THE HONG KONG POLYTECHNIC UNIVERSITY

Department of Electrical Engineering

Application of Quantum-Inspired Evolutionary Algorithm in Solving the Unit Commitment Problem

Lau Tat Wa

A thesis submitted in partial fulfilment of the requirements for the degree of Master of Philosophy

August 2010

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

Lau Tat Wa

Abstract

Unit commitment (UC) represents an important optimization problem in power systems. The UC problem is to schedule the on and off statuses of the generating units over a time horizon such that the production cost is minimized and all operation constraints are satisfied. The UC problem is characterized by being large-scale, mixed-integer, complicated and highly constrained. These characteristics make the UC problem one of the most difficult optimization problems in power systems.

Coal-fired thermal plants account for a significant percentage of emissions from generation plants. With the increasing environmental awareness, emission performance becomes an important part to power utilities. With regards to environmental protection, emissions from thermal plants can be included into the UC problem to form a bi-objective UC problem considering both cost and emission objective functions. The two objective functions are conflicting in nature. The bi-objective UC gives a set of Pareto-optimal or compromise solutions, and these solutions facilitate the decision making of power system operators on plant scheduling. Nevertheless, the bi-objective UC problem involves binary and

continuous control variables, non-linearity, bi-objectives and a lot of operating constraints. It is highly difficult to handle the bi-objective UC.

In the literature, a lot of optimization techniques have been proposed to solve the UC problem. These techniques consist of deterministic approaches and computational intelligence algorithms. Deterministic methods include the priority list, dynamic programming, the branch-and-bound method, Lagrangian relaxation and mixed-integer programming. These methods may suffer from convergence problems. For example, the priority list approach is simple and fast, but it usually yields high production cost. The dynamic programming is flexible but suffers from the problem of high dimensionality. The Lagrangian relaxation method offers a faster solution; it may encounter numerical convergence problems. Recently, computational intelligence algorithms (CIAs) have been applied to cope with the UC problem, such as genetic algorithm (GA), simulated annealing (SA), evolutionary programming (EP), particle swarm optimization (PSO). CIAs are general-purpose stochastic optimization algorithms, and they offer great potential to attain global convergence, easy implementation and solution of complicated optimization problems. Nevertheless, CIAs are parameter-sensitive and computationally expensive. They often consume a considerable amount of computational time when dealing with large-scale UC problems.

The UC problem is characterized by numerous operating constraints. These constraints make the problem difficult to solve. As a result, constraint handling also represents a vital part in the UC problem. Currently, constraint handling on UC can be divided into penalty-based method and feasibility-based method. The penalty-based constraint handling method is used to help generating feasible unit schedules, and it does not guarantee feasible unit schedules in UC problems. In addition, the penalty approach needs to set an appropriate penalty factor. In different UC problems, the penalty factor is needed to be fine tuned again. The feasibility-based method always creates feasible solutions. However, the feasibility-based technique may require a large computational time.

This thesis presents a novel method for solving the UC problem based on quantum-inspired evolutionary algorithm (QEA). The proposed method applies QEA to handle the unit-scheduling problem and the Lambda-iteration technique to solve the economic dispatch problem. The QEA method is based on the concept and principles of quantum computing, such as quantum bits, quantum gates and superposition of states. QEA employs quantum bit representation, which has better population diversity compared with other representations used in evolutionary algorithms, and uses quantum gate to drive the population towards the best solution. The mechanism of QEA can inherently treat the balance between exploration and exploitation and also achieve better quality of solutions, even with a small population. In addition, an effective constraint handling technique is developed to ensure that feasible and potential UC solutions are produced in the optimization process.

The proposed QEA-UC method is applied to systems with the number of generating units in the range of 10 to 100 in a 24-hour scheduling horizon and is compared to conventional methods in the literatures. Moreover, the proposed method is extended to solve a large-scale UC problem in which 100 units are scheduled over a 7-day horizon with unit ramp-rate limits considered. The application studies have demonstrated the superior performance and feasibility of the proposed algorithm. Furthermore, the proposed QEA-UC is modified to solve the bi-objective UC considering two conflicting objective functions, 24-hour horizon and 10 generating units. The simulation results have shown the potential of QEA-UC to solve bi-objective UC problem.

Acknowledgements

I would like to express my most sincere gratitude to my chief supervisor Dr. C.Y. Chung, and co-supervisors Prof. T.S. Chung and Prof. S.L. Ho. In my research study, Dr. C.Y. Chung gave me invaluable guidance and comments. Prof. T.S. Chung and Prof. S.L. Ho provided me with many helps and suggestions throughout my study period.

Many thanks are also given to Prof. K.P. Wong. Prof. K.P. Wong provided me with helpful support and comments.

Last but no least, I would like to thank my family members and friends for their continuous supports and encouragements. Their supports and encouragements are also important to the achievement of this thesis.

Table of Contents

List of Abbreviation

CIAs	Computational intelligence algorithms
EAs	Evolutionary algorithms
EP	Evolutionary programming
GA	Genetic algorithm
PSO	Particle swarm optimization
QEA	Quantum-inspired evolutionary algorithm
QEA-UC	Quantum-inspired evolutionary algorithm-unit commitment
SA	Simulated annealing
UC	Unit commitment
EED	Environmental and economic dispatch

List of Figures

List of Tables

Chapter 1 Introduction

1.1. Background and Motivation

Unit commitment (UC) is an important optimization problem in power system operation. Its objective is to schedule the generating units online or offline over a scheduling horizon such that the power production cost is minimized with the load demand fully met and the operation constraints satisfied. In solving this problem, generator schedules are first found and their costs are evaluated through the economic dispatch calculation. The UC problem is highly difficult and combinatorial in nature, and it consists of many hard constraints. To show the complexity of the problem, different unit commitment scales are considered in Table 1.1.

Table 1.1 Problem complexity vs. problem scales

No. of Units	Time Horizon	Total Possible Combinations
1	24	16,777,216 or 2^{1x24}
3	24	4.72237E+21
5	24	1.32923E+36
7	24	3.74144E+50
9	24	1.05312E+65
10	24	1.76685E+72
20	24	3.1217E+144

Therefore, the problem complexity exponentially rises as the problem scale increases.

Coal-fired thermal plants give a significant percentage of emissions from generation plants. Nowadays, emission performance becomes increasingly important to power companies. To take emissions from power plants into account, emissions from thermal plants can be formulated into a bi-objective UC problem considering cost and emission objective functions. These two objective functions are conflicting in nature. The bi-objective UC gives a set of compromise solutions, which help power system operators on plant scheduling considering emissions. Because of the complicated problem features, such as mixed integer, bi-objectives and numerous operating constraints, the bi-objective UC is very difficult to solve.

1.2. Deterministic Approaches to Solving UC

Conventionally, deterministic techniques are applied to deal with the UC problem. These techniques consist of the priority list [1], dynamic programming [2-4], Lagrangian relaxation [5, 6], mixed-integer programming [7] and the branch-and-bound method [8]. The priority list approach is simple and fast, but it usually yields high production cost. The dynamic programming is flexible but suffers from the problem of high dimensionality. The branch-and-bound method uses a linear function to represent the fuel consumption and time-dependent start-up cost and obtains the required lower and upper bounds. However, its computational time increases exponentially with the increment of the dimension of the UC problem. The mixed-integer programming method employs linear programming technique to solve and check for an integer solution. But it also suffers from an excessive computational time requirement. While the Lagrangian relaxation method offers a faster solution, it may encounter numerical convergence problems. Artificial intelligence method based on heuristic depth-first search method [9, 10] was also developed but it can be limited in its application to large-scale UC problem.

1.3. Computational Intelligence Algorithms for Solving UC

Recently, computational intelligence algorithms (CIAs) have been successfully applied to solve the UC problem, such as genetic algorithm (GA) [11, 12], simulated annealing (SA) [13, 14, 15, 16], evolutionary programming (EP) [17], particle swarm optimization (PSO) [18], and hybrid methods [19]-[26]. These approaches are general-purpose stochastic optimization techniques, and they operate on a group of candidate solutions with different search mechanisms. These techniques have been reported to be capable of attaining global/near-global solution search. They have attracted much attention, owing to their great potential to escape from local convergence, easy implementation and accommodation of complex problem characteristics. Nevertheless, CIAs are parameter-sensitive and computationally expensive. They often require a considerable amount of computational time when solving the UC problem.

In the following sections, an overview of some CIA-based techniques for coping with the UC problem is presented.

1.3.1. Particle Swarm Optimization

PSO was introduced by Eberhart and Kennedy. It is an efficient, stochastic and population-based computational method [18, 35]. PSO starts by randomly initializing a group of candidate solutions, called particles. Each particle is evaluated by its fitness value and is governed by its velocity and position update equations. In PSO, particles converge under the influence of the global best particle and their best personal record. The PSO process stops when the pre-defined condition is met. On the other hand, the conventional PSO method may suffer from pre-mature convergence.

A PSO approach [18] to solving the UC problem has been proposed to apply the discrete PSO to solve the binary unit scheduling and real-coded PSO to calculate the economic load dispatch problem. The effectiveness of this approach has been tested in a small UC test system of 10-unit and a scheduling period of 24-hour. The results obtained have been shown to be better than some conventional methods, like GA, EP, DP, and LR. The effectiveness of this approach, however, has not been demonstrated to handle large-scale UC problems.

1.3.2. Evolutionary Programming

EP is a stochastic and population-based optimization algorithm based natural evolution [17, 36]. It starts by randomly initializing a population of candidate solutions or parents. Parents are evaluated by their fitness values. Off-springs are generated by mutating or altering their parents with respect to a Gaussian distribution. The population is evaluated and evolved through mutation, competition, and selection. The EP process terminates if the pre-set conditions are satisfied.

An EP method [17] has been applied to tackle the UC problem. In this method, a unit schedule or a solution is coded as a string of symbols. A population of candidate solutions are randomly initialized and evolved through applying EP operators, such as fitness evaluation, mutation, competition, and selection. This EP method has been validated in UC test systems of up to 100 units with a scheduling horizon of 24 hours. This method has performed better than other conventional techniques in terms of the quality of solutions and computational time.

1.3.3. Genetic Algorithm

GA is also a powerful, population-based and general-purpose stochastic optimization algorithm based on natural evolution [17], [37]. GA contains four genetic-inspired operators, namely mutation, crossover, inversion, and selection. In GA, each candidate is represented as a chromosome or a string of genes. A group of chromosomes are initialized randomly, and the chromosomes are evolved to explore and exploit the solution space by applying mutation, crossover, selection and inversion.

Various GA-based techniques have been proposed to solve the UC problem, including a GA solution to the UC problem [11], solving the UC problem with a GA through a constraint satisfaction technique [12], and a UC problem by using GA based on unit characteristic classification [20].

1.3.4. Simulated Annealing

SA is a powerful and population-based stochastic optimization algorithm simulated as an annealing process [24]-[26], [38]. SA simulates an annealing process that starts from a high energy state and transits to a low energy state. In the SA process, initial solutions and a high temperature are firstly created. The solutions are evaluated, disturbed and accepted based on the Boltzman function throughout the optimization process. The SA process stops if the per-set criteria are reached. Theoretically, SA can converge to a global optimum, but it demands a large computational time.

Many SA-based approaches to dealing with the UC problem have been reported in the literature [13]-[16], [19], [24]-[26]. These approaches consist of UC by SA [13], an SA algorithm for UC [14], SA for the UC problem [15], and UC by an enhanced SA algorithm [16]. Hybrid SA methods have also been suggested, such as thermal generator scheduling using hybrid GA/SA approach [24], combined GA/SA/fuzzy set approach to short-term generation scheduling with take-or-pay fuel contract [25], and hybrid GA/SA approach to short-term multiple-fuel-constrained generation scheduling [26].

1.4. Quantum-inspired Evolutionary Algorithm for Solving UC

Quantum-inspired evolutionary computing represents a combination of evolutionary computation and quantum computing. Quantum computing is a branch of study on evolutionary computation and employs the certain principles of quantum mechanics, such as superposition, interference and uncertainty [27, 28, 29, 30]. Based on the concept and principles of quantum computing, such as quantum bits (Q-bits), quantum gates (Q-gates) and superposition of states, Han and Kim [30] developed a quantum-inspired evolutionary algorithm (QEA), which can achieve a better balance between exploration and exploitation of the solution space and also obtain better solutions, even with a small population, compared with the conventional CIAs. The superior performance of QEA for combinatorial optimization problems was demonstrated in [30, 31].

By the development of QEA and its promising capability of for solving combinatorial optimization problem, this thesis was motivated to propose a novel QEA-based UC method (QEA-UC) to solve the UC problem. In QEA-UC, unit-scheduling problem is handled by QEA and a developed constraint handling technique. The economic dispatch problem is solved by the commonly-used method, Lambda-iteration technique.

1.5. Main Contributions

The main contributions of this thesis are highlighted as follows:

i) Development of a novel QEA-UC

QEA is naturally suitable for dealing with combinatorial optimization problems, like the UC problem. This thesis proposes a novel QEA-UC approach to solve the UC problem. In the QEA-UC technique, QEA is applied to produce unit schedules and Lambda iteration is used to solve the economic dispatch problem. The effectiveness and superior performance of QEA-UC are demonstrated by solving UC from small-scale to large-scale problems. Compared with those existing methods, QEA-UC performs much better in terms of the quality of solutions and computational efforts.

It is noted that the application of quantum-inspired evolution in power systems is still very new, and this thesis has successfully developed a novel technique to tackle the UC problem.

ii) Development of an efficient constraint handling

Constraint satisfaction is a vital part in the UC problem. To generate feasible unit schedules in the UC problem, an efficient constraint handling is developed. In the developed constraint handling technique, a heuristic method is firstly applied to satisfy spinning reserve constraints as well as minimum up/down time constraints of unit schedules, and then the unit schedules are improved simply by repeating once the heuristic method for constraint satisfaction.

iii) Modified QEA-UC for solving bi-objective unit commitment

In power system operation, the bi-objective unit commitment considering emission and cost minimization is one of the most difficult optimization problems. The bi-objective UC gives a trade-off curve between emission and cost, which is beneficial to the decision-making of system operators. The proposed QEA-UC is modified to solve the bi-objective UC problem by considering a weighted sum objective function combining emission and cost. Moreover, a random order list and cost-based priority list are proposed to construct the trade-off curve.

1.6. List of Publication

1. T. W. Lau, C. Y. Chung, K. P. Wong, T. S. Chung, S. L. Ho, "Quantum-Inspired Evolutionary Algorithm Approach for Unit Commitment", IEEE Transactions on Power Systems, Vol. 24, No. 3, August 2009, Page: 1053-1512.

1.7. Thesis Layout

Chapter 1 introduces the background and motivation of this thesis, different deterministic approaches to solving UC, various computational intelligence algorithms for solving the UC problem, main contributions and QEA. Main contributions, thesis layout and the list of publications are also presented in this Chapter.

Chapter 2 shows the mathematical formulation of the UC problem, including formulation of objective function, constraints and control variables.

Chapter 4 describes the basic principle and representation of QEA. The procedure of QEA is also described.

Chapter 5 presents the proposed method to solving the UC problem and the approach to handling the constraints in the UC problem.

Chapter 6 provides the introduction of bi-objective UC problem and the modification of the QEA-UC approach to solve the bi-objective UC problem with cost and emission objectives, 24-hour scheduling horizon and 10-unit.

Chapter 7 demonstrates the effectiveness of the QEA-UC method through the application studies of the proposed algorithm to UC test systems up to 100 units. A parameter sensitivity analysis and a performance comparison with other methods are also presented. Furthermore, the potential of QEA-UC to deal with bi-objective UC is shown and the results are discussed.

Chapter 8 gives concluding remarks of this thesis and future work.

Chapter 2 Problem Formulation

2.1. Introduction

The UC problem represents an important part in power systems. The objective of the UC problem is to schedule the generating units over a scheduling horizon such that the production cost is minimized and the operation constraints are satisfied. The UC problem can be divided into two parts, namely unit scheduling and economic dispatch. The unit scheduling involves selecting units to be turn on and off in a time horizon. The complexity of the problem increases with the problem dimension. For instance, given a UC problem with 10 generating units and a scheduling time period of 24 hours, the total possible combinations are equal to 2^{10x24} or approximately 1.7668×10^{72}. A large computational resource is required in order to completely solve the problem. The economic dispatch is to minimize the total production cost incurred from the power outputs of the scheduled units. Thus, the UC problem is combinatorial in nature and highly constrained.

In the following sections, the problem formulation is presented, including the mathematical representation of the objective function, numerous constraints and control variables. The list of symbols used in this Chapter is as follows:

N	Number of generating units
H	Number of hours
k	Index of unit (k = 1,2, …, N)
h	Index of time (h = 1,2, …, H)
p_{kh}	Control variable for the generation of unit k at hour h
u_{kh}	Control variable for the on/off status of unit k at hour h
F_H	Total system production cost within H hours
$F_{kh}(p_{kh})$	Fuel cost function of unit k at hour h
a_k , b_k , c_k	Cost function parameters of unit k
ST_{kh}	Start-up cost of unit k at hour h
HSC_k/ CSC_k	Hot/cold start-up cost of the k-th unit

MDT_k / MUT_k	Minimum down/up time of the *k*-th unit
CSH_k	Cold start hours of unit *k*
T_k^{off}	Duration during which unit *k* is continuously OFF
T_k^{on}	Duration during which unit *k* is continuously ON
D_h	System peak demand at hour *h*
R_h	Spinning reserve at hour *h*
$p_{k(max)} / p_{k(min)}$	Maximum/minimum output limit of unit *k*

2.2. Objective Function

The objective of UC problem is to minimize the total power production cost, comprising the fuel cost and the start-up cost, over a specified period of time or the scheduling horizon. The objective function can be expressed by:

$$\min.\ F_H = \sum_{h=1}^{H}\sum_{k=1}^{N}[F_{kh}(p_{kh}) + ST_{kh}(1-u_{k(h-1)})]u_{kh} \tag{1}$$

where $F_{kh}(p_{kh})$ and ST_{kh} are given by:

$$F_{kh}(p_{kh}) = c_k(p_{kh})^2 + b_k(p_{kh}) + a_k \tag{2}$$

$$ST_{kh} = \begin{cases} HSC_k, if \quad MDT_k \le T_k^{off} \le MDT_k + CSH_k \\ CSC_k, if \quad T_k^{off} > MDT_k + CSH_k \end{cases} \tag{3}$$

2.3. Constraints

In the UC problem, coupling constraints and local constraints are considered. The coupling constraints are related to all generating units for scheduling, while the local constraints are associated with the operating limits of each generator. The minimization of the objective in the UC problem is subjected to the following coupling and local constraints.

2.3.1. Power Balance Constraint

The total generation outputs must be equal to the total load demands over a time horizon.

$$\sum_{k=1}^{N} p_{kh} u_{kh} = D_h \tag{4}$$

2.3.2. Spinning Reserve Constraint

To provide reliable generation and sufficient on-line reserve power, the spinning reserve constraint is necessary to be met.

$$\sum_{k=1}^{N} p_{k(\max)} u_{kh} \geq D_h + R_h \tag{5}$$

2.3.3. Unit Output Constraint

The generation output of each generating unit is restricted to its maximum and minimum power outputs.

$$p_{k(\max)} \geq p_{kh} \geq p_{k(\min)} \tag{6}$$

2.3.4. Minimum Up Time Limit

Generating units are limited to a minimum up time. Once a generation unit is ON, it can not be shut down immediately.

$$T_k^{on} \geq MUT_k \tag{7}$$

2.3.5. Minimum Down Time Limit

Generators are also subjected to a minimum down time. Once a generation unit is OFF, it can not be started up immediately.

$$T_k^{off} \geq MDT_k \tag{8}$$

2.3.6. Ramp Rate Limits

Generating units cannot be adjusted to an output level instantaneously. They encounter ramp rate limits. When a unit is commanded to increase it output, it must not exceed its ramp-up power limit. On ramping down, each unit is required to satisfy its ramp-down limit.

2.3.7. Other Limits

In the UC problem, except for the above-mentioned constraints, other practical limitations can be included, such as crew constraints, fuel consumption constraints, network security constraints, stability constraints, and emission constraints.

2.4. Control Variables

The UC problem contains mixed-integer variables. These variables include the on or off states of each unit, and the power outputs of each generators. The state variables of each unit are discrete and usually represented as 1 or 0, whereas the power outputs are continuous variables.

Chapter 3 Quantum-inspired Evolutionary Algorithm

3.1. Introduction

Like other evolutionary algorithms (EAs), QEA [30] consists of the representation of individuals, evaluation functions as well as population dynamics. In quantum computing, a Q-bit is the smallest unit of information stored in a two-state quantum computer. QEA employs a Q-bit as a probabilistic representation, instead of binary, numeric or symbolic representation used in other EAs. A Q-bit individual is defined by a string of Q-bits and can represent a linear superposition of all the possible states in the search space. With this property, QEA requires only a small population size to provide good population diversity to effectively explore the solution space. A Q-gate is defined as a variation operator of QEA to drive the probability of each Q-bit to either 1 or 0 and toward the best single state with a gradual diminishing diversity property in the optimization process. By using the concept of Q-bit representation and superposition principle, each Q-bit individual can represent and explore all possible states. Moreover, with Q-gate operation, each Q-bit is driven to exploit a single state. Thus, the mechanism of the QEA method can inherently treat the balance between exploration and exploitation. In the following sections, the Q-bit representation and the principle and procedure of the QEA method are described.

3.2. Representation

A Q-bit, which is defined as the smallest unit of information, can be represented as:

$$\begin{bmatrix} \alpha \\ \beta \end{bmatrix} \tag{9}$$

where α and β are a pair of numbers with $|\alpha|^2 + |\beta|^2 = 1$. $|\alpha|^2$ and $|\beta|^2$ give the probabilities that the Q-bit will be found in the "0" and "1" states, respectively.

The state of a Q-bit may be "0", "1" or a linear superposition of the two and is expressed by:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \tag{10}$$

where $|0\rangle$ and $|1\rangle$ mean the states "0" and "1" respectively. $|\alpha|^2$ and $|\beta|^2$ determine the probabilities of states $|0\rangle$ and $|1\rangle$ respectively. Specifically, the larger the $|\alpha|^2$ value is, the higher the probability of the state $|0\rangle$ will be observed.

A Q-bit individual with a string of *m* Q-bits is defined as:

$$\left[\begin{array}{c|c|c|c|c} \alpha_1 & \alpha_2 & \alpha_3 & \cdots & \alpha_m \\ \beta_1 & \beta_2 & \beta_3 & \cdots & \beta_m \end{array}\right] \tag{11}$$

where $|\alpha_i|^2 + |\beta_i|^2 = 1$ for i=1, 2, …, m.

The merit of Q-bit representation is that a Q-bit individual can represent a linear superposition of states. By adopting the concept of Q-bit representation and superposition principle, a system with m Q-bits can represent 2^m states at the same time. When a Q-bit individual contains all α and β values equal to $1/\sqrt{2}$, the linear superposition of all possible states with the same probability can be represented by:

$$|\psi\rangle = \sum_{k=1}^{2^m} \frac{1}{\sqrt{2^m}} |X_k\rangle \tag{12}$$

where X_k is the k-th state and represented by the binary string $(x_1 x_2 \dots x_m)$ and x_i is either 0 or 1.

For instance, a Q-bit individual with two Q-bits is given by:

$$\left[\begin{array}{c|c} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{array}\right] \tag{13}$$

The states of the Q-bit individual can be represented as:

$$\frac{1}{2}|00\rangle + \frac{1}{2}|01\rangle + \frac{1}{2}|10\rangle + \frac{1}{2}|11\rangle \tag{14}$$

The results of (14) indicate that the probabilities of all states are 1/4, and the individual with two Q-bits holds the information of four states at the same time.

3.3. Quantum-inspired Evolution

QEA is characterized by the Q-bit representation for the population diversity, the observation process for making binary solutions from Q-bit individuals, the update process for driving the individuals towards better solutions by the rotation Q-gate, and termination conditions. The detailed procedure and mechanism of QEA, for solving a minimization problem with the objective function $f(X)$ and the binary control variables (*X*), are described as follows:

Step 1) Set the generation counter $t = 0$.

Step 2) Initialize *Q(t)*:

Q(t) represents a group of Q-bit individuals, which is initialized at $t = 0$, and $Q(t) = [q_1^t, q_2^t, ..., q_n^t]$, where subscript *n* is the total number of Q-bit individuals and q_j^t is the *j*-th Q-bit individual at generation *t* which is defined as:

$$q_j^t = \left[\begin{array}{c|c|c|c} \alpha_{j1}^t & \alpha_{j2}^t & ... & \alpha_{jm}^t \\ \beta_{j1}^t & \beta_{j2}^t & ... & \beta_{jm}^t \end{array} \right] \tag{15}$$

where, *j*=1, 2, …, *n* and *m* is the string length. If all α_{ji}^t and β_{ji}^t, for *i*=1,2,…,*m* , are initialized with 1/√2, then the probability of observing the state "1" or "0"of each Q-bit is the same.

Step 3) Determine *X(t)* by observing *Q(t)*:

X(t) is a group of binary solutions and are obtained by observing *Q(t)*. $X(t) = [X_1^t, X_2^t, ..., X_n^t]$, where X_j^t is a binary solution and obtained by observing q_j^t . $X_{j1}^t = [x_{j1}^t, x_{j2}^t, ..., x_{jn}^t]$, where x_{ji}^t is binary and determined by comparing $\left|\beta_{ji}^t\right|^2$ to a uniformly distributed random number in the range of 0 to 1. Here, x_{ji}^t is set to 1 if *random*[0,1] $<\left|\beta_{ji}^t\right|^2$; otherwise x_{ji}^t is set to 0.

Step 4) Evaluate *X(t)*:

The fitness or objective function values of the solutions in *X(t)* are evaluated.

Step 5) Store the best solution in *X(t)* into *B(t)*:

B(t) is a matrix that stores the best solution in the whole population. It should be noted that the local best solutions in subpopulations can also be considered. The details can be found in [30].

Step 6) Set $t = t+1$.

Step 7) Determine *X(t)* by observing *Q(t-1)*.

Step 8) Evaluate *X(t)*.

Step 9) Update *Q(t)* using Q-gates:

Q-bit individuals are updated by using Q-gates. A Q-gate is a variation operator of QEA to update the Q-bits, and the updated Q-bit at generation t $(\alpha_{ji}^t, \beta_{ji}^t)$ should meet the normalization condition, $\left|\alpha_{ji}^t\right|^2 + \left|\beta_{ji}^t\right|^2 = 1$. Rotation gates are considered in QEA. The rotation gate $U(\Delta\theta_{ji}^t)$ and the update operation are expressed as:

$$U(\Delta\theta_{ji}^t) = \begin{bmatrix} \cos(\Delta\theta_{ji}^t) & -\sin(\Delta\theta_{ji}^t) \\ \sin(\Delta\theta_{ji}^t) & \cos(\Delta\theta_{ji}^t) \end{bmatrix} \tag{16}$$

$$\begin{bmatrix} \alpha_{ji}^t \\ \beta_{ji}^t \end{bmatrix} = U(\Delta\theta_{ji}^t) \begin{bmatrix} \alpha_{ji}^{t-1} \\ \beta_{ji}^{t-1} \end{bmatrix} \tag{17}$$

where $\Delta\theta_{ji}^t$ is a rotation angle which determines the magnitude and direction of rotation. Fig. 3.1 illustrates the polar plot of the rotation gate for Q-bit individuals.

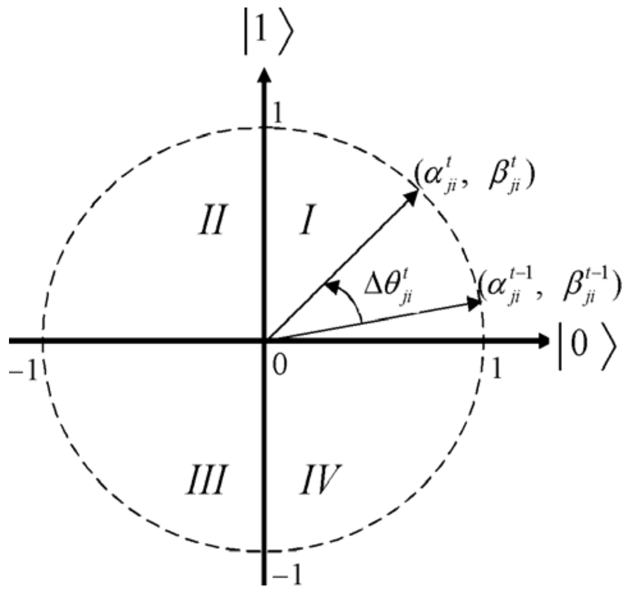

Fig. 3.1 Polar plot of the rotation gate for Q-bit individuals

Table 3.1 Lookup table of the rotation angle

x^t_{ji}	b^t_i	***Quadrant***	$f(X^t_j) \leqq f(B^t)$	$\Delta\theta^t_{ji}$
0	1	I/ III II/ IV	false false	$+\theta$ $-\theta$
1	0	I/ III II/ IV	false false	$-\theta$ $+\theta$
0	1	×	true	0
1	0	×	true	0
0	0	×	×	0
1	1	×	×	0

'x' denotes 'don't care'.

At generation *t*, the rotation angle $\Delta\theta^t_{ji}$ is updated according to the criteria summarized in Table 3.1, where x^t_{ji} and b^t_i are the binary control variables in solution X^t_j and the best solution B^t of $B(t)$, respectively. $f(X^t_j)$ and $f(B^t)$ represent the objective function values of X^t_j and B^t. For example, when x^t_{ji} and b^t_i are 0 and 1, and $f(X^t_j)$ is larger than $f(B^t)$, the rotation angle $\Delta\theta^t_{ji}$ is updated according to the following conditions:

i) if the Q-bit is in the first or third quadrant in Fig. 3.1, the value of $\Delta\theta^t_{ji}$ is set to a positive value or $+\theta$ to increase the probability of the state "1".

ii) if the Q-bit is in the second or fourth quadrant, the value of $\Delta\theta^t_{ji}$ is set to a negative value or $-\theta$ to increase the probability of the state

"1".

It is noted that the same lookup table can be used for the maximization problem. The details can be found in [30].

Step 10) Store the best solution into B(t):

The best solution among *X(t)* and *B(t-1)* is stored to *B(t)*.

Step 11) Check whether the stopping conditions are met:

Terminate if the stopping conditions are met; else go to Step 6.

3.4. Application of Quantum-inspired Evolution

Based on the concept and principles of quantum computing, Han and Kim [30] developed a QEA, which can better balance exploration and exploitation of the solution space and also obtain better solutions that the conventional EAs. The superior performance of QEA for combinatorial optimization problems was demonstrated in [30]. QEA for solving the classical knap-sack optimization problem [30] has been shown that QEA has outperformed conventional EAs in terms of the convergence speed and the quality of solutions. Even with a small population size, QEA can obtain better solutions than other EAs.

Chapter 4 Proposed Approach to the UC Problem

4.1. Introduction

The QEA method has been demonstrated to effectively solve combinatorial problems [30]. In this thesis, the QEA method is applied to handle the unit-scheduling problem. The economic dispatch of each UC schedule is calculated by the Lambda-iteration method to determine the optimal generation outputs of committed units. Representation for the UC problem, constraint handling, and the procedures of the QEA-UC method are described in the following sections.

4.2. Representation for the UC Problem

4.2.1. Q-bit Individuals for the UC Problem

A population of Q-bit individuals is initialized, $Q(t)=[q_1^t, q_2^t, ..., q_n^t]$, where q_j^t is defined as the *j*-th Q-bit individual at generation or iteration *t*; and *j* = 1, 2, …, *n*. Here '*n*' is the population size. To adopt QEA to the UC problem, each Q-bit individual is given by a *2N-by-H* matrix. *N* here is the total number of units and *H* is the total number of scheduling intervals in the scheduling horizon, *k* =1, 2, …, *N* and *h* =1, 2, …, *H*. Thus Q-bit individual q_j^t is represented by:

$$q_j^t = \left[\begin{array}{c} \begin{array}{c|c|c|c} \alpha_{j11}^t & \alpha_{j12}^t & \cdots & \alpha_{j1H}^t \\ \beta_{j11}^t & \beta_{j12}^t & \cdots & \beta_{j1H}^t \end{array} \\ \hline \cdots \\ \hline \begin{array}{c|c|c|c} \alpha_{jN1}^t & \alpha_{jN2}^t & \cdots & \alpha_{jNH}^t \\ \beta_{jN1}^t & \beta_{jN2}^t & \cdots & \beta_{jNH}^t \end{array} \end{array} \right] \tag{18}$$

4.2.2. Binary Solutions for Unit Schedules

U_j^t is a group of unit schedules, $U(t) = [U_1^t, U_2^t, ..., U_n^t]$, and each schedule U_j^t is an *N-by-H* matrix. By observing q^t_j, a binary solution or unit schedule U_j^t is formed as:

$$U_j^t = \begin{bmatrix} u_{j11}^t & u_{j12}^t & \cdots & u_{j1H}^t \\ u_{j21}^t & u_{j12}^t & \cdots & u_{j2H}^t \\ \cdots & \cdots & \cdots & . \\ u_{jN1}^t & u_{jN2}^t & \cdots & u_{jNH}^t \end{bmatrix} \quad (19)$$

4.2.3. Variables for Unit Outputs

For the unit schedules obtained in (2) in Section 2.2, the Lambda-iteration economic dispatch method is used to decide the optimal generation outputs of committed units, $P(t) = [P_1^t, P_2^t, ..., P_n^t]$, where *P(t)* represents the power generation of unit schedules at iteration *t*. The variables for the power generation of the *j*-th schedule p_j^t are given by:

$$P_j^t = \begin{bmatrix} p_{j11}^t & p_{j12}^t & \cdots & p_{j1H}^t \\ p_{j21}^t & p_{j12}^t & \cdots & p_{j2H}^t \\ \cdots & \cdots & \cdots & . \\ p_{jN1}^t & p_{jN2}^t & \cdots & p_{jNH}^t \end{bmatrix} \tag{20}$$

where p_{jkh} represents the generation of unit *k* at time interval *h* of the *j*-th unit schedule.

4.2.4. Evaluation Function

Since the minimization of the total operation cost is the objective of the UC problem, the objective function in (1) is used as the evaluation function of each unit schedule and the corresponding p_{kh} is obtained the Lambda-iteration method.

4.3. Constraint and Over-commitment Handling

In the UC problem, an optimum solution not only gives minimum production cost, but also it satisfies all operating constraints. Constraint satisfaction of unit scheduling solutions can be divided into penalty-based constraint handling and feasibility-based constraint handling [9]-[20], [24]-[26]. Compared with the feasibility-based constraint handling, the penalty-based constraint handling method has no guarantee that it will produce feasible unit schedules, especially in large-scale UC problems. Moreover, with the penalty approach, there is a need to find appropriate values for the penalty factors. For different problems, the penalty factors must be tuned again. However, the feasibility-based constraint handling approach can take a large computational time to produce feasible and potential solutions.

This thesis proposes a simple constraint handling approach to ensure that any unit schedule generated by QEA-UC is feasible. In the proposed constraint handling method, all feasible unit schedules are also be improved to reduce the production cost associated with over-commitment. The procedure of the constraint handling method is described in the following steps.

Step 1) Satisfying the spinning reserve constraints:

If the spinning reserve constraint at any scheduling interval is violated, further commitment in that interval is required and the start-up order is based on the full load average cost as in the priority list method [1]. De-committed unit with the highest priority are selected to be online first and the further commitment procedure stops immediately when the constraint is met.

Step 2) Handling over-commitment:

Excessive generation capacity may result in expensive production cost. When the total maximum generation capacity in a scheduling interval is higher than the summation of the load demand and spinning reserve, units are selected offline in the reverse order according to their priority orders until any further de-commitment will lead to deficiency in generation capacity.

Step 3) Satisfying the minimum up/down time constraint:

Extra units are committed over a period of time to observe the minimum up time constraint whenever the minimum up/down time constraint is dissatisfied.

Step 4) Improving unit schedules:

Excessive commitment may be caused by the action of Step 3, but it can be efficiently solved by repeating Steps 1 to 3. In our experiments, one iteration should be sufficient to produce potential solutions that meet constraints (5), (7) and (8).

Step 5) Examining unit schedules:

For every schedule, examine whether or not the generation capacity in all scheduling intervals is adequate. If not, a new schedule is created by observing the corresponding Q-bit individual, and repeats the above steps until a feasible schedule is produced.

4.4. Procedure of the QEA-UC Method

The procedure of the proposed QEA-UC approach is presented in the following steps and the corresponding flowchart is provided in Fig. 4.1.

Step 1) Set the generation counter $t = 0$.

Step 2) Initialize a group of Q-bit individuals, $Q(t) = [q_1^t, q_2^t, ..., q_n^t]$, with all α and β values equal to $1/\sqrt{2}$. Here q_j^t is defined as the j-th candidate solution at generation or iteration t and n is the population size. Initializing all α and β to $1/\sqrt{2}$ gives that all binary solutions are observed with the same probability, contributing to a good diversity at the initial stage.

Step 3) Determine unit schedules $U(t)$ by observing the states of Q-bit individuals, $U(t) = [U_1^t, U_2^t, ..., U_n^t]$. For U^t_j, u^t_{jNH} is set to 1 if *random* [0, 1] < $|\beta^t_{jNH}|^2$, else u^t_{jNH} is set to 0.

Step 4) Improve the obtained unit schedules by the proposed constraint handling method. According to the proposed constraint handling, all unit schedules generated in Step 3) are handled in the following steps.

Step 4.1) Satisfy the spinning reserve constraints

Step 4.2) Handle over-commitment.

Step 4.3) Satisfy the minimum up/down time constraint.

Step 4.4) Improve unit schedules.

Step 4.5) Examine unit schedules.

Step 5) Determine the cost of the schedule by determining the optimal economic dispatch of the units in each schedule by the Lambda-iteration method. The cost of each unit schedule is directly used as its evaluation value.

Step 6) If $t = 0$, then go to Step 8).

Step 7) Update Q-bit individuals by using Q-gates according to the lookup table in Table 3.1.

Step 8) Compare the costs of the schedules and store the best solution schedule.

Step 9) Set $t = t + 1$.

Step 10) Terminate if t is larger than the maximum number of generations; else go to Step 3).

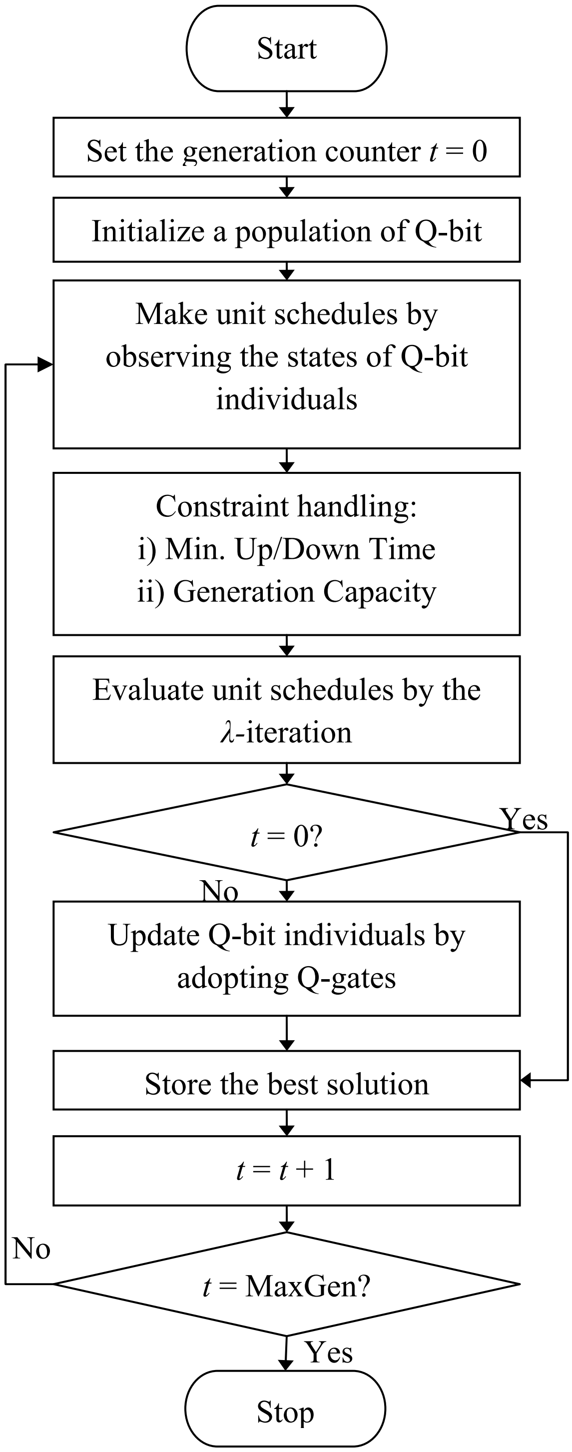

Fig. 4.1 Flowchart of the QEA-UC method

Chapter 5 Modified QEA-UC to Solve Bi-Objective UC

5.1. Introduction

With the increasing environmental awareness, emission performance becomes an important part to power utilities. Although a lot of researchers have suggested bi-objective economic dispatch or environmental and economic dispatch (EED) to reduce emissions during power generation activities [39]-[58], the EED problems only consider power dispatch at one time interval. Compared with the EED calculation, the UC problem including environmental considerations is to schedule units over a spectrum of time intervals and thus contributes much more to emission mitigation. However, it has been found that only limited research has been conducted on the UC problem considering emission impacts [59], [60], [61] and 62].

Emissions from thermal plants can be incorporated into the UC problem considering minimization of both cost and emission objective functions [59], [60]. The two objective functions are conflicting in nature. The bi-objective UC gives a set of compromise solutions, which benefit the decision making of power system operators on plant scheduling considering emission control. This Chapter presents the formulation of the bi-objective UC problem and modification of QEA-UC for solving the problem.

5.2. Formulation of Bi-Objective UC

Conventionally, the objective of the UC problem is to minimize the total power production cost, comprising the fuel cost and the start-up cost, over a specified period of time. Different from the conventional UC problem, the bi-objective UC problem considers minimization of two conflicting objective functions, namely emission and cost functions. The emission function can be approximated by a quadratic or exponential function [59]-[60]. In this thesis, a quadratic emission function is used for studying the bi-objective UC problem with the modified QEA-UC. The total emission function E_H can be expressed by:

$$min.\ E_H = \sum_{h=1}^{H} \sum_{k=1}^{N} [\ E_{kh}(p_{kh})\ ]\ u_{kh} \tag{21}$$

$$E_{kh}(p_{kh}) = e_{3k}(p_{kh})^2 + e_{2k}(p_{kh}) + e_{1k} \tag{22}$$

where $E_{kh}(p_{kh})$ is the emission function of the *k-th* unit at hour *h*. $e_{3k},\ e_{2k}$ and e_{1k} are the emission function coefficients of the *k-th* unit.

5.3. Modified QEA-UC to Solve Bi-Objective UC

5.3.1. Representation of Objective Function

The bi-objective UC problem involves a set of compromise solutions. These solutions can be obtained by optimizing a weighted sum function [23], [59]-[60]. To demonstrate the performance of the modified QEA-UC, this thesis considers a weighted sum method to deal with the bi-objective problem. The weighted sum function $F(c,e)$ can be formulated as:

$$min.\ F(c,e) = w(F_H) + (1 - w)\,\kappa\,(E_H) \tag{23}$$

where F_H and E_H represents the total cost function and total emission function, respectively. κ is a scaling factor. w is a weighting factor between 1 and 0. If w is set to 1, only F_H is considered as the weighted sum function. Only E_H is regarded as the weighted sum function if w is equal to 0. By varying w between 1 and 0, a set of trade-off solutions of these two conflicting functions can be produced by optimizing the weighted sum function.

5.3.2. Cost-based Priority List and Random Order List

5.3.2.1.Cost-based Priority List

As presented in Chapter 4, the proposed QEA-UC considers scheduling units according to their full load average cost as in the priority list method [1], or called the cost-based priority list as described in Chapter 4.3. Intuitively, the modified QEA-UC approach with the cost-based priority list method gives cost-biased solutions and may not give a full picture of the trade-off curve. Nevertheless, the cost-based priority list can provide extra information for power system operators to deal with plant cost-consciousness or the cost objective function weighing higher than that of the emission function.

5.3.2.2. Random Order List

In addition to the cost-based priority list, a random order list is also developed. Different form the cost-based priority list, the random order list randomly generate an order list to schedule the on or off state of generating units without any objective biasing. Thus, the random order list can provide power system operators with a full picture of the trade-off between emissions and cost. In Chapter 6.7, numerical simulation and comparison on the performance of these two lists are made.

5.3.3. Procedure of Modified QEA-UC for Bi-Objective UC

The proposed QEA-UC is modified to adapt to the bi-objective UC problem. The modified QEA-UC involves finding a set of compromise solutions obtained after a number of separate optimization runs with different weighting factors. For each individual optimization run, the weighting factor, population size and maximum number of generations of QEA-UC are specified. The global control parameters of QEA-UC, the listing method and the weighting step size are firstly defined in the overall optimization program. The procedure of the modified QEA-UC method for coping with the bi-objective UC problem is given in the following steps.

Step 1) Set the weighting factor $w = 1$ and select either cost-based priority list or random order list. It is noted that $w = 1$ means only the cost objective function is considered.

Step 2) Set the generation counter $t = 0$.

Step 3) Initialize a group of Q-bit individuals, $Q(t) = [q_1^t, q_2^t, ..., q_n^t]$, with all α and β values equal to $1/\sqrt{2}$.

Step 4) Determine unit schedules by observing the states of Q-bit individuals.

Step 5) Improve the obtained unit schedules by the proposed constraint handling technique.

Step 5.1) Satisfy the spinning reserve constraints

Step 5.2) Handle over-commitment.

Step 5.3) Satisfy the minimum up/down time constraint.

Step 5.4) Improve unit schedules.

Step 5.5) Examine unit schedules.

Step 6) Determine the evaluation value of each schedule by determining the weighted sum value of each solution by the Lambda-iteration method.

Step 7) If $t = 0$, then go to Step 9).

Step 8) Update Q-bit individuals by using Q-gates according to the lookup table in Table 3.1.

Step 9) Compare the evaluation values of the schedules and store the best solution schedule.

Step 10) Set $t = t + 1$.

Step 11) Store the best compromise solution and go to Step 12) if t is larger than the maximum number of generations; else go to Step 4).

Step 12) Decrease w by Δw, where w is between 0 and 1.

Step 13) Terminate if w is smaller than 0; else go to Step 2).

Chapter 6 Numerical Results and Discussions

6.1. Introduction

The proposed QEA-UC method is tested on systems with the number of units in the range of 10 to 100 and considering a 24-hour scheduling horizon. The 10-unit system data and load demands are given in [11] and shown in Tables 6.1 and 6.2. The 20-unit, 40-unit, 60-unit, 80-unit and 100-unit data are obtained by duplicating the 10-unit case and adjusting the load demands in proportion to the size of the system. The spinning reserve requirements are assumed to be 10% of the load demand. For each test case, totally 30 trial runs are performed to verify the robustness of the QEA-UC method. The proposed QEA-UC method has been developed based on MATLAB and executed on a computer with Intel Core of 2.39 GHz and 1.99 GB RAM.

In this Chapter, parameter sensitivity analysis is first performed. Case studies on the performance of the QEA-UC method on different test systems are then reported. The results obtained are compared with some published methods in the literatures. Furthermore, the proposed method is extended to solve a large-scale UC problem in which 100 units are scheduled over a 7-day horizon with unit ramp-rate limits considered. Finally, the proposed QEA-UC is modified to cope with bi-objective UC. The simulation results obtained are analyzed.

Table 6.1.i) Coefficients of the test system [11]

	Unit 1	Unit 2	Unit 3	Unit 4	Unit 5
Pmax (MW)	455	455	130	130	162
Pmin (MW)	150	150	20	20	25
c ($/MW^{2}h)	0.00048	0.00031	0.002	0	0.00398
b ($/MWh)	16.19	17.26	16.6	16.5	19.7
a ($/h)	1000	970	700	680	450
Min Up (h)	8	8	5	5	6
Min Down (h)	8	8	5	5	6
Hot start cost ($)	4500	5000	550	560	900
Cold start cost ($)	9000	10000	1100	1120	1800
Cold start hrs (h)	5	5	4	4	4
Initial status (h)	8	8	-5	-5	-6

Table 6.1.ii) Coefficients of the test system [11]

	Unit 6	Unit 7	Unit 8	Unit 9	Unit 10
Pmax (MW)	80	85	55	55	55
Pmin (MW)	20	25	10	10	10
c ($/MW^{2}h)	0.007	0.00079	0.00413	0.00222	0.00173
b ($/MWh)	22.26	27.74	25.92	27.27	27.79
a ($/h)	370	480	660	665	670
Min Up (h)	3	3	1	1	1
Min Down (h)	3	3	1	1	1
Hot start cost ($)	170	260	30	30	30
Cold start cost ($)	340	520	60	60	60
Cold start hrs (h)	2	2	0	0	0
Initial status (h)	-3	-3	-1	-1	-1

Table 6.2 Load demand of the test system [11]

Hour	Load (MW)	Hour	Load (MW)
1	700	13	1400
2	750	14	1300
3	850	15	1200
4	950	16	1050
5	1000	17	1000
6	1100	18	1100
7	1150	19	1200
8	1200	20	1400
9	1300	21	1300
10	1400	22	1100
11	1450	23	900
12	1500	24	800

6.2. Parameter Sensitivity Analysis

The effects of the magnitude of rotation angle in radians and the population size are studied on the 10-unit test system. The maximum number of generations is set to 100 for the parameter sensitivity tests.

6.2.1. Determination of the Rotation Angle

The value of θ is problem-dependent [30, 32]. In this study, the population size is set to 4 and the values of θ from 0.005π to 0.05π with a step size of 0.005π are examined. The results are tabulated in Table 6.3. It can be observed that the QEA-UC method is sensitive to the magnitude of θ. Since large angles may cause premature convergence, small angles generally produce better solutions. The performance is the best when $\theta = 0.02\pi$.

Table 6.3 Effects of the magnitude of rotation angle on cost results

θ (×π radians)	**Cost ($)**		
	Best	**Mean**	**Worst**
0.005	563,974	564,420	564,833
0.010	563,938	564,304	564,917
0.015	563,948	564,333	564,711
0.020	563,938	564,268	564,711
0.025	563,956	564,305	565,031
0.030	563,938	564,331	565,067
0.035	563,938	564,358	564,927
0.040	563,938	564,413	565,093
0.045	563,948	564,381	564,946
0.050	563,977	564,454	565,031

6.2.2. Determination of the Population Size

The effects of the population size are investigated by varying the size from 2 to 30 with a step size of 2, and $\theta = 0.02\pi$. The results are showed in Table 6.4 and Fig. 6.1.

Noticeably, in Table 6.4, the best solutions are the same for the use of different population sizes. A large population size can slightly improve the mean value of the solutions, but it increases the computational time. From the size of 16 to 30, the improvement of the solutions is not pronounced, but the computational time increases linearly as shown in Fig. 6.1. Although the population size is problem-dependent, the size of 18 is the best choice in compromise between computational time and solutions. It is also agreed with the range of population size from 10 to 30 suggested in [32].

Table 6.4 Effects of the population size on mean time and cost

Population Size	Mean Time (s)	Cost ($)		
		Best	Mean	Worst
2	1.14	563,938	564,415	565,168
4	2.22	563,938	564,289	564,714
6	3.27	563,938	564,193	564,711
8	4.31	563,938	564,128	564,672
10	5.42	563,938	564,091	564,672
12	6.46	563,938	564,115	564,711
14	7.54	563,938	564,100	564,711
16	8.59	563,938	564,032	564,729
18	9.62	563,938	563,994	564,672
20	10.93	563,938	564,020	564,672
22	12.02	563,938	564,036	564,711
24	12.93	563,938	564,023	564,711
26	14.00	563,938	564,016	564,672
28	15.06	563,938	564,008	564,672
30	16.39	563,938	564,043	564,672

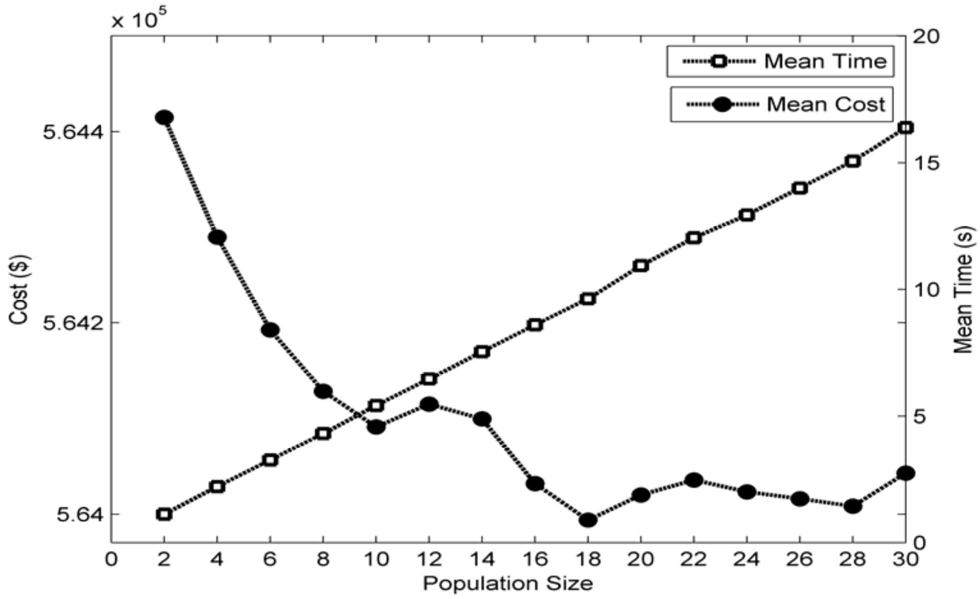

Fig. 6.1 Effects of the population size on cost and computational time

6.3. Effectiveness of the Proposed Constraint Handling Technique

In Chapter 4.3., the proposed constraint and over-commitment handling is presented. Over-commitment can be caused in the steps that satisfy the minimum up/down time constraint and generation capacity constraint. A simple technique is proposed to handle the over-commitment problem and improve unit schedules. Details can be referred to Chapter 4.3.

The effectiveness of the proposed method with over-commitment handling and without over-commitment handling is shown in Fig. 6.2. Obviously, the average convergence curve obtained by the proposed method considering over-commitment handling is much better than the curve generated without considering over-commitment handling. This shows the effectiveness and efficiency of the proposed constraint handling technique.

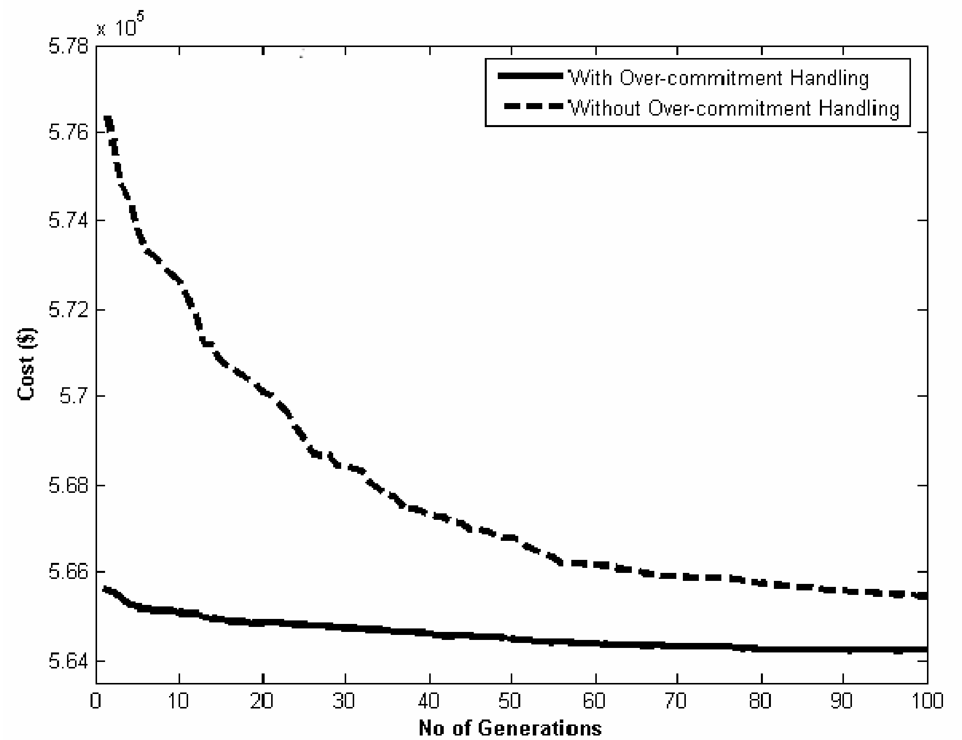

Fig. 6.2 Average convergence curves of the proposed QEA-UC method with/without over-commitment handling

6.4. Case Studies

The performance of the proposed QEA-UC method is tested on the system with the number of units from 10 to 100. Based on the selected settings in the previous Section, the population size equal to 18, $\theta = 0.02\pi$ and the maximum number of generations equal to 100, the results of different case studies are obtained and tabulated in Table 6.5. The population size of 4 is also included for comparison.

Obviously, the QEA-UC method with the population size of 18 outperforms the one with the size equal to 4 in terms of the best, mean as well as worst costs, and the standard deviation of the results on the systems with different problem scales. As shown in Table 6.6, when the maximum number of generations is set to 200, the better solutions can be obtained, especially for small population and large system sizes, but longer computation is needed.

The average cost convergence curves of the QEA-UC method for different cases are presented in Figs. 6.3-6.8. It can be observed that the convergence behavior of the QEA-UC method is very smooth. For the 10-unit system, the convergence curves become nearly steady after 100 generations so the improvement of solutions are not significant for the use of higher maximum number of generation. For 100-unit system, the solution can be further improved after 100 generations so higher number of generation can achieve a solution with lower cost. However, the curve becomes nearly steady after 200 generations and further increment of the number of generation only improves the solution slightly. A compromise between them can be considered because of the linear relationship between the computational time and the number of generation.

Table 6.5 Results of Proposed method with maximum no. of generations set to 100

No. of Units	Population Size	Maximum Generation	Mean Time (s)	Cost ($)		
				Best	Mean	Worst
10	4	100	2.18	563,938	564,289	564,714
	18		9.62	563,938	563,994	564,672
20	4	100	3.05	1,124,244	1,125,777	1,126,672
	18		13.71	1,123,933	1,125,048	1,125,926
40	4	100	4.42	2,247,036	2,248,207	2,249,702
	18		19.83	2,246,381	2,247,154	2,249,242
60	4	100	5.82	3,369,351	3,371,039	3,373,062
	18		26.23	3,367,186	3,369,203	3,370,699
80	4	100	7.30	4,491,896	4,494,710	4,497,278
	18		32.78	4,490,537	4,491,903	4,494,269
100	4	100	8.75	5,615,242	5,618,308	5,622,678
	18		39.24	5,611,696	5,614,434	5,616,478

Table 6.6 Results of Proposed method with maximum no. of generations set to 200

No. of Units	Population Size	Maximum Generation	Mean Time (s)	Cost ($)		
				Best	Mean	Worst
10	4	200	4.22	563,938	564,212	564,711
	18		19.23	563,938	563,969	564,672
20	4	200	6.14	1,123,824	1,125,387	1,126,578
	18		27.69	1,123,607	1,124,689	1,125,715
40	4	200	9.00	2,246,601	2,247,798	2,249,732
	18		43.13	2,245,557	2,246,728	2,248,296
60	4	200	11.81	3,367,278	3,369,357	3,372,181
	18		54.34	3,366,676	3,368,220	3,372,007
80	4	200	14.78	4,489,324	4,491,819	4,494,788
	18		66.42	4,488,470	4,490,128	4,492,839
100	4	200	17.90	5,611,344	5,614,397	5,616,847
	18		79.98	5,609,550	5,611,797	5,613,220

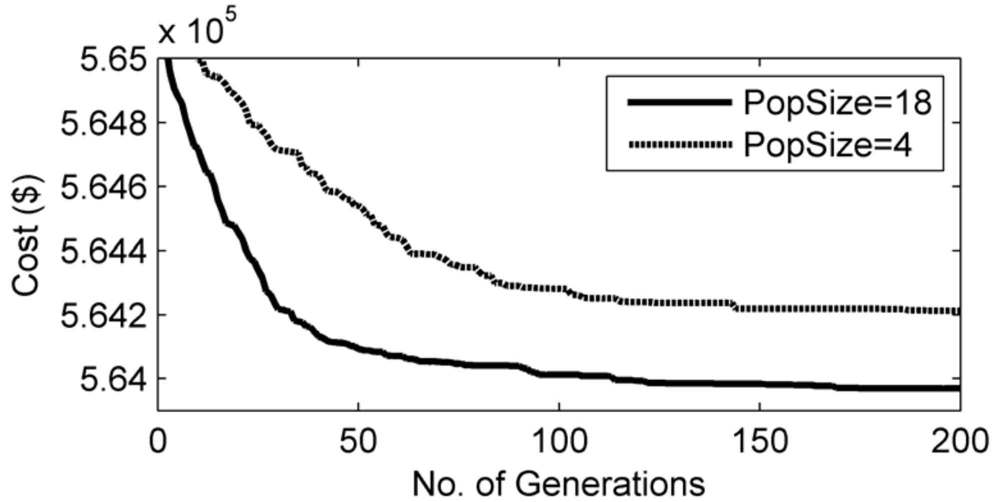

Fig. 6.3 Average convergence curves of the proposed QEA-UC method for the test systems with 10 units

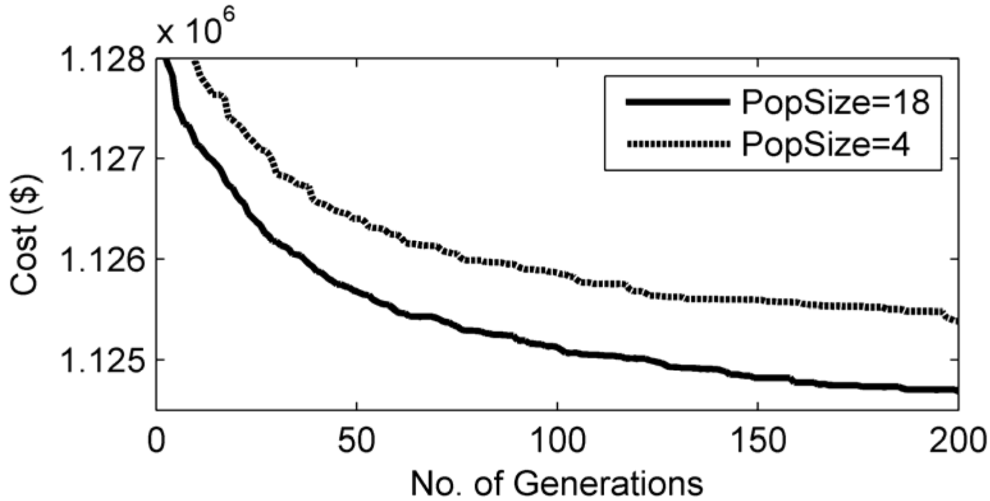

Fig. 6.4 Average convergence curves of the proposed QEA-UC method for the test systems with 20 units

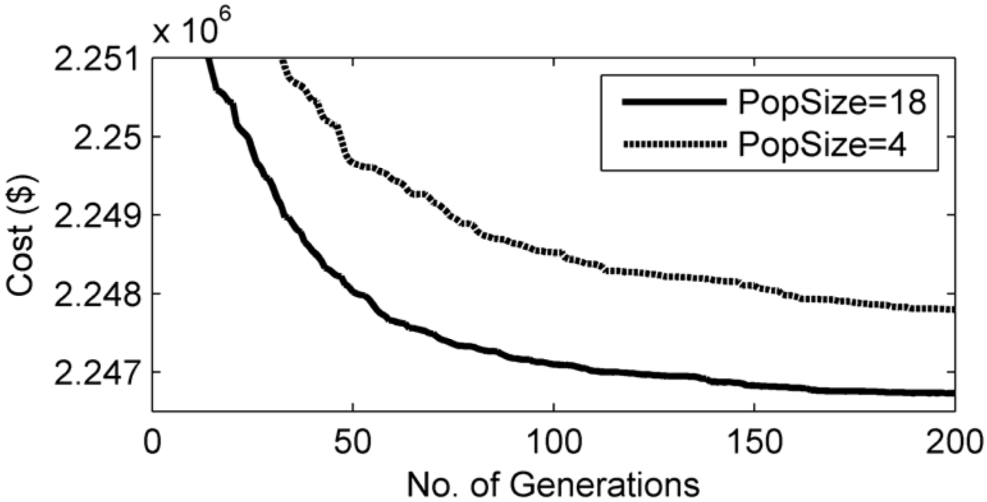

Fig. 6.5 Average convergence curves of the proposed QEA-UC method for the test systems with 40 units

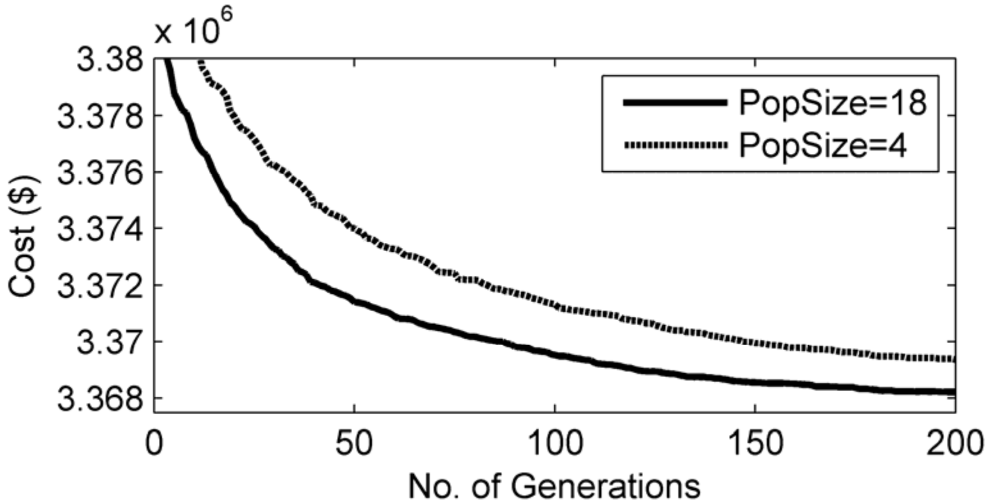

Fig. 6.6 Average convergence curves of the proposed QEA-UC method for the test systems with 60 units

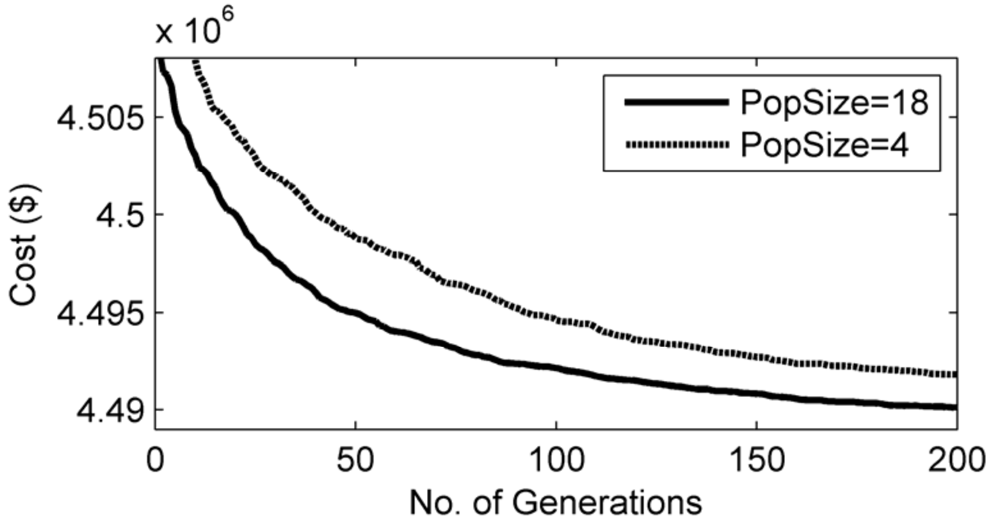

Fig. 6.7 Average convergence curves of the proposed QEA-UC method for the test systems with 80 units

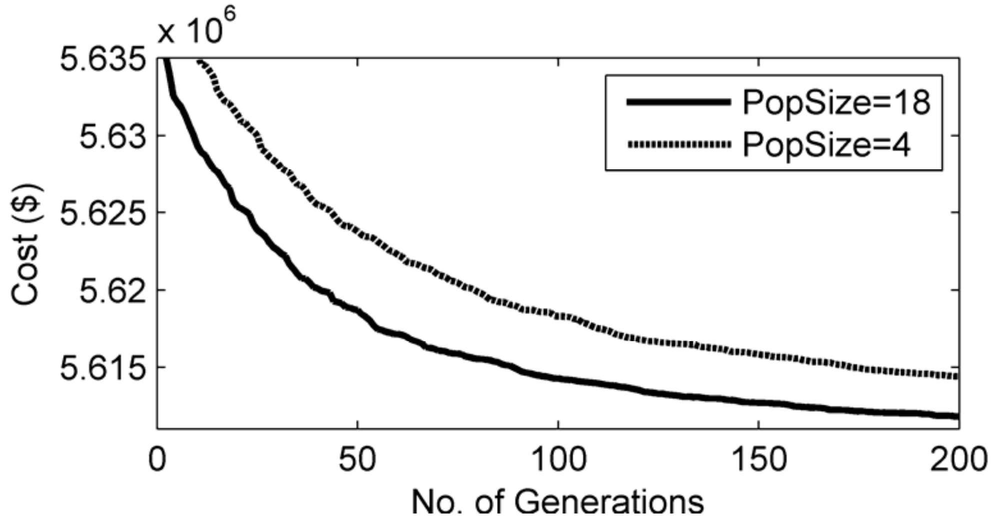

Fig. 6.8 Average convergence curves of the proposed QEA-UC method for the test systems with 100 units

6.5. Comparison of Results among Various Approaches

Table 6.7 to Table 6.12 summarize the study results on the test systems in the last section above obtained by the proposed QEA-UC method and other methods including LR[34], GA[11], EP[17], HPSO[18], SA[16], and GAUC[20]. In Tables 6.7-6.12, for the QEA-UC method, the results obtained with both population size of 4 and 18 are tabulated to show further the effects of the population size. In addition, the results obtained for both the maximum number of generations of 100 and 200 are summarized for the same reason.

In Tables 6.7-6.12, it can be observed that the solutions of the QEA-UC method are more attractive than those obtained by other techniques. Besides, it can be observed that the population size and maximum number of generations required by the QEA-UC method are much smaller than that of the other techniques in all the study cases. Although the mean times consumed by various approaches cannot be directly compared due to different computing machines used by other researchers, it can still be able to indicate that the computational time of the QEA-UC method increases linearly with the system size while that of other techniques increases dramatically. This efficient characteristic of the QEA-UC method indicates that QEA-UC has large capability in solving large-scale UC problems. Even for the case of population size of 4 and the maximum generation of 100, QEA-UC finds better solution than all the other methods considered and with a very short computational time. As expected, with the population size of 18 and maximum generation of 200, a much better solution is found by the proposed method. Owing to the linear relationship between the computational time and the system size, the superiority of the proposed QEA-UC method over the other methods considered in terms of solution quality and computation time is more significant in the 100-unit system.

Table 6.7 Comparison of QEA-UC with other methods in the test system with 10 units

No. of Units	Method	No. of Trials	Population Size	Maximum Generation	Cost ($)		
					Best	Mean	Worst
10	LR [34]	-	-	-	566,107	-	-
	GA [11]	20	50	500	565,825	-	570,032
	EP [17]	20	50	500	564,551	565,352	566,231
	HPSO [18]	50	20	1,000	563,942	564,772	565,785
	SA [16]	10	-	-	565,828	565,988	566,260
	UCC-GA [20]	20	20	500	563,977	-	565,606
	QEA-UC	30	4	100	563,938	564,289	564,714
			18	200	563,938	563,969	564,672

Table 6.8 Comparison of QEA-UC with other methods in the test system with 20 units

No. of Units	Method	No. of Trials	Population Size	Maximum Generation	Cost ($)		
					Best	Mean	Worst
20	LR [34]	-	-	-	1,128,362	-	-
	GA [11]	20	50	1,000	1,126,243	-	1,132,059
	EP [17]	20	50	1,000	1,125,494	1,127,257	1,129,793
	SA [16]	10	-	-	1,126,251	1,127,955	1,129,112
	UCC-GA [20]	20	20	1,000	1,125,516	-	1,128,790
	QEA-UC	30	4	100	1,124,244	1,125,777	1,126,672
			18	200	1,123,607	1,124,689	1,125,715

Table 6.9 Comparison of QEA-UC with other methods in the test system with 40 units

No. of Units	Method	No. of Trials	Population Size	Maximum Generation	Cost ($)		
					Best	Mean	Worst
40	LR [34]	-	-	-	2,250,223	-	-
	GA [11]	20	50	2,000	2,251,911	-	2,259,706
	EP [17]	20	50	2,000	2,249,093	2,252,612	2,256,085
	SA [16]	10	-	-	2,250,063	2,252,125	2,254,539
	UCC-GA [20]	20	20	2,000	2,249,715	-	2,256,824
	QEA-UC	30	4	100	2,247,036	2,248,207	2,249,702
			18	200	2,245,557	2,246,728	2,248,296

Table 6.10 Comparison of QEA-UC with other methods in the test system with 60 units

No. of Units	Method	No. of Trials	Population Size	Maximum Generation	Cost ($)		
					Best	Mean	Worst
60	LR [34]	-	-	-	3,374,994	-	-
	GA [11]	20	50	3,000	3,376,625	-	3,384,252
	EP [17]	20	50	3,000	3,371,611	3,376,255	3,381,012
	SA [16]	-	-	-	-	-	-
	UCC-GA [20]	20	20	3,000	3,375,065	-	3,382,886
	QEA-UC	30	4	100	3,369,351	3,371,039	3,373,062
			18	200	3,366,676	3,368,220	3,372,007

Table 6.11 Comparison of QEA-UC with other methods in the test system with 80 units

No. of Units	Method	No. of Trials	Population Size	Maximum Generation	Cost ($)		
					Best	Mean	Worst
80	LR [34]	-	-	-	4,496,729	-	-
	GA [11]	20	50	4,000	4,504,933	-	4,510,129
	EP [17]	20	50	4,000	4,498,479	4,505,536	4,512,739
	SA [16]	10	-	-	4,498,076	4,501,156	4,503,987
	UCC-GA [20]	20	20	4,000	4,505,614	-	4,527,847
	QEA-UC	30	4	100	4,491,896	4,494,710	4,497,278
			18	200	4,488,470	4,490,128	4,492,839

Table 6.12 Comparison of QEA-UC with other methods in the test system with 100 units

No. of Units	Method	No. of Trials	Population Size	Maximum Generation	Cost ($)		
					Best	Mean	Worst
100	LR [34]	-	-	-	5,620,305	-	-
	GA [11]	20	50	5,000	5,627,437	-	5,637,914
	EP [17]	20	50	5,000	5,623,885	5,633,800	5,639,148
	SA [16]	10	-	-	5,617,876	5,624,301	5,628,506
	UCC-GA [20]	20	20	5,000	5,626,514	-	5,646,529
	QEA-UC	30	4	100	5,615,242	5,618,308	5,622,678
			18	200	5,609,550	5,611,797	5,613,220

For the 10-unit test case, the best UC schedule obtained by the QEA-UC method is given in Table 6.13. In Table 6.13, the power outputs of the scheduled-on generating units at a scheduling period of 24-hour are shown. The best UC schedule gives a total production cost of $563,938 and satisfies all operating constraints. The best cost obtained by QEA-UC is much better than those found by LR[34], GA[11], EP[17], HPSO[18], SA[16], and GAUC[20]. Although the global optimum solution is unknown yet, the best result ($563,938) obtained by the proposed QEA-UC approach has shown its high effectiveness and promising capability to solve the UC problem.

Table 6.13 Best UC schedule of the 10-unit test system on 24-hour scheduling horizon with one-hour interval

Hour	Generating Unit									
	1	2	3	4	5	6	7	8	9	10
1	455	245	0	0	0	0	0	0	0	0
2	455	295	0	0	0	0	0	0	0	0
3	455	370	0	0	25	0	0	0	0	0
4	455	455	0	0	40	0	0	0	0	0
5	455	390	0	130	25	0	0	0	0	0
6	455	360	130	130	25	0	0	0	0	0
7	455	410	130	130	25	0	0	0	0	0
8	455	455	130	130	30	0	0	0	0	0
9	455	455	130	130	85	20	25	0	0	0
10	455	455	130	130	162	33	25	10	0	0
11	455	455	130	130	162	73	25	10	10	0
12	455	455	130	130	162	80	25	43	10	10
13	455	455	130	130	162	33	25	10	0	0
14	455	455	130	130	85	20	25	0	0	0
15	455	455	130	130	30	0	0	0	0	0
16	455	310	130	130	25	0	0	0	0	0
17	455	260	130	130	25	0	0	0	0	0
18	455	360	130	130	25	0	0	0	0	0
19	455	455	130	130	30	0	0	0	0	0
20	455	455	130	130	162	33	25	10	0	0
21	455	455	130	130	85	20	25	0	0	0
22	455	455	0	0	145	20	25	0	0	0
23	455	425	0	0	0	20	0	0	0	0
24	455	345	0	0	0	0	0	0	0	0

6.6. Large-Scale Unit Commitment Considering Ramp Rate Limits

The QEA-UC method is now further extended to solve a UC problem with ramp-rate limits. The ramp-rate limits can be easily handled using the ramp-rate limit handling method in [33] by constraining all on-line units to operate within their feasible output limits. The details can be found in [33]. In this study, 100 units are scheduled over a horizon of 7 days. The up-ramp/down-ramp limits of unit 1 to unit 10 of the 10-unit system [11] are set to 160, 160, 100, 100, 100, 60, 60, 40, 40 and 40 in MW respectively. The same limiting values for the corresponding units are assumed in the 100-unit study case. The population size and maximum number of generations of the proposed method are 18 and 200.

Table 6.14 and Fig. 6.9 present the scheduling results obtained either with or without unit ramp-rate limits. It is obvious that the generating cost is increased due to the incorporation of unit ramp-rate characteristics in the UC problem. The effectiveness of the proposed method is again observed in terms of the speed of convergence and the computational time. The results illustrate the effectiveness and the feasibility of the proposed QEA algorithm in solving large-scale UC problems with practical and complicated constraints.

Table 6.14 Results of UC with seven days and 100 units

Ramp-rate limits	Mean Time (s)	Cost ($)			
		Best	Mean	Worst	S.D.
Without ramp-rate limits	582	39,294,084	39,309,631	39,318,478	9,155
With ramp-rate limits	610	39,296,686	39,313,231	39,324,428	10,111

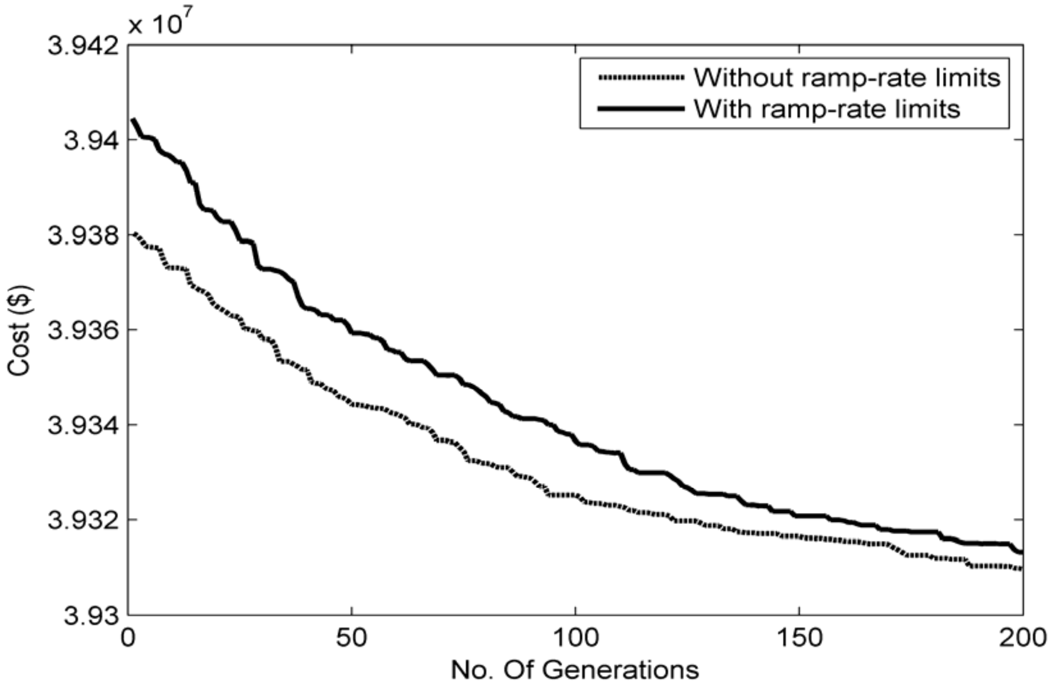

Fig. 6.9 Average cost convergence curves of the proposed QEA-UC method for the UC problem considering 7 days with/without ramp-rate limits

6.7. Bi-Objective Unit Commitment with Modified QEA-UC

In this Section, the QEA-UC method is modified to solve the bi-objective UC. Two trade-off curves are constructed by the modified QEA-UC with random order list and cost-based priority order. The effectiveness of the modified QEA-UC with different order lists is compared and analyzed. Moreover, the effects on generator outputs considering best emission and best cost are given.

The test system is derived from the 10-unit system [11] and the emission coefficients [59]. The emission coefficients of the test system are shown in Table 6.15.

Table 6.15-i) Emission coefficients of generating units

	Emission Coefficients of Generation Units				
	1	**2**	**3**	**4**	**5**
e_1 **(kg)**	25.8	26.9	30.1	25.3	30.1
e_2 **(kg/MW-h)**	-0.52	-0.54	-0.49	-0.56	-0.39
e_3 **(kg/MW^2-h)**	0.007	0.007	0.004	0.004	0.004

Table 6.15-ii) Emission coefficients of generating units

	Emission Coefficients of Generation Units				
	6	**7**	**8**	**9**	**10**
e_1 **(kg)**	25.3	23.9	23.9	31.6	34.3
e_2 **(kg/MW-h)**	-0.53	-0.4	-0.4	-0.63	-0.68
e_3 **(kg/MW^2-h)**	0.004	0.008	0.008	0.004	0.004

6.7.1. Bi-Objective UC by Modified QEA-UC with Random Order List

In the UC solution, units are turned on or off based on a random order list instead of the cost-based priority list. A set of compromise solutions is obtained by varying the weighting factor from 1 to 0 with a step size of 0.02, totally 51 optimization runs. κ is set 1 because the numeric order between the emission and cost solutions obtained in the test bi-objective system are similar, 10^4 and 10^5 respectively. For each optimization run, the maximum generation and population size are set to 100 and 20. The trade-off curve is presented in Fig. 6.10.

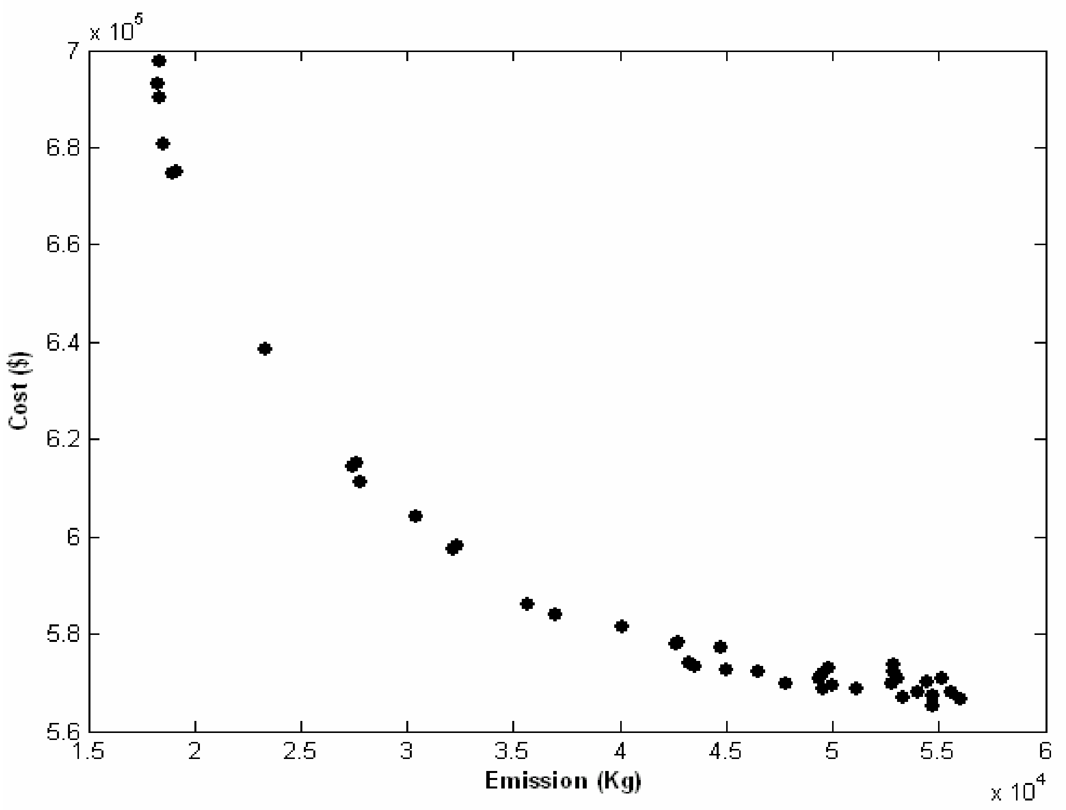

Fig. 6.10 Trade-off curve of the bi-objective UC with the modified QEA-UC considering the random order list

6.7.2. Bi-Objective UC by Modified QEA-UC with Cost-based Priority List

In the modified QEA-UC, units are turned on or off according to the cost-based priority list. A set of compromise solutions is obtained by varying the weighting factor from 1 to 0 with a step size of 0.02, κ=1, totally 51 optimization runs. For each optimization run, the maximum generation and population size are set to 100 and 20. Fig. 6.11 shows the trade-off curve.

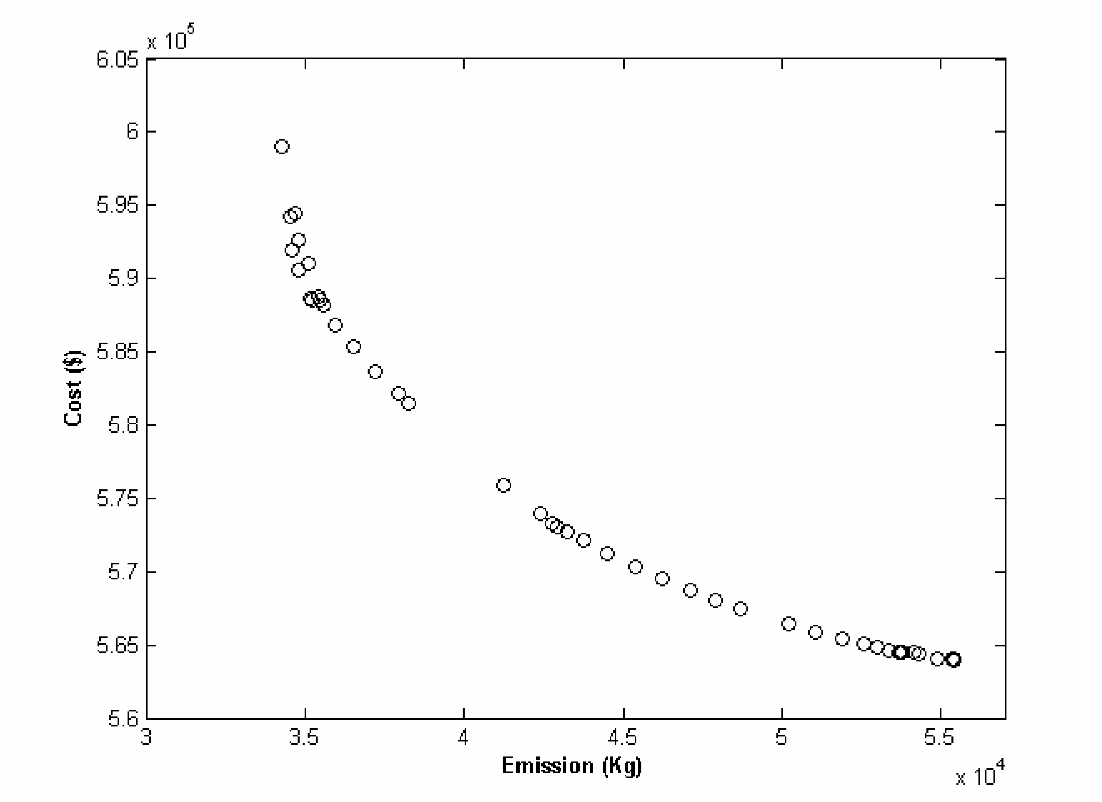

Fig. 6.11 Trade-off curve of the bi-objective UC with the modified QEA-UC considering the cost-based priority order list

6.7.3. Comparison of Bi-objective UC with Different Order List

6.7.3.1. Comparison of the Trade-off Curves

Fig. 6.12 presents a comparison of the two trade-off curves obtained by the modified QEA-UC approach with the cost-based priority list and random order list respectively. As shown in Fig. 6.12, the trade-off curve obtained by the random order list presents a cost range between $564,000 and $690,000 as well as an emission range between 19,000Kg and 55,000Kg, whereas the curve obtained by the cost-based priority list gives a cost range between $564,000 and $600,000 as well as an emission range between 35,000Kg and 55,000Kg.

Apparently, the bi-objective UC solution with the random order list gives a more widespread trade-off curve than that given by the cost-based priority list. The results show that the modified QEA-UC approach considering the cost-based priority list can only construct a partial coverage of the trade-off curve with a cost rang [$564,000, $600,000] and an emission range [35,000Kg, 55,000Kg], but it performs better than the random order list within this partial coverage for the same population size and maximum number of generations. If the emission range [35,000Kg, 55,000Kg] is acceptable to system operators when trading off emission and cost in the bi-objective UC problem, the cost-based priority list is more preferable than the random order list in terms of the quality of solutions and total function evaluation.

6.7.3.2. Comparison of the Solution Convergence

Fig. 6.12 compares the two performances given by the modified QEA-UC approach considering the cost-based priority list and random order list respectively. As compared to the trade-off curve given by the cost-based priority list, the trade-off curve obtained by the random order list presents a wider cost and emission ranges. Definitely, the bi-objective UC solution with the random order list gives a more widespread trade-off curve than that given by the cost-based priority list.

Although the results show that the cost-based priority list can only construct a partial coverage of the trade-off curve with a narrower cost and emission ranges with a cost range [$564,000, $600,000] and an emission range [35,000Kg, 55,000Kg], the cost-based priority list performs better than the random order list within this partial coverage for the parameter settings. In a cost-driven power system environment or the cost objective function with a higher weighting factor than that of the emission function, the cost-based priority list is more preferable than the random order list in terms of the quality of solutions and total function evaluation.

Also given in Fig. 6.12, in the emission range [45,000Kg, 55,000Kg], the modified QEA-UC using the cost-based priority list find better solutions than the one using the random order list. At weighting factors close to or equal to 1 and with the same population size and maximum iteration, the modified QEA-UC with the cost-based priority list converges to better solutions than that of the modified approach with the random order list.

These two comparisons imply that the cost-based priority list facilitates a fast solution convergence but gives a partial coverage of the trade-off curve, while the random order list converge slowly but give a widespread coverage of the trade-off curve.

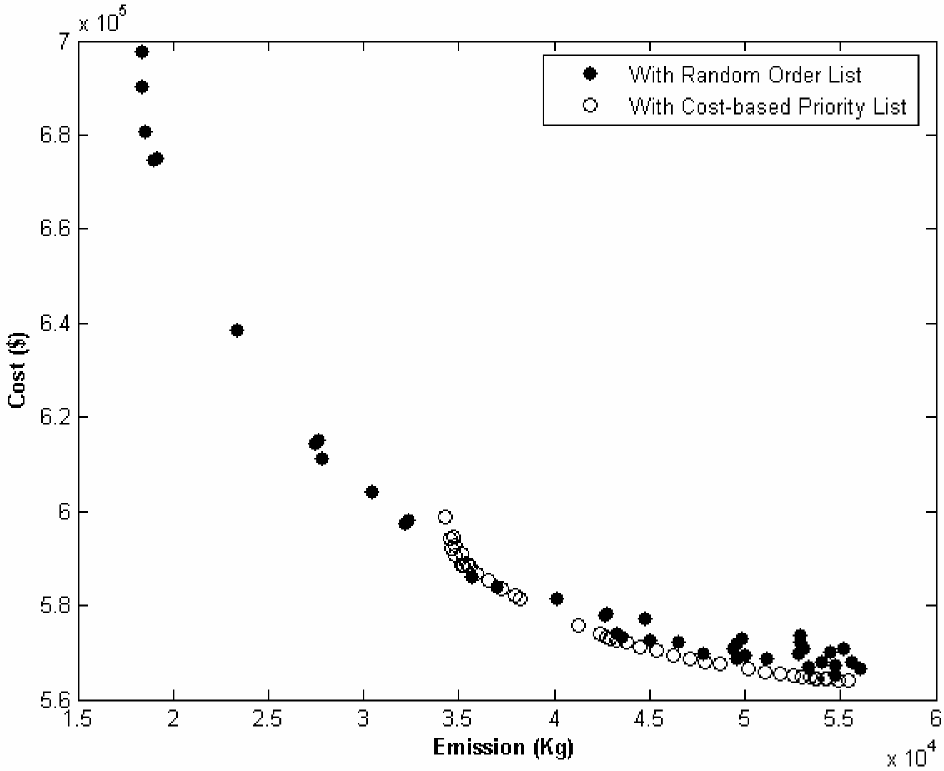

Fig. 6.12 Trade-off curves of obtained by the modified QEA-UC with the cost-based priority order list and random order list.

6.7.4. Discussions on Best Emission Solution and Best Cost Solution

The best emission solution and best cost solution can be obtained by running the modified QEA-UC method considering $w = 0$ and $w = 1$ respectively. The best emission solution and the best cost solution are tabulated in Table 6.16 to Table 6.17. In Tables 6.16 and 6.17, the two solutions present two different unit schedules and unit load dispatch. In the best emission solution, the generating units are scheduled and dispatched to produce a minimum emission so that units, such as Units 6, 9 and 10, giving cleaner outputs have higher priorities to be selected and loaded. For the best cost solution, less expensive units, such as Units 1 and 2, are preferred to be scheduled in order to minimize the total production cost.

Fig. 6.13 and Fig. 6.14 illustrate the unit scheduling and unit outputs of the two best solutions in a graphical manner.

Table 6.16 Best emission solution

Hour	Unit Output (MW)									
	1	2	3	4	5	6	7	8	9	10
1	150	150	0	116	94	80	0	0	55	55
2	150	150	106	115	94	80	0	0	55	0
3	150	150	94	103	81	80	41	41	55	55
4	150	150	119	128	106	80	54	54	55	55
5	150	150	130	130	130	80	65	55	55	55
6	173	175	130	130	162	80	85	55	55	55
7	198	200	130	130	162	80	85	55	55	55
8	223	225	130	130	162	80	85	55	55	55
9	273	275	130	130	162	80	85	55	55	55
10	323	325	130	130	162	80	85	55	55	55
11	348	350	130	130	162	80	85	55	55	55
12	373	375	130	130	162	80	85	55	55	55
13	323	325	130	130	162	80	85	55	55	55
14	273	275	130	130	162	80	85	55	55	55
15	223	225	130	130	162	80	85	55	55	55
16	150	150	130	130	162	80	83	55	55	55
17	150	150	130	130	130	80	65	55	55	55
18	173	175	130	130	162	80	85	55	55	55
19	223	225	130	130	162	80	85	55	55	55
20	323	325	130	130	162	80	85	55	55	55
21	273	275	130	130	162	80	85	55	55	55
22	173	175	130	130	162	80	85	55	55	55
23	0	150	130	130	162	80	83	55	55	55
24	0	150	119	128	106	80	54	54	55	55

Table 6.17 Best cost solution

Hour	Generating Unit									
	1	2	3	4	5	6	7	8	9	10
1	455	245	0	0	0	0	0	0	0	0
2	455	295	0	0	0	0	0	0	0	0
3	455	370	0	0	25	0	0	0	0	0
4	455	455	0	0	40	0	0	0	0	0
5	455	390	0	130	25	0	0	0	0	0
6	455	360	130	130	25	0	0	0	0	0
7	455	410	130	130	25	0	0	0	0	0
8	455	455	130	130	30	0	0	0	0	0
9	455	455	130	130	85	20	25	0	0	0
10	455	455	130	130	162	33	25	10	0	0
11	455	455	130	130	162	73	25	10	10	0
12	455	455	130	130	162	80	25	43	10	10
13	455	455	130	130	162	33	25	10	0	0
14	455	455	130	130	85	20	25	0	0	0
15	455	455	130	130	30	0	0	0	0	0
16	455	310	130	130	25	0	0	0	0	0
17	455	260	130	130	25	0	0	0	0	0
18	455	360	130	130	25	0	0	0	0	0
19	455	455	130	130	30	0	0	0	0	0
20	455	455	130	130	162	33	25	10	0	0
21	455	455	130	130	85	20	25	0	0	0
22	455	455	0	0	145	20	25	0	0	0
23	455	425	0	0	0	20	0	0	0	0
24	455	345	0	0	0	0	0	0	0	0

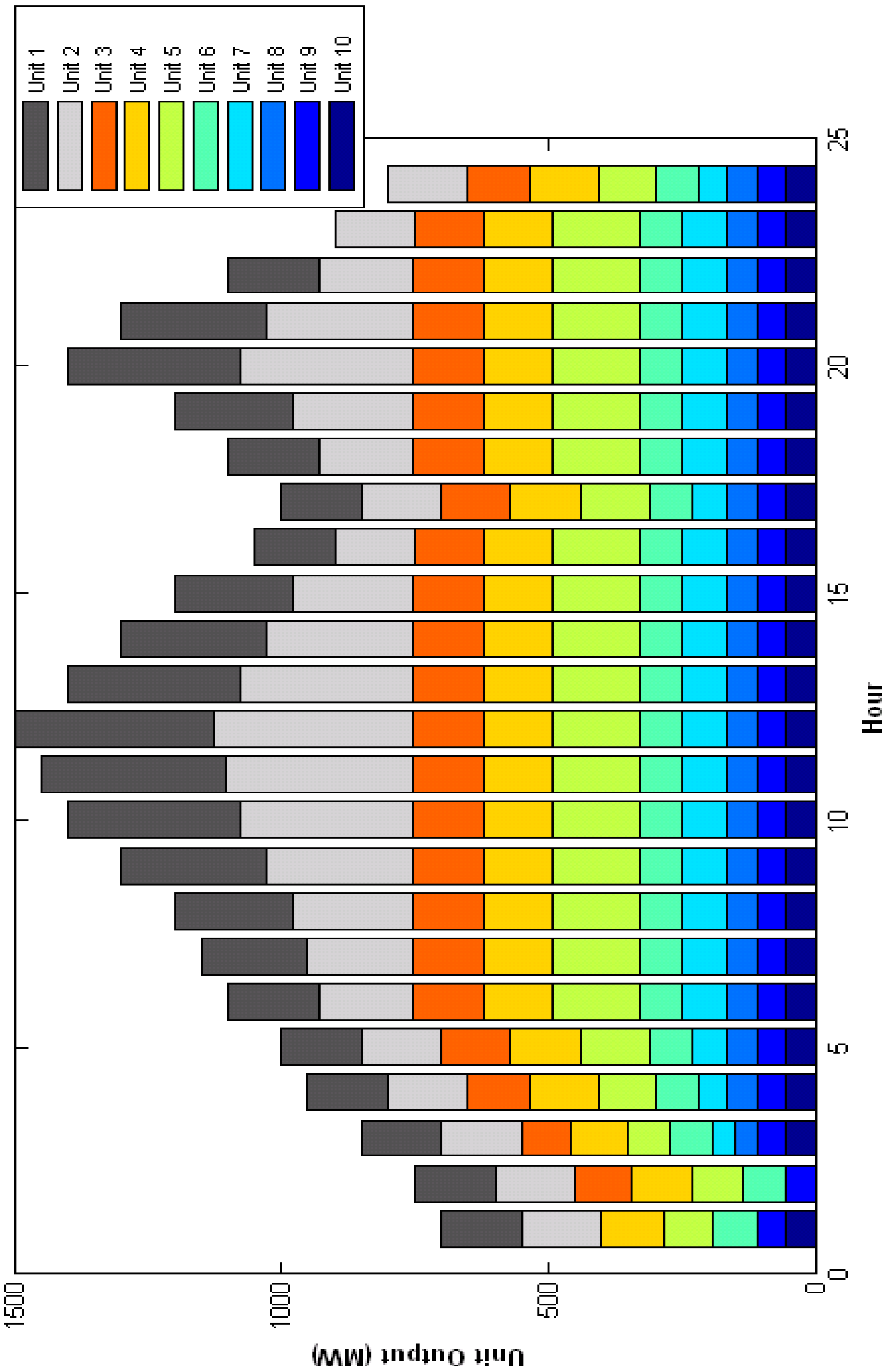

Fig. 6.13 Generator outputs of the best emission solution

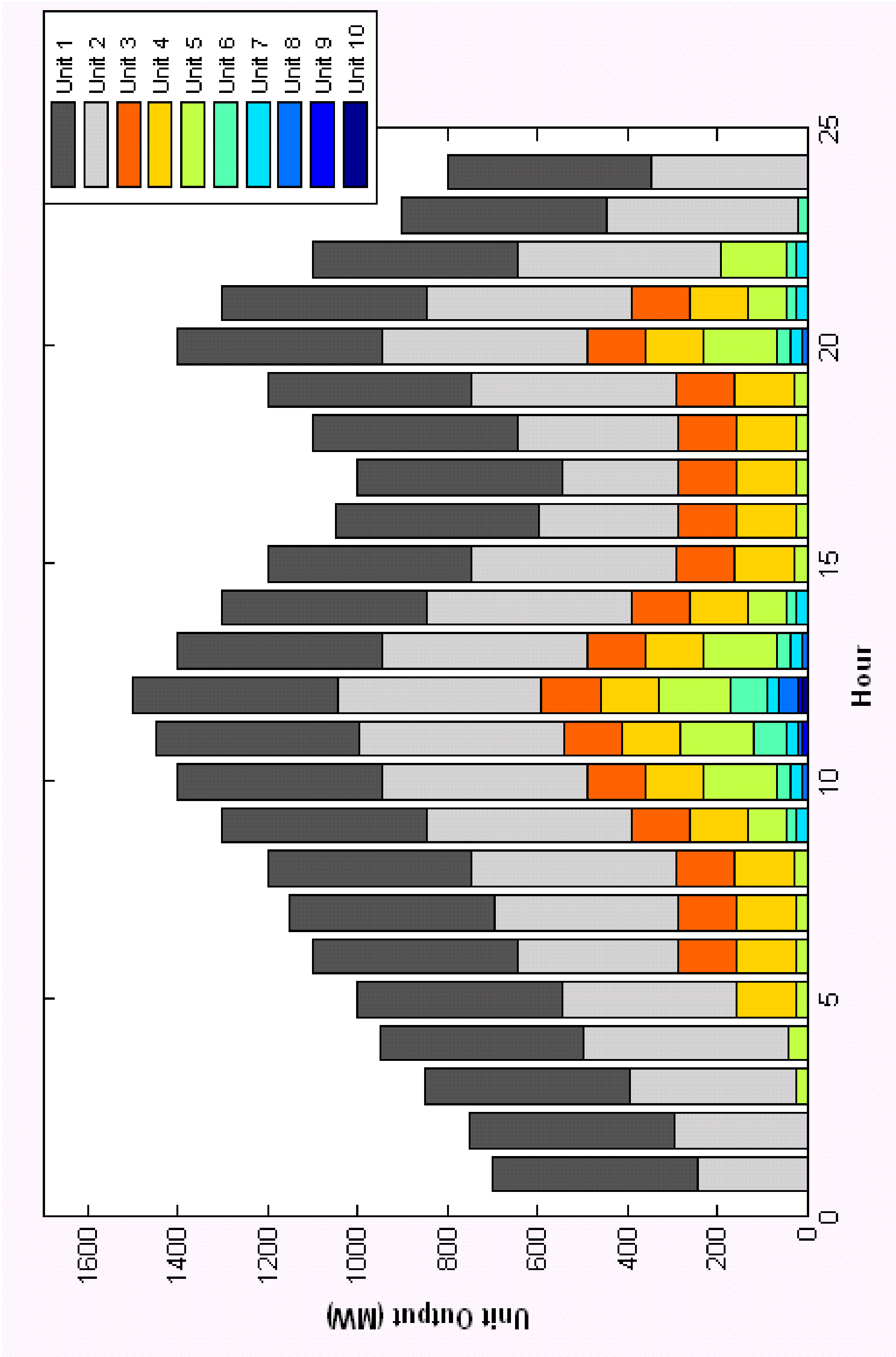

Fig. 6.14 Generator outputs of the best cost solution

Chapter 7 Conclusions and Future Work

7.1. Conclusions

This thesis has successfully introduced a novel optimization technique based on the quantum-inspired evolutionary algorithm for solving the unit commitment problem (QEA-UC). The effectiveness and feasibility of the QEA-UC algorithm have been demonstrated through its applications to test systems with the number of units from 10 to 100. It has been shown that the QEA-UC algorithm is very powerful and efficient and it outperforms many other existing methods. The QEA-UC algorithm can perform well even with a small population size, and it has been found to have a linear relationship between the scale of the UC problem and computational time. Moreover, the proposed algorithm has been successfully applied to solve a large-scale UC problem in which 100 units have been scheduled over a 7-day horizon with unit ramp-rate limits considered. The QEA-UC algorithm is therefore very promising to be applied to large-scale UC problems.

The proposed QEA-UC has also been modified to solve the bi-objective UC problem considering both cost and emission objective functions, 24-hour horizon and 10 generating units. Two listing methods, namely cost-based priority list and random order list, has been proposed and tested. The performances of these tow list methods have been discussed. The simulation results have demonstrated that the QEA-UC approach also is promising to deal with the bi-objective UC problem.

7.2. Future Work

In this thesis, the proposed QEA-UC has given a very promising capability of solving the UC problem in terms of its effectiveness and efficiency. In the future, the QEA-UC can be adopted to deal with practical UC problems considering other problem features, including non-smooth fuel cost functions, network security constraints, stability constraints, and fuel consumption constraints. For instance, when the UC problem involves non-smooth cost functions, the economic dispatch problem cannot be simply calculated by the Lambda iteration method.

Emission performance in power utilities becomes more and more significant. An efficient and effective optimization approach for solving multi-objective UC problems is useful for power system operators to trade off emission, cost and other objectives in real-time plant scheduling. The proposed method can be further extended to cope with real-time multi-objective UC problems.

In power systems, the application of QEA is novel and very promising. Future work can also consider applications of QEA to handle other power system optimization problems, such as economic dispatch, reactive power dispatch, capacitor bank placement, optimal power flow, and power system planning.

References

[1] R. M. Burns and C. A. Gibson, “Optimization of priority lists for a unit commitment program,” in Proc. IEEE/Power Engineering Society Summer Meeting, Paper A 75 453-1, 1975.

[2] W. L. Snyder Jr., H. D. Powell Jr., and J. C. Rayburn, “Dynamic programming approach to unit commitment,” IEEE Trans. Power App. Syst., vol.PAS-2, pp. 339–350, May 1987.

[3] P. G. Lowery, “Generation unit commitment by dynamic programming,” IEEE Trans. Power App. Syst., vol. PAS-102, pp. 1218–1225, 1983.

[4] C. K. Pang, G. B. Sheble, and F. Albuyeh, “Evaluation of dynamic programming based methods and multiple area representation for thermal unit commitment,” IEEE Trans. Power App. Syst., vol. PAS-100, no. 3, pp. 1212–1218, May 1981.

[5] H. H. Balci and J. F. Valenzuela, “Scheduling electric power generators using particle swarm optimization combined with the Lagrangian relaxation method,” Int. J. Appl. Math. Comput. Sci., vol. 14, no. 3, pp. 411–421, 2004.

[6] F. Zhuang and F. D. Galiana, “Toward a more rigorous and practical unit commitment by Lagrangian relaxation,” IEEE Trans. Power Syst., vol.3, no. 2, pp. 763–770, May 1988.

[7] J. A. Muckstadt and R. C. Wilson, “An application of mixed-integer programming duality to scheduling thermal generating systems,” IEEE Trans. Power App. Syst., pp. 1968–1978, 1968.

[8] A. I. Cohen and M. Yoshimura, “A branch-and-bound algorithm for unit commitment,” IEEE Trans. Power App. Syst., vol. PAS-102, no. 2, pp.444–451, 1983.

[9] K.P. Wong and K. Doan, “An artificial intelligence algorithm for daily scheduling of thermal generators”, IEE Proc. Part C, 1991, 138, (6), pp. 518-534.

[10] E. Tsoi and K.P. Wong, “Artificial intelligence algorithms for short-term scheduling of thermal generators and pumped storage”, IEE Proc. Gen. Trans. and Distrib., 144, (2), 1997, pp. 193-200.

[11] S. A. Kazarlis, A. G. Bakirtzis, and V. Petridis, “A genetic algorithm solution to the unit commitment problem,” IEEE Trans. Power Syst., vol.11, no. 1, pp. 83–92, Feb. 1996.

[12] P.C. Yang, H.T. Yang, and C.-L. Huang, “Solving the unit commitment problem with a genetic algorithm through a constraint satisfaction technique,” Elect. Power Syst. Res., vol. 37, pp. 55–65, 1996.

[13] F. Zhuang and F. D. Galliana, “Unit commitment by simulated annealing,” IEEE Trans. Power Syst., vol. 5, no. 1, pp. 311–318, Feb.1990.

[14] A. H. Mantawy, Y. L. Abdel-Magid, and S. Z. Selim, “A simulated annealing algorithm for unit commitment,” IEEE Trans. Power Syst., vol.13, no. 1, pp. 197–204, Feb. 1998.

[15] A. Viana, J. P. Sousa, and M. Matos, “Simulated annealing for the unit commitment problem,” presented at the IEEE Porto Power Tech Conf., Porto, Portugal, Sept. 10–13, 2001.

[16] Simopoulos, D.N. Kavatza, S.D. Vournas, C.D., "Unit commitment by an enhanced simulated annealing algorithm", IEEE Trans. Power Syst., vol.21, no. 1, pp. 193-201, Feb. 2006.

[17] K. A. Juste, H. Kita, E. Tanaka, and J. Hasegawa, "An evolutionary programming solution to the unit commitment problem," IEEE Trans. Power Syst., vol. 14, pp. 1452–1459, Nov. 1999.

[18] T. O. Ting, M. V. C. Rao, and C. K. Loo, "A novel approach for unit commitment problem via an effective hybrid particle swarm optimization," IEEE Trans. Power Syst., vol. 21, no. 1, pp. 411–418, Feb. 2006.

[19] G. K. Purushothama and L. Jenkins, "Simulated annealing with local search—A hybrid algorithm for unit commitment," IEEE Trans. Power Syst., vol. 18, no. 1, pp. 273–278, Feb. 2003.

[20] T. Senjyu, H. Yamashiro, K. Uezato, and T. Funabashi, "A unit commitment problem by using genetic algorithm based on unit characteristic classification," in Proc. 2002 IEEE Power Eng. Soc. Winter Meeting, vol. 1, pp. 58–63.

[21] K.P. Wong and C.C. Fung, "Simulated-annealing-based economic dispatch algorithm", IEE Proc. Part C, 1993, 140, (6), pp. 509-515.

[22] K.P. Wong and Y.W. Wong, "Genetic and genetic/simulated-annealing approaches to economic dispatch", IEE Proc., Gener. Transm. Distrib., 1994, 141, (5), pp.507-513.

[23] K.P. Wong and J. Yuryevich, "Evolutionary-programming-based algorithm for environmentally-constrained economic dispatch", IEEE Trans. on Power Systems, Vol.13, No.2, 1998, pp.301-306.

[24] K.P. Wong and Y.W. Wong, "Thermal generator scheduling using hybrid genetic/simulated-annealing approach", IEE Proc. Gener. Transm. Distrib., 142, (4), 1995, pp. 372-380.

[25] K.P. Wong and S.Y.W. Wong, "Combined genetic algorithm/simulated annealing/fuzzy set approach to short-term generation scheduling with take-or-pay fuel contract", IEEE Trans. on Power Systems, Vol. 11, No. 1, 1996, pp.112-118.

[26] K.P. Wong and S.Y.W. Wong, ”Hybrid genetic/simulated annealing approach to short-term multiple-fuel-constrained generation scheduling”, IEEE Trans. on Power Systems, Vol. 12, No. 2, 1997, pp. 776- 784.

[27] A. Narayanan and M. Moore, “Quantum-inspired genetic algorithms,” in Proc. 1996 IEEE Int. Conf. Evolutionary Computation. Piscataway, NJ: IEEE Press, May 1996, pp. 61–66.

[28] A. Narayanan, “Quantum computing for beginners,” in Proc. 1999 Congress on Evolutionary Computation. Piscataway, NJ: IEEE Press, July 1999, vol. 3, pp. 2231–2238.

[29] K.H. Han and J.H. Kim, “Genetic quantum algorithm and its application to combinatorial optimization problem,” in Proc. 2000 Congress on Evolutionary Computation. Piscataway, NJ: IEEE Press, July 2000, vol. 2, pp. 1354–1360.

[30] K.H. Han and J.H. Kim, “Quantum-inspired evolutionary algorithm for a class of combinatorial optimization,” IEEE Trans. Evol. Comput, vol. 6, pp. 580–593, Dec. 2002.

[31] K.H. Han, K.H. Park, C.H. Lee, and J.H. Kim, "Parallel quantum-inspired genetic algorithm for combinatorial optimization problem," in Proc. 2001 Congr. Evolutionary Computation, vol. 2, Seoul, Korea, May 2001, pp. 1422–1429.

[32] K.H. Han and J.H. Kim, "On setting the parameters of quantum-inspired evolutionary algorithm for practical applications," in Proc. 2003 Congr. Evolutionary Computation, Canberra, Australia, Dec. 2003, pp.178–184.

[33] C. Wang and S. M. Shahidehpour, "Effects of ramp-rate limits on unit commitment and economic dispatch," IEEE Trans. Power Syst., vol. 8, no. 3, pp.1341-1350, Aug. 1993.

[34] H. H. Balci and J. F. Valenzuela, "Scheduling electric power generators using particle swarm optimization combined with the Lagrangian relaxation method," Int. J. Appl. Math. Comput. Sci., vol. 14, no. 3, pp. 411–421, 2004.

[35] J. Kennedy and R. C. Eberhart, "A discrete binary version of the particle swarm algorithm," in Proc. Int. Conf. Systems, Man, Cybernetics. Piscataway, NJ, 1997, pp. 4104–4109.

[36] L. J. Fogel, A. J. Owens, and M. J. Walsh, “Artificial Intelligence through Simulated Evolution,” New York John Wiley & Sons, Inc., 1966.

[37] J. H. Holland, “Adqmtion in Natural and Artificial System,” 2nd ed., Cambridge, MA: MIT Press. 1992.

[38] Kirkpatrick, S., Gelatt, C.D., and Vecchi, M.P., “Optimization by simulated annealing,” Science, 1983, 220, (4598), pp. 671-680.

[39] M. A. Abido, “Environmental/economic power dispatch using multiobjective evolutionary algorithms”, IEEE Trans. Power Syst., Vol. 18, No. 4, pp 1529-1537, Nov. 2003.

[40] M. Muslu, “Economic dispatch with environmental considerations: Tradeoff curves and emission reduction rates”, Electric Power Systems Research, Vol. 71, No. 2, pp 153-158, Oct. 2004.

[41] C. M. Huang, H. T. Yang, and C. L. Huang, “Bi-Objective power dispatch using fuzzy satisfaction-maximizing decision approach,” IEEE Trans. Power Syst., vol. 12, pp. 1715–1721, Nov. 1997.

[42] M. A. Abido, “A new multiobjective evolutionary algorithm for environmental/ economic power dispatch,” in IEEE Summer Meeting, Vancouver, BC, Canada, July 15–19, 2001.

[43] J. H. Talaq, F. El-Hawary, and M. E. El-Hawary, “A summary of environmental/economic dispatch algorithms,” IEEE Trans. Power Syst., vol. 9, pp. 1508–1516, Aug. 1994.

[44] J. S. Helsin and B. F. Hobbs, “A multiobjective production costing model for analyzing emission dispatching and fuel switching,” IEEE Trans. Power Syst., vol. 4, pp. 836–842, Aug. 1989.

[45] C. Huang and Y. Huang, “A novel approach to real-time economic emission power dispatch”, IEEE Transactions on Power Systems, Vol. 18, No. 1, pp 288-294, February 2003.

[46] J. S. Dhillon and D. P. Kothari, “The surrogate worth trade-off approach for multiobjective thermal power dispatch problem”, Electric Power Systems Research, Vol. 56, No. 2, pp 103-110, November 2000.

[47] A. A. El-Keib, H. Ma, and J. L. Hart, “Economic dispatch in view of the clean air act of 1990,” IEEE Trans. Power Syst., vol. 9, pp. 972–978, May 1994.

[48] J. H. Talaq, F. El-Hawary, and M. E. El-Hawary, “A summary of environmental/economic dispatch algorithms,” IEEE Trans. Power Syst.,vol. 9, pp. 1508–1516, Aug. 1994.

[49] J. S. Helsin and B. F. Hobbs, “A multiobjective production costing model for analyzing emission dispatching and fuel switching,” IEEE Trans. Power Syst., vol. 4, pp. 836–842, Aug. 1989.

[50] G. P. Granelli, M. Montagna, G. L. Pasini, and P. Marannino, “Emission constrained dynamic dispatch,” Electric Power Syst. Res., vol. 24, pp. 56–64, 1992.

[51] A. Farag, S. Al-Baiyat, and T. C. Cheng, “Economic load dispatch multiobjective optimization procedures using linear programming techniques,” IEEE Trans. Power Syst., vol. 10, pp. 731–738, May 1995.

[52] J. S. Dhillon, S. C. Parti, and D. P. Kothari, “Stochastic economic emission load dispatch,” Electric Power Syst. Res., vol. 26, pp. 186–197, 1993.

[53] C. S. Chang, K. P. Wong, and B. Fan, “Security-Constrained multiobjective generation dispatch using bicriterion global optimization,” Proc.Inst. Elect. Eng.-Gen. Transm. Dist., vol. 142, no. 4, pp. 406–414, 1995.

[54] R. Yokoyama, S. H. Bae, T. Morita, and H. Sasaki, "Multiobjective generation dispatch based on probability security criteria," IEEE Trans. Power Syst., vol. 3, pp. 317–324, Feb. 1988.

[55] C. M. Huang, H. T. Yang, and C. L. Huang, "Bi-Objective power dispatch using fuzzy satisfaction-maximizing decision approach," IEEETrans. Power Syst., vol. 12, pp. 1715–1721, Nov. 1997.

[56] D. B. Das and C. Patvardhan, "New multi-objective stochastic search technique for economic load dispatch," Proc. Inst. Elect. Eng.-Gen. Transm. Dist., vol. 145, no. 6, pp. 747–752, 1998.

[57] M. A. Abido, "A new multiobjective evolutionary algorithm for environmental economic power dispatch," in IEEE Summer Meeting, Vancouver, BC, Canada, July 15–19, 2001.

[58] D. C.Walters and G. B. Sheble, "Genetic algorithm solution of economic dispatch with valve point loading," IEEE Trans. Power Syst., vol. 8, pp. 1325–1332, Aug. 1993.

[59] João Catalão, Sílvio Mariano, Victor Mendes, Luís Ferreira, "Unit commitment with environmental considerations: A practical approach, " 15th PSCC, Liege, Session 18, Paper 3, pp.1-5, 22-26 Aug. 2005.

[60] D. Srinivasan, C. S. Chang, and A. C. Liew, "Multiobjective generation schedule using fuzzy optimal search technique," Proc. Inst. Elect. Eng.-Gen. Transm. Dist., vol. 141, pp. 231–241, 1994.

[61] T. Gjengedal, "Emission constrained unit-commitment (ECUC)", IEEE Transactions on Energy Conversion, Vol. 11, No. 1, pp 132-138, March 1996.

[62] V. M. F. Mendes, S. J. P. S. Mariano, J. P. S. Catalão and L. A. F. M. Ferreira, "Emission constraints on short-term schedule of thermal units", 39th UPEC Proceedings, pp 1068-1072, September 2004.