutation operators have two

Table 11.2 Range of β in (10) for x^e, x^m and x^n

Formula	Range of β	Calculated point
$x = x^B + \beta(x^B - x^D)$	$\beta > 1$	x^e
	$\beta = 1$	x^m
	$0 < \beta < 1$	x^n

kinds in real-coded GA: one is to generate a new random real number within the domain and the other is to add a new random real number to the original one. Both of these two operators lack support from the principles of biological natural mutation processes. In a process of biological evolution, a gene often changes dramatically after it is mutated. In real-coded GA, decimal digits are used to represent genes. According to the principles of a natural biological mutation, these digits should also change significantly after a mutation operation. In other words, they should become bigger when they are small enough (< 5), or become smaller when they are big enough ($\geqslant 5$). Based on this idea, we propose a new real-coded mutation operator, which is described as follows. Suppose x_{ij}, $i = 1,2,\ldots,\mu$, $j = 1,2,\ldots,n$, represents the j-th component in the i-th individual, where μ is the size of a population and n is the dimension of each individual. In a real-coded scheme, x_{ij} can be expressed as a sequence of decimal numbers including the decimal point:

$$x_{ij} = d_{ij}^{w_1} d_{ij}^{w_2} \cdots d_{ij}^{w_p} \bullet d_{ij}^{f_1} d_{ij}^{f_2} \cdots d_{ij}^{f_q} \tag{12}$$

where superscript w and f denote the integer part and the fractional part respectively, and p and q are constants representing the number of digits in the integer part and the fractional part for a given x_{ij}, respectively. In application, p is determined by the maximum value that this sequence can represent, q is determined by the precision required by the problems and its maximum value will be determined by the hardware used in computing. If the digits in the sequence are randomly selected to undertake a mutation, the new sequence after the mutation can be represented as:

$$\overline{x_{ij}} = \overline{d_{ij}^{w_1}}\, \overline{d_{ij}^{w_2}} \cdots \overline{d_{ij}^{w_p}} \bullet \overline{d_{ij}^{f_1}}\, \overline{d_{ij}^{f_2}} \cdots \overline{d_{ij}^{f_q}} \tag{13}$$

where each decimal digit is determined by:

$$\overline{d_{ij}^{r}} = 9 - d_{ij}^{r} \text{ if } d_{ij} \text{ is selected} \tag{14}$$

or

$$\overline{d_{ij}^{r}} = d_{ij}^{r} \text{ if } d_{ij} \text{ is not selected} \tag{15}$$

where $r = w_1, w_2, \ldots, w_p,\ f_1, f_2, \ldots, f_q$.

11.2.1.4 *HRGAQT algorithm procedure*

The HRGAQT algorithm can be outlined in the following steps:

Step 1 Initialize a random population X with size $\mu = K(n+1)$.

Step 2 Divide the population X into K sub-populations with each sub-group consisting of $n+1$ individuals.

Step 2.1 Select the best individual x from the population X.

Step 2.2 Select n individuals which are most close to x in terms of their Euclid distances.

Step 2.3 Combine the individuals obtained from steps 2.1 and 2.2 to form a sub-population S.

Step 2.4 Remove S from the original population.

Step 2.5 Repeat Steps 2.1 – 2.4 for the sub-population until no individuals are left.

Step 3 Each sub-population evolves into a new group.

Step 3.1 Produce two new individuals using quasi-simplex techniques.

Step 3.2 Implement elitist strategy, i.e., reserve the best one in the sub-population to be a part of offspring.

Step 3.3 Produce new individuals by reproducing by linear ranking. The reproduction probability of the i-th individual x^i, in the target sub-group (sorted by descending the fitness) calculated by the following formula

$$P_i = \frac{1}{n+1}\left(\eta - 2(\eta-1)\cdot\frac{\text{rank}(x_i)-1}{n}\right) \tag{16}$$

where $\eta > 1$, which can be determined by the desired probability of the best individual.

Step 3.4 Crossover operation is processed as follows:

Select $[{(n-2)P_C}/{2}]$ pairs of parents randomly, where [] is an operator producing the maximum integer which is less than or equal to the operand. For every pair of the selected parent,

$$x^i = \left(x_1^i, x_2^i, \ldots, x_{m1}^i, \ldots, x_{m2}^i, \ldots, x_n^i\right) \tag{17}$$

$$x^j = \left(x_1^j, x_2^j, \ldots, x_{m1}^j, \ldots, x_{m2}^j, \ldots, x_n^j\right) \tag{18}$$

where the superscripts i and j denote the i-th and the j-th individual in the population respectively. The subscript $m1$ and $m2$ are two random numbers between

1 and n. The two new individuals will be:

$$x^i_{\text{new}} = \left(x^i_1, x^i_2, \ldots, x^j_{m1}, \ldots, x^j_{m2}, \ldots, x^i_n\right) \quad (19)$$

$$x^j_{\text{new}} = \left(x^j_1, x^j_2, \ldots, x^i_{m1}, \ldots, x^i_{m2}, \ldots, x^j_n\right) \quad (20)$$

Step 3.5 The remaining individuals will participate in the mutation operation. For each individual, a new individual will be produced by (12)–(15).

11.2.2 *A Fuzzy Multi-object Non-linear Optimization Method*

11.2.2.1 *A weight idea point method of multi-objective optimization problems*

Both weighting and reference point methods are all powerful methods to achieve Pareto optimal solutions for multi-objective non-linear programming problems. Strictly speaking, the weight method only represents the relative importance of goal values from an objective rather than from different objectives. It is hard to know the magnitude of effect of the set of weights to each objective function value. The reference point method is a relatively practical interactive approach to multi-objective optimization problems. It introduces the concept of a reference point suggested by decision makers and presents some desired values of the objective functions. It is very hard to determine weightings and reference points in applications, and the interactive approach increases computing burden heavily. This section proposes a new weighting ideal point method (WIPM), which doesn't require any interaction, and can predict the magnitude of effect of the set of any weights to each objective function value.

To describe the proposed WIPM method, we write a general multi-objective non-linear programming problem as:

$$\min_{x \in S} f(x) = (f_1(x), f_2(x), \ldots, f_k(x)) \quad (21)$$

where $f_1(x), \ldots, f_k(x)$ are k distinct objective functions and S is the constrained set defined by

$$S = \{x \in \mathbb{R}^n \mid g_j(x) \leqslant 0,\ j = 1, \ldots, m\} \quad (22)$$

In this section, we propose a weighted ideal method (WIPM) as follows: let

$$g(x) = w_1 \left(\frac{f_1 - f_1^{\min}}{f_1^{\min}}\right)^2 + \cdots + w_k \left(\frac{f_k - f_k^{\min}}{f_k^{\min}}\right)^2 \quad (23)$$

where

$$f_i^{\min} = \min_{x \in S} f_i(x), \quad f_i^{\min} \neq 0, \quad i = 1, 2, \ldots, k.$$

$f^{\min} = (f_1^{\min}, \ldots, f_k^{\min})$ is a so-called ideal or utopia point, $w = (w_1, \ldots, w_k) > 0$, $\sum_{i=1}^{k} w_i = 1$ is a weight vector.

To get the Pareto optimal solution of the problem (21), it can be transferred to solve the single objective optimization problem below

$$\min_{x \in S} g(x) \tag{24}$$

Since the values of different objective functions in (21) can be very different, it is hard to know the magnitude of the effect of the set of weights to each objective function value. In the model (24), all objectives are converted into the same magnitude by the formula

$$\frac{f_i - f_i^{\min}}{f_i^{\min}}.$$

We can therefore predict the effect quantity of the set of weights to each objective function value. For example, if $w_1 = 2w_2$, then

$$\frac{f_2^* - f_2^{\min}}{f_2^{\min}} \approx 2\frac{f_1^* - f_1^{\min}}{f_1^{\min}},$$

where $f_i^* = f_i(x^*)$, $i = 1, 2$, x^* is the optimal solution of (23). In other words, the weights given in WIPM can reflect the trade-off rate information among the objective functions.

11.2.2.2 *A weight idea point method of fuzzy multi-objective optimization problems*

A problem becomes a fuzzy multi-objective non-linear programming problem if the objective function f_i includes uncertainty represented by fuzzy numbers in the multi-objective non-linear programming problem (21). We will give a solution based on the weight idea point method and the fuzzy number ranking.

When the non-linear objective functions are fuzzy functions, we also use (23) to convert (21) into a corresponding single objective fuzzy optimization problem. Now we need to solve a single objective fuzzy programming problem (SOFPP). One of the methods to solve fuzzy optimization problems is the maximum satisfaction factor method [11]. Another one is to convert a fuzzy optimization problem into several classical optimization problems. In this section we do not use the above methods, but directly apply HRGAQT to search for optimum solutions. We then compare the function values of different solutions by the method of ranking fuzzy numbers.

Different methods for ranking fuzzy numbers have been proposed [5, 10, 15, 19]. The definition below comes from Lee and Li [15].

Definition 11.1. Let $\widetilde{a}$, $\widetilde{b} \in F(R)$ be two fuzzy numbers. The definition of ranking two fuzzy numbers is as follows:

$$\widetilde{a} \leqslant \widetilde{b} \text{ if } m(\widetilde{a}) < m(\widetilde{b}) \tag{25}$$

or

$$m(\widetilde{a}) = m(\widetilde{b}) \text{ and } \sigma(\widetilde{a}) \geqslant \sigma(\widetilde{b}) \tag{26}$$

where mean $m(\widetilde{a})$ and standard deviation $\sigma(\widetilde{a})$ are defined as

$$m(\widetilde{a}) = \frac{\int_{s(\widetilde{a})} x\widetilde{a}(x)dx}{\int_{s(\widetilde{a})} \widetilde{a}(x)dx} \tag{27}$$

$$\sigma(\widetilde{a}) = \left(\frac{\int_{s(\widetilde{a})} x^2\widetilde{a}(x)dx}{\int_{s(\widetilde{a})} \widetilde{a}(x)dx} - \big(m(\widetilde{a})\big)^2 \right)^{\frac{1}{2}} \tag{28}$$

where $s(\widetilde{a}) = \{x \mid \widetilde{a}(x) > 0\}$ is the support of fuzzy number $\widetilde{a}$.
For a triangular fuzzy number $\widetilde{a} = (l,m,n)$,

$$m(\widetilde{a}) = \frac{1}{3}(l+m+n) \tag{29}$$

$$\sigma(\widetilde{a}) = \frac{1}{18}\big(l^2+m^2+n^2-lm-ln-mn\big) \tag{30}$$

The main steps of the weight idea point method for fuzzy multi-objective optimization problems are as follows:
Step1 Convert problem (21) into a single objective optimization by using (23) and (24);
Step2 Solve the single objective optimization (24) by HRGAQT.
Note:

(1) $f^i_{\min}$ can be given by a desired value or determined by solving the corresponding single objective optimization problem by HRGAQT.
(2) In solving the single objective optimization by using HRGAQT, for each individual we compute fuzzy function values according to the fuzzy number operation principle, then directly compare the function values of different solutions by the fuzzy ranking method given above.

Table 11.3 Technical data of units

Unit No.	P_{min}(MW)	P_{max}(MW)	a	b	c	d	e	D	R
1	00	680	550	8.10	0.00028	300	0.035	60	50
2	00	360	309	8.10	0.00056	200	0.042	50	35
3	00	360	307	8.10	0.00056	200	0.042	50	35
4	60	180	240	7.74	0.00324	150	0.063	40	30
5	60	180	240	7.74	0.00324	150	0.063	40	30
6	60	180	240	7.74	0.00324	150	0.063	40	30
7	60	180	240	7.74	0.00324	150	0.063	40	30
8	60	180	240	7.74	0.00324	150	0.063	40	30
9	60	180	240	7.74	0.00324	150	0.063	40	30
10	40	120	126	8.6	0.00284	100	0.084	30	25
11	40	120	126	8.6	0.00284	100	0.084	30	25
12	55	120	126	8.6	0.00284	100	0.084	30	25
13	55	120	126	8.6	0.00284	100	0.084	30	25

11.3 Illustrations on Load Dispatch for Day-ahead Market

11.3.1 *An example of load dispatch in a day-ahead market*

To test the effectiveness of the proposed HRGAQT in solving a DED problem (M1), a typical dynamic dispatch case consisting of 13 committed units and 24 time intervals is chosen. The data of the unit techniques and predicted load demands in each dispatch period are listed in Tables 11.3 and 11.4. Experimental results are as follows.

Because the objective function is a high-dimensional function with multi extremum points, it is unknown where the real optimal solution is. In order to demonstrate the effectiveness of the proposed algorithms, the mean value and standard deviation of total cost corresponding with optimal outputs would be significant and convincing. Table11.3.1 lists the optimal total cost, the mean value and standard deviation of 10 results obtained by the proposed algorithm running independently 10 times. Table 11.6 gives optimal power output of units and total cost corresponding to the best results.

The best result occurred in the 7-th time, and the optimal power output of units and total cost corresponding to the best result is listed in Table 11.3.1. It is obvious that the standard deviation is small and the results are believable.

The experiments show that the proposed method with hybrid real-coded generic algorithms and the quasi-simplex techniques is very effective and the results are convincing.

In Table 11.4, T is time interval, $P_D(t)$ is the load demand.

Table 11.4 Load demands in different time intervals

T	1	2	3	4	5	6	7	8	9	10	11	12
$P_D(t)$	1550	1500	1520	1540	1600	1680	1780	1880	1950	2010	1970	1970
T	13	14	15	16	17	18	19	20	21	22	23	24
$P_D(t)$	1910	1830	1850	1880	1920	2150	2370	2280	2130	1950	1790	1670

Table 11.5 The total cost, the mean value and standard deviation

i-th time	1	2	3	4	5
i-th result	461839	461138	460893	461382	460562
Mean Value	46113.47				
Std Dev	582.5868				

i-th time	6	7	8	9	10
i-th result	461540	459898	461779	461134	461182
Mean Value					
Std Dev					

11.3.2 *An Example of Uncertain Environment/Economic Load Dispatch*

In this section, we solve an uncertain environment/economic load dispatch problem (M2) by using the WIPM, HRGAQT and fuzzy number ranking methods. We convert (M2) into a single objective optimization problem by using WIPM. We then use the Lagrange relaxation method to form a Lagrange function. Finally, we use the HRGAQT to optimize the Lagrange function. In the process of the iteration, the fuzzy number ranking method is used to compare fuzzy function values of different points for the single objective function.

Tables 11.7–11.10 show the test data of the units output, cost function, emission function, and load demand, respectively.

Penalty function h is a high-dimension non-linear function, and therefore it is hard to know where the global minimum point is. In order to demonstrate the effectiveness of the proposed algorithm, the mean and standard deviation of fuzzy fuel cost, fuzzy emission and fuzzy total cost corresponding with the optimal outputs are tested. In addition, in order to compare the magnitude of effect of the set of weights to fuzzy fuel cost and fuzzy emission, we calculate three group weights. Table 11.11 lists the means and standard deviations of fuzzy fuel cost, fuzzy emission and fuzzy total cost. Table 11.12 shows results obtained by the proposed algorithm through 10 independent runs.

Table 11.6 Optimal power output of units and total cost

Time	Unit number												
Interval	1	2	3	4	5	6	7	8	9	10	11	12	13
1	359.04	188.05	185.14	94.88	91.76	96.90	94.75	94.62	93.25	42.41	61.47	55.32	92.40
2	355.22	174.74	150.39	98.53	89.47	99.36	60.03	89.58	99.78	67.83	40.02	55.00	120.04
3	358.60	194.27	171.71	95.95	98.97	95.05	95.05	60.00	96.98	40.44	40.25	55.00	117.72
4	355.86	191.76	170.36	94.75	94.41	93.27	93.41	93.67	92.31	64.67	40.01	63.12	92.40
5	359.07	195.37	209.22	95.14	97.82	98.79	96.65	109.10	99.99	51.47	40.00	55.00	92.40
6	359.92	220.06	219.94	108.86	109.86	108.75	107.60	106.96	109.34	42.50	48.31	55.00	82.91
7	429.37	214.85	224.40	104.71	109.48	108.45	109.67	108.96	111.03	72.50	76.61	55.00	54.99
8	447.57	224.22	224.40	114.39	114.28	117.04	113.02	112.42	120.43	75.81	77.40	55.00	84.02
9	448.80	236.88	236.66	112.20	112.45	121.77	123.68	118.19	113.14	78.15	77.72	77.78	92.60
10	451.33	244.58	239.12	116.05	121.79	126.59	114.95	136.89	120.75	81.82	79.05	84.67	92.41
11	450.08	247.24	234.06	122.68	125.44	111.44	126.06	126.88	120.07	77.66	80.75	55.07	92.58
12	452.43	247.49	227.11	155.96	111.87	111.35	153.38	120.04	114.74	78.89	77.43	55.00	64.31
13	448.18	244.02	233.11	116.56	117.81	111.47	114.24	110.29	113.09	77.39	76.49	55.00	92.34
14	448.77	224.37	224.35	113.11	108.14	114.43	110.06	109.66	115.02	69.20	46.70	55.00	91.20
15	448.80	224.36	224.37	110.91	112.02	113.43	110.03	109.86	110.14	75.72	76.72	55.00	78.65
16	448.80	230.23	234.55	114.92	114.41	114.51	113.52	113.41	124.86	47.12	76.09	55.02	92.56
17	449.96	234.55	238.24	111.35	115.71	110.27	114.63	129.68	115.84	76.26	75.85	55.00	92.65
18	470.15	245.53	251.87	114.17	120.11	113.43	112.95	115.91	118.19	75.88	78.42	75.85	67.55
19	539.01	285.47	288.81	147.11	152.92	147.66	147.38	150.76	152.97	103.90	98.27	93.78	91.98
20	538.64	298.90	287.01	143.60	150.57	159.74	158.98	147.86	159.84	77.40	77.45	92.48	57.54
21	519.91	251.07	244.73	131.80	154.34	152.89	120.22	122.64	158.77	77.40	78.20	57.67	60.35
22	448.23	240.95	234.14	116.17	114.36	113.01	111.22	120.62	132.66	80.32	77.28	71.37	89.68
23	440.31	218.10	224.41	109.87	109.86	109.87	112.50	110.42	109.87	77.43	55.55	55.01	56.80
24	360.31	224.52	224.46	109.87	109.68	109.86	108.57	109.82	109.90	52.85	40.00	55.00	55.15
Total cost		459898											

Table 11.7 Limits of unit output and ramp rate

Unit No.	1	2	3	4	5	6	7
$P_{min}(MW)$	20	20	35	35	130	120	125
$P_{max}(MW)$	125	150	150	210	325	310	315
D_j	40	40	40	50	60	60	60
R_j	30	30	30	40	50	50	50

Table 11.8 Fuzzy coefficients of the cost function

Unit No	a_0	a_1	a_2	b_0	b_1	b_2	c_0	c_1	c_2
1	800.95401	825.72578	846.36892	37.46062	38.53973	39.46468	0.15813	0.16218	0.16559
2	625.96538	645.32513	661.45826	41.32673	42.51721	43.53762	0.12050	0.12359	0.12619
3	1107.49967	1135.89710	1158.61504	38.83637	39.83217	40.62881	0.02651	0.02705	0.02754
4	1168.89357	1198.86520	1222.84250	36.90654	37.85286	38.60992	0.03403	0.03472	0.03534
5	1555.00481	1586.73960	1610.54069	36.58126	37.32782	37.92507	0.02478	0.02521	0.02559
6	1269.74602	1295.65920	1315.09409	38.29901	39.08062	39.70591	0.01653	0.01682	0.01707
7	1466.71867	1496.65170	1519.10148	36.52011	37.26542	37.86167	0.01979	0.02013	0.02043

Table 11.9 Fuzzy coefficients of the emission function

Unit No.	α_0	α_1	α_2	β_0	β_1	β_2	γ_0	γ_1	γ_2
1	15.18178	15.65132	16.04260	0.28456	0.29276	0.29979	0.003822	0.00392	0.00400
2	15.18178	15.65132	16.04260	0.28456	0.29276	0.29979	0.00382	0.00392	0.00400
3	34.69310	35.58267	36.29432	−0.54136	−0.52816	−0.51760	0.00698	0.00712	0.00725
4	34.69310	35.58267	36.29432	−0.54136	−0.52816	−0.51760	0.00698	0.00712	0.00725
5	42.03762	42.89553	43.53896	−0.52138	−0.51116	−0.50298	0.00453	0.00461	0.00468
6	40.92147	41.75660	42.38295	−0.53245	−0.52201	−0.51366	0.00464	0.00472	0.00479
7	40.92147	41.75660	42.38295	−0.53245	−0.52201	−0.51366	0.00464	0.00472	0.00479

In Table 11.10, T represents a time segment, $\mathrm{P_D}(t)$ represents the load demand of the correspondence to the time segment.

In Table 11.11, MFC, MEC, and MTC present the means of the fuel cost, the emission, and the total cost respectively, STDEV-FC, STDEV-EC and STDEV-TC present corresponding standard deviations. As the standard deviations of every result are all significantly small, the results are believable. It can be seen that the fuel cost decreases and the emission increases when the weight of the fuel cost increases.

The model (M2) is a new environmental economic load dispatch model which considers

Table 11.10 Load demands in different time intervals

T	1	2	3	4	5	6	7	8	9	10	11	12
$P_D(t)$	690	670	670	680	730	800	870	840	890	920	950	910
T	13	14	15	16	17	18	19	20	21	22	23	24
$P_D(t)$	890	890	930	970	930	950	1070	1040	950	850	760	730

the uncertainty in coefficients of the fuel cost and emission functions. The weighting ideal point method, hybrid genetic algorithms with quasi-simplex techniques and fuzzy number ranking method are developed and used to solve the optimization problem described in model (M2). Compared with other fuzzy multi-objective programming methods, the proposed method has three main advantages:

(1) To describe the coefficients of the fuel cost and emission functions by fuzzy numbers can more precisely characterize the fuel cost and emission, and can get a more accurate FDEELD model(M2).

(2) The results described by using fuzzy numbers can provide more information than real numbers. The fuzzy minimum tells not only the approximate fuel cost and emission, but also the dispersivity of the minimal fuel cost and emission.

Table 11.11 The comparison of the results obtained for different weights

(w_1, w_2)	(0.3, 0.7)	(0.5, 0.5)	(0.7, 0.3)
	1067359	1061711	1053936
MFC	1092154	1086218	1078110
	1112300	1106213	1097695
	291.4	472.8	57
STDEV-FC	303.8	377.1	58.3
	312.4	615.9	60
	11423.23	11466.7	11600.89
MEC	11993.81	12041.25	12184.12
	12539.55	12596.67	12744.69
	2.2	4.2	1.3
STDEV-EC	2.5	5.3	1.4
	2.7	9.4	1.5
	1078780	1073181	1065537
MTC	1104148	1098320	1090295
	1124838	1118805	1110440
	290.7	468.5	55
STDEV-TC	300.7	540.6	58.2
	310.4	607.9	60.9

(3) The optimum solution obtained is steady and trustworthy, because the solution of minimum dispersivity has been chosen when candidate solutions have approximately the same total cost.

Table 11.12 Optimal power output of units for weights (0.3, 0.7)

Time segment	Unit number 1	2	3	4	5	6	7
1	51.46	53.05	91.02	88.99	136.41	134.07	134.99
2	49.21	49.30	89.04	88.18	131.88	130.08	132.34
3	49.52	49.59	88.67	87.94	132.89	129.43	131.96
4	50.27	50.22	89.99	89.38	133.92	133.12	133.10
5	56.06	61.42	95.36	93.57	141.20	140.42	141.96
6	68.33	71.53	100.77	99.66	153.84	152.07	153.80
7	76.66	82.65	110.36	107.55	165.83	164.45	162.50
8	69.73	78.13	106.59	105.29	160.19	160.29	159.79
9	80.40	86.77	111.39	110.45	168.51	165.29	167.20
10	86.43	90.07	114.66	112.83	174.17	170.07	171.78
11	90.33	92.03	119.42	116.21	181.36	174.44	176.22
12	86.41	87.87	113.15	112.45	171.72	168.67	169.74
13	80.76	85.96	111.19	110.33	169.91	165.58	166.26
14	75.09	88.24	111.12	110.66	169.85	165.97	169.07
15	87.70	89.80	116.03	112.73	176.57	172.51	174.67
16	94.07	98.21	119.83	116.49	183.22	177.82	180.37
17	86.22	90.68	118.52	114.26	175.70	172.11	172.51
18	87.79	93.94	120.20	116.19	181.15	176.30	174.42
19	103.47	111.97	132.23	130.48	200.75	195.36	195.75
20	106.17	104.57	128.55	126.04	194.27	190.44	189.96
21	89.82	92.90	119.69	115.03	181.85	174.92	175.78
22	68.41	81.11	109.17	105.86	163.49	161.03	160.93
23	57.34	66.97	98.25	97.35	147.49	145.48	147.11
24	55.12	62.23	94.79	94.43	141.83	140.30	141.30
Fuel cost		Emission			Total cost		
1066800		11427.6			1078220		
1091570		11998.6			1103570		
1111700		12544.8			1124240		

11.4 Conclusions and Further Research

Aiming at power markets characterized by the PX and ISO structure, and based on the analysis of market competition mode, this chapter proposes a competition mode of "SMP plus PAB", and establishes a load dispatch mode which considers both ramp rate and valve-point-loading effects. As the pollution caused by power generation becomes an increasingly urgent issue, we analyze the uncertainty of power generation cost function and harmful gas emissions function, and develop a fuzzy environment/economic load dispatch mode by fuzzy set theory to minimize power generation costs and harmful gas emissions.

To solve load dispatch problems, we develop an evolutionary method, which combines an evolutionary method and the quasi-simplex technique to improve the convergence rate and global search capability. This method imposes no special requirement on objective functions, and has very wide applications.

To solve fuzzy dynamic environment/economic load dispatch problems, we propose a weight point method, which converts the FDEELD problem into a single objective fuzzy optimization problem. We also present a fuzzy-number-ranking-based optimization method, which can make the most of available information and give more powerful assistance to decision makers. Experiments reveal the effectiveness of this method.

Based on the research in this chapter, we will focus our future research as follows:

Considering the profit for both generating companies and power corporations at the same time, we will use bilevel programming technique, together with Game theory, to develop bidding models.

We will apply multiple objective techniques and fuzzy non-linear programming technique on our current methods.

Multiple leaders multiple followers bilevel game models will be studied for more practical methods.

Bibliography

[1] Abido, M. A. (2003). Environmental/economic power dispatch using multiobjective evolutionary algorithms, IEEE Transactions on Power Systems, 18, pp. 1529–1537.

[2] Albuyeh, F. and Alaywan, Z. (1999). California ISO formation and implementation, IEEE Computer Applications in Power, 12, pp. 30–34.

[3] Alvey, T, Goodwin, D, Ma, X. (1998). Security-constrained Bid-clearing System for the New Zealand Wholesale Electricity Market, IEEE Transactions Power Systems, 13, pp. 340–346.

[4] Attaviriyanupap, P., Kita, H., Tanaka, E. and Hasegawa, J. (2002). A hybrid EP and SQP for dynamic economic dispatch with nonsmooth fuel cost function, IEEE Transactions on Power Systems, 17, pp. 411–416.

[5] Cheng, C. H. (1998). A new approach for ranking fuzzy numbers by distance method, Fuzzy Sets and Systems, 95, pp. 307–317.

[6] Cheung, K. W., Payman, S. and Asteriadis, S. (1999). Functional Requirements of Energy and Ancillary Service Dispatch for the Interim ISO New England Electricity Market, IEEE Engineering Society Winter Meeting, pp. 269–273.

[7] Danai, B., Kim, J. and Cohen, A. I. (2001). Scheduling Energy and Ancillary Service in the New Ontario Electricity Market, IEEE Power Industry Computer Applications Conference. Sydney, pp. 161–165.

[8] Gao F. and Sheble G. B., Stochastic Optimization Techniques for Economic Dispatch with Combined Cycle Units, Probabilistic Methods Applied to Power Systems, 2006. PMAPS

2006. International Conference on, pp.1–8.
[9] Holland, J. H. (1975). Adaptation in Natural and Artificial Systems, Ann Arbor, MI: Univ.
[10] Herrera, F. and Lozano, M. (2000). Gradual distributed real-coded genetic algorithms, IEEE Trans. Evol. Comput., 4, pp. 43–63.
[11] Huang, C. M., Yang, H. T. and Huang, C. L. (1997). Bi-objective power dispatch using fuzzy satisfaction-maximizing decision approach, IEEE Transactions on Power Systems, 12, pp. 1715–1721.
[12] Irisarri, G., Kimball, L. M., Clements. K. A., Bagchi, A. and Davis, P. W. (1998). Economic dispatch with network and ramping constraints via interior point methods, IEEE Transactions on Power Systems, 13, pp. 236–242.
[13] Li Y., McCalley, J.D. and Ryan S. (2007), Risk-based Unit Commitment, Power Engineering Society General Meeting, IEEE, pp. 1–7.
[14] Li D. P., Pahwa A., Das S., and Rodrigo D. (2007), A New Optimal Dispatch Method for the Day-Ahead Electricity Market Using a Multi-objective Evolutionary Approach, Power Symposium, 2007. NAPS '07. 39th North American, pp. 433–439.
[15] Lee, E. S. and Li, R. L. (1988). Comparison of fuzzy numbers based on the probability measure of fuzzy events, Comput. Math. Appl., 15, pp. 887–896.
[16] Morgan, L. F. and Williams, R. D. (1997). Towards more cost saving under stricter ramping rate constraints of dynamic economic dispatch problems-a genetic based approach, Genetic Algorithms In Engineering Systems: Innovations And Applications, pp. 221–225.
[17] Petrowski, A. (1996). A clearing procedure as a niching method for genetic algorithms, in Proc. IEEE Evolutionary Computation, pp. 798–803.
[18] Rughooputh, H. C. S. and King, R. T. F. (2003). Environmental/economic dispatch of thermal units using an elitist multiobjective evolutionary algorithm, Industrial Technology, 2003 IEEE International Conference on, 1, pp. 48–53.
[19] Venkatesh, P., Gnanadass, R. and Padhy, N. P. (2003). Comparison and application of evolutionary programming techniques to combined economic emission dispatch with line flow constraints, IEEE Transactions on Power Systems, 18, pp. 688–697.
[20] Watts, D., Atienza, P. and Rudnick, h. (2002). Application of the Power Exchange-Independent System Operator Model in Chile, Power Engineering Society Summer Meeting, 2002 IEEE, 3, pp. 1392–1396.
[21] Wen, F. and David, A.K. (2002). Coordination of bidding strategies in day-ahead energy and spinning reserve markets, International Journal of Electrical Power & Energy Systems, 24, pp. 251–261.
[22] Wood, A. J. and Wollenberg, B. F. (1996). Power Generation, Operation and Control.
[23] Walters, D.C. and Sheble, G. B. (1993). Genetic algorithm solution of economic dispatch with valve point loading, IEEE Transactions on Power Systems, 8, pp. 1325–1332.
[24] Yang, H. T., Yang, P. C. and Huang, C. L. (1996). Evolutionary programming based economic dispatch for units with non-smooth fuel cost functions, IEEE Transactions on Power Systems, 11, pp. 112–118.
[25] Yao, X. and Liu, Y. (1999). Evolutionary Programming Made Fast, IEEE Trans. Evol. Comput., Vol. 3, pp. 82–102.
[26] Zhang, G. Wu, Y, Remias, M. and Lu, J. (2003). Formulation of fuzzy linear programming problems as four-objective constrained optimization problems, Applied Mathematics and Computation, 139, pp. 383–399.