

Research Article

Preventive Maintenance Scheduling for Multigeneration Plants with Production Constraints Using Genetic Algorithms

Khaled Alhamad,¹ Mohsen Alardhi,² and Abdulla Almazrouee³

¹*Technological Science Department, College of Technological Studies, PAAET, P.O. Box 42325, Shuwaikh 70654, Kuwait*

²*Automotive & Marine Engineering Technology Department, College of Technological Studies, PAAET, P.O. Box 42325, Shuwaikh 70654, Kuwait*

³*Manufacturing Engineering Technology Department, College of Technological Studies, PAAET, P.O. Box 42325, Shuwaikh 70654, Kuwait*

Correspondence should be addressed to Khaled Alhamad; km.alhamad@paaet.edu.kw

Received 13 August 2014; Revised 20 November 2014; Accepted 9 January 2015

Academic Editor: Viliam Makis

Copyright © 2015 Khaled Alhamad et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper describes a method developed to schedule the preventive maintenance tasks of the generation and desalination units in separate and linked cogeneration plants provided that all the necessary maintenance and production constraints are satisfied. The proposed methodology is used to generate two preventing maintenance schedules, one for electricity and the other for distiller. Two types of crossover operators were adopted, 2-point and 4-point. The objective function of the model is to maximize the available number of operational units in each plant. The results obtained were satisfying the problem parameters. However, 4-point slightly produce better solution than 2-point ones for both electricity and water distiller. The performance as well as the effectiveness of the genetic algorithm in solving preventive maintenance scheduling is applied and tested on a real system of 21 units for electricity and 21 units for water. The results presented here show a great potential for utility applications for effective energy management over a time horizon of 52 weeks. The model presented is an effective decision tool that optimizes the solution of the maintenance scheduling problem for cogeneration plants under maintenance and production constraints.

1. Introduction

Desalination, desalinization, desalinisation, or desalting refers to any of several processes employed to produce fresh water suitable for human consumption by removing some amount of salt from the sea water. It is known that salty water cannot sustain the human life, farming, or industry; therefore, people have strived to develop cost-effective and reliable ways for desalination. One of the widely used procedures for water production is cogeneration. This is an energy-efficient method of producing power and fresh water at the same time, in one process with one fuel. The cogeneration power plants (considered as complex process engineering systems consisting of various subsystems) are operated for achieving two major goals, namely, maximizing both generation of power and production of distilled water. The cogeneration power plants have been widely used worldwide because

of their reliability, efficiency, and economic values [1–3]. Currently, around 120 countries have adopted this technology for producing power and water [4].

The cost effectiveness represents a real challenge that the cogeneration power plants technology has been facing recently. This effectiveness is related to the reduction of the different cost variables such as fuel and operational cost.

In the classical scheduling, the common assumption is that processing times of the jobs are fixed and known a priori with certainty [5]. In practice, however, machine production rate becomes less than normal due to a mal-position of tools, mal-alignment jobs, abrasion of tools, and scraps of operations, and so forth. In these situations, the processing time of the jobs increases depending upon the sequence of jobs or the starting time of jobs. Such problems are generally known as machine scheduling problems with deterioration [6]. The deteriorated processing time of jobs is recovered to

an original processing time by the maintenance or cleaning process machines. The recovering process bringing back to a normal processing time is called rate-modifying activity (RMA) [7]. Lee and Leon [8] consider several problems of scheduling on a single machine with an RMA (minimizing makespan, flow-time, weighted flow-time, and maximum lateness).

The maintenance, which is defined as a combination of several functions needed to keep in or restore the equipment to a designated operational state [9–11], is an essential activity in these cogeneration power plants. It is known as an increasing portion of the operational costs and as a result any saving in maintenance cost will lead to huge reduction in the overall cost. Maintenance also has a great effect on the plant and unit availability and production cost. In addition, the long term maintenance scheduling (MS) of the generation and desalination units is an important challenge for these plants. Past experiences indicated that effective scheduling can save considerable operational costs that help the electric and water utilities to be more competitive in terms of the energy and water price while increasing system reliability [2, 3].

Maintenance optimization consists in broad terms of those mathematical models aimed at finding either the optimum balance between costs and benefits of maintenance or the most appropriate moment to execute maintenance [12]. It is a well-established area as several reviews show [13, 14]. For introductions and frameworks, see [15, 16]. For recent problem oriented overviews, see [17–19].

Maintenance scheduling is usually considered as an optimization problem. This problem is subject to a number of constraints such as the crew constraint, maintenance window constraint, time limitation constraint, and the output load constraints of water and electric power demanded [9]. The main goal of this optimization problem is to allow the power generation and water production to meet the demand without any delay by keeping the required equipment available during operation. Availability is “the ability of equipment to successfully perform its required function at a stated instant of time over a stated period of time” [20]. Any failure which may occur in any unit in the cogeneration plant will affect the overall availability of the system and therefore will cause an increase in the overall cost.

As an example, some earlier studies that dealt with cogeneration maintenance are [21–23]. Authors treating desalination include [1, 2, 24, 25]. Several methods have been proposed in the literature to solve thermal unit MS; these have similarities with cogeneration units. The solution methods can be categorized as follows: integer programming [1, 26, 27], Benders’ decomposition technique [28], dynamic programming [11, 29], and exact and metaheuristic algorithms [30], which give exact optimal solution. However, when the size of the problem increases the solution space increases greatly as a result causing the computational time to increase. Therefore, the exact solution might be sacrificed in order to reduce the computational time using other algorithms such as Tabu search [31], simulated annealing methods [32], and genetic algorithm (GA) [33, 34].

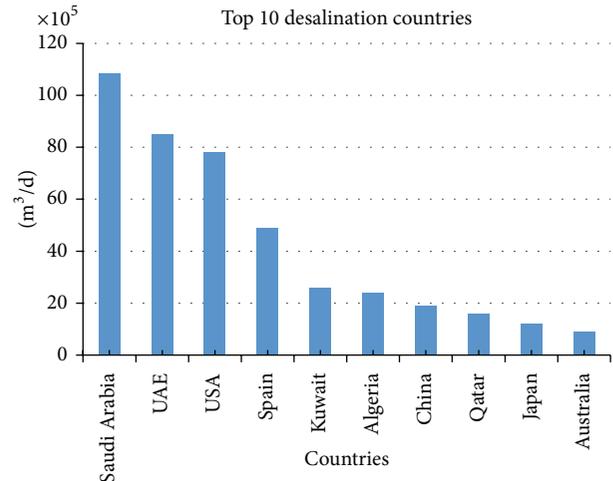


FIGURE 1: Desalination capacity for top 10 countries in terms of installed capacities in m^3/d (2008) [35].

This paper presents a genetic algorithm (GA) model designed to produce an optimal maintenance schedule for cogeneration plants in terms of maximizing the available number of units in each plant for a 12-month demand cycle. GA was used in many other problems and found to produce competitive results; see, for example, Liu et al. [34] and Al-Hamad et al. [33]. The model designed in this paper is reliable and capable of generating a good schedule for industrial sectors. Moreover, this model can be used to solve large-scale problems practically and efficiently.

The structure of the paper is as follows. In the next section, the problem background and the main features of the genetic algorithm are covered. Section 3 is devoted to formulate the model mathematically. An illustrative example is given in Section 5 to test the model. This is finally followed by the conclusion.

2. Problem Background and Genetic Algorithms

2.1. Problem Background. Around 120 countries have adopted the cogeneration power plants for producing power and water, (see Figure 1). Fifty percent of them are the Middle East region [36]. Among these Middle Eastern countries that have been extensively using the cogeneration power plant is the state of Kuwait.

In Kuwait, the Ministry of Electricity and Water (MEW) is responsible for supplying power and water to the population. The MEW has met the growth in demand for electricity and water in the last 30 years by the construction of a system of cogeneration plants, exceeding 12 Gigawatts (GW) and 400 million imperial gallons per day (MIGD) [37]. There are 6 power plants in Kuwait, namely: Doha East, Doha West, Shuaiba, Al-Zour South, Sabiya, and Shuwaikh. These plants employ both thermal steam turbines and gas turbines for power generation. These plants are identical in layout but different in sizes and installed capacities. Each plant has

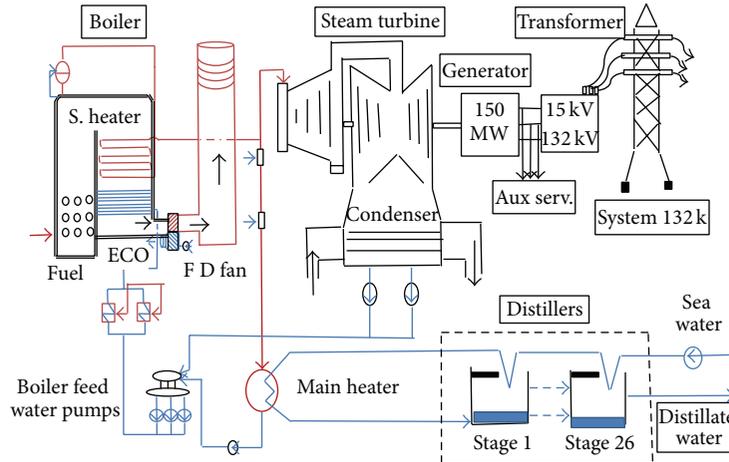


FIGURE 2: The elements of the units of a typical cogeneration plant in Kuwait.

a boiler, a turbo-generator, and a distiller as shown in Figure 2. The boilers produce a high pressure steam from fuel; the turbo-generators produce an electric power from the high pressure steam and the distillers produce desalinated water from the salty water using low pressure steam. A common steam header connects the boilers of a plant so that the steam from a boiler in one unit can be used by a distiller in another unit. The water produced is stored in reservoirs which are used to meet the predicted demand at each time period of planning horizon. The plants should also satisfy the demand for electric power and water during each time period.

The main target of this research is to present a genetic algorithm model that produces a solution of an optimal maintenance schedule for cogeneration plants in terms of maximizing the available number of units in each plant for a 12-month demand cycle.

2.2. Genetic Algorithm Definition. Genetic algorithms (GAs) belong to a class of efficient algorithms which generate solutions to optimization problems using techniques inspired by natural evolution, such as inheritance, mutation, selection, and crossover. They are very common algorithms and work well in any search space. GAs were formally introduced in the United States in the 1970s by John Holland at University of Michigan. The starting point of GA is to create a population of chromosomes (i.e., a collection of chromosomes) where each chromosome represents a proposed solution to the problem into consideration and is defined as a simple string. Two chromosomes are selected from the population and the crossover and mutation operators are then applied to generate new chromosomes. Crossover is a genetic operator that combines (mates) two chromosomes (parents) to produce a new chromosome (offspring). The idea behind crossover is that the new chromosome may be better than both of the parents if it takes the best characteristics from each of the parents. Mutation operator alters one or more gene values in a chromosome from its initial state. In mutation, the solution may change entirely from the previous solution. Hence GA can produce better solutions by using mutation. The purpose

of mutation in GAs is to preserve the diversity and to allow the algorithm to avoid local minima or maxima. In each generation, the fitness function of every chromosome in the population is evaluated. (Note that the fitness function measures the quality of the represented solution—chromosome.)

The three most important aspects of using genetic algorithms are (1) definition of the objective function, (2) definition and implementation of the genetic representation of chromosomes, and (3) definition and implementation of the genetic operators. There are various selection criteria so that it picks the best individuals for mating (see [38]).

3. Problem Description and Mathematical Formulation

The problem of MS that concerns this paper is defined formally as follows. Let P be a set of plants with $P = \{1, \dots, p\}$ where p is the total number of plants. There is a set of t units of type boiler-turbine and a set of d units of type distiller such that the total number of type boiler-turbine is equal to the total number of type distiller. Each unit must go through a maintenance process for a period of time; more than one unit can go through maintenance simultaneously. The maintenance durations of the units are not necessarily identical. During maintenance, a minimum number of units for each plant must stay in operation. The goal is to schedule the preventive maintenance tasks for p plants consisting of g units of boiler-turbine and g units of distiller in a way that maximizes the number of available units over the operational planning time-horizon T .

The schedule of maintenance consists of a strip of all the units and of their corresponding maintenance times. The schedule displays all the units and their maintenance starting week. Figure 3 illustrates the schedule of 10 units to be maintained in time-horizon of 52 weeks. The upper numbers represent the unit number and the lower numbers represent the starting week of maintenance.

For example, unit 1 starts in week 5 and unit 2 starts in week 12. Units 4 and 9 start in the same week, 24.

1	2	3	4	5	6	7	8	9	10
5	12	16	24	50	4	44	36	24	29

FIGURE 3: Schedule of 10 units to be maintained in time-horizon of 52 weeks.

1	2	3	4	5	6	7	8	9	10	Fitness
5	12	16	24	50	4	44	36	24	29	5.34

FIGURE 4: An illustration of a chromosome.

The GA adapted in this paper starts by generating a number of chromosomes, NM say, where each chromosome is formed of ND genes (nodes). Each chromosome represents a schedule of starting maintenance (week)—same Figure 3—for unit t in case of boiler-turbine, or for unit d in case of distiller. The fitness value will be evaluated and displayed separately for each chromosome. An example of chromosome is illustrated in Figure 4.

Notation. The following notations are used in the GA approach.

$P = \{1, 2, \dots, p\}$ denotes the set of p plants, indexed by $i \in P$.

$G = \{1, 2, \dots, g\}$ denotes the set of g units of type boiler-turbine indexed by $t \in G$.

$N = \{1, 2, \dots, g\}$ denotes the set of g units of type distiller indexed by $d \in N$.

(Note that the total number of boiler-turbine units is equal to the total number of distiller units.)

$T = 1, 2, \dots, h$ is length of the maintenance planning horizon (week), indexed by $k \in T$.

NM is number of chromosomes (population).

ND is number of genes (nodes) in chromosome where $ND = g$.

NB is portion of chromosome, called neighborhood.

ItIn is number of inner iterations (iterations within specific population NM).

ItOut is number of outer iterations which is equal to the number of populations that were generated during the GA model.

ItAll is number of iterations = ItIn * ItOut.

E_t is earliest time (week) that maintenance on unit t can start, where $t \in G$.

L_t is latest time (week) that maintenance on unit t can start, where $t \in G$.

E_d is earliest time (week) that maintenance on unit d can start, where $d \in N$.

L_d is latest time (week) that maintenance on unit d can start, where $d \in N$.

K_t is maximum electricity production capacity for unit t , where $t \in G$.

J_t is maximum water production capacity for unit d , where $d \in N$.

OP_t is minimum number of units t in operation, where $t \in G$.

OP_d is minimum number of units d in operation, where $d \in N$.

DE_k is total demand of electricity in period k .

DW_k is total demand of water in period k .

W_{\min} is minimum allowed reservoir level amount of water.

W_{\max} is maximum allowed reservoir level amount of water.

INw is initial reservoir amount of water.

TCE is the total amount of manpower available for electricity equipment.

TCW is the total amount of manpower available for water equipment.

$MANE_t$ is amount of manpower required for unit t , where $t \in G$.

$MANW_d$ is amount of manpower required for unit d , where $d \in N$.

A_t is electricity production rate for unit t , where $t \in G$.

A_d is water production rate for unit d , where $d \in N$.

M_t is maintenance duration of unit t , where $t \in G$.

M_d is maintenance duration of unit d , where $d \in N$.

WE_{ik} is number of electricity units operational in week k , where $i \in P$.

WW_{ik} is number of water units operational in week k , where $i \in P$.

Decision Variables. Consider

$$x_{tk} = \begin{cases} 1 & \text{unit } t \text{ strats maintenance} \\ & \text{in the } k\text{th week, } t \in G, \\ 0 & \text{otherwise,} \end{cases}$$

where $k \in [E_t, L_t]$, $t \in G$,

(1)

$$y_{dk} = \begin{cases} 1 & \text{unit } d \text{ strats maintenance} \\ & \text{in the } k\text{th week, } d \in N, \\ 0 & \text{otherwise,} \end{cases}$$

where $k \in [E_d, L_d]$, $d \in N$.

The available production of electricity (available MW) fl_k and the available production of water (available MGID) fw_k at period k are calculated as follows:

$$fl_k = \sum_{t=1}^g (1 - x_{tk}) A_t, \quad (2)$$

$$fw_k = \sum_{d=1}^g (1 - y_{dk}) A_d.$$

45	9	21	5	8	36	19	46	22	28
----	---	----	---	---	----	----	----	----	----

FIGURE 5: A chromosome that represents one feasible solution.

The reservoir amount of water Res_1 in first week is given by

$$Res_1 = fw_1 + INw - DW_1 \quad (3)$$

and the reservoir amount of water Res_k at period $k \neq 1$ is computed by

$$Res_k = Res_{k-1} + fw_k - DW_k. \quad (4)$$

4. Genetic Algorithm Method

This section explains how the genetic algorithm procedure is implemented.

4.1. Representation. GAs work with a population of chromosomes (strings) where each chromosome represents one potential solution to the optimization problem. Most of the GAs apply binary chromosomes to represent chromosome. Here an integer chromosome of length g as the representation of chromosome is used, where g is the number of units in the problem. There are two types of chromosomes: one represents units of boiler-turbine t , while the other one represents units of distiller, d . An important part of GAs is encoding the value of each gene in the chromosome belonging to the population. This encoding is significant to the GA operations. The value of each gene in the chromosome is the unit maintenance week starting time, and the sequence of the genes is the order of units. For example, a chromosome can be represented as given in Figure 5.

This chromosome represents one feasible solution of 10 genes (node MS). In this representation, the maintenance of unit number 1 starts at week 45, and unit number 2 starts at week 9, and so on for the other units. There are two types of chromosomes as mentioned previously: one holds only units of boiler-turbine t , while the second holds units of distiller d .

The operations accomplished when implementing a GA model are in the following order:

- (1) Initialize the population.
- (2) Calculate the fitness for each chromosome in the population.
- (3) Perform the crossover and mutation operators on the population.
- (4) Go to step (2) until a certain number of iterations is reached or a specific criterion is satisfied.
- (5) Reproduce a new chromosome to form a new population; then go to step (2).

4.2. Initial Population. There are two types of units, t and d ; two GA models for each type of unit were created using the same methodology. On the other hand, each model works separately; this means if boiler-turbine is under

Parent 1	45	9	21	5	8	36	19	46	22	28
Parent 2	14	26	8	44	32	1	27	32	19	50

FIGURE 6: Two parent chromosomes.

maintenance, the distiller will not be influenced. First, each GA model generates an initial population consisting of NM chromosomes of type t and NM chromosomes of type d , respectively; the generation is carried out randomly using a trial process, where all conditions mentioned in Section 4.3 are satisfied. The second step is to calculate the fitness value (objective function) for each chromosome where the fitness value associated with each chromosome is the *standard deviation* of the gaps between the demand and the available production on the whole maintenance planning horizon. The reason behind using the standard deviation is to keep the gaps between the demand and the available production almost equal during the time horizon T . This means that there is a sufficient production in period k in case any breakdown occurred. Chromosomes with good fitness value (lower standard deviation) will have the higher chance to be selected as parents for mating. The standard deviations S_t and S_d for both types, t and d , respectively, are calculated as follows:

$$S_t = \sqrt{\frac{\sum_{k=1}^h (\text{gap}l_k - \text{ave})^2}{h-1}}, \quad (5)$$

$$S_d = \sqrt{\frac{\sum_{k=1}^h (\text{gap}w_k - \text{avd})^2}{h-1}}.$$

Let $\text{gap}l_k$ stand for the electricity gap between the demand of electricity and available MW in period (k), while $\text{gap}w_k$ stand for the water gap between the demand of water and the available MIGD in period (k). These gaps are given as

$$\text{gap}l_k = fl_k - DE_k, \quad \text{gap}w_k = fw_k - DW_k \quad (6)$$

and their respective averages ave and avd are found by

$$\text{ave} = \frac{\sum_{k=1}^h \text{gap}l_k}{h}, \quad \text{avd} = \frac{\sum_{k=1}^h \text{gap}w_k}{h}. \quad (7)$$

Once two parents are selected, offspring are produced using crossover and mutation operators simultaneously.

4.3. Crossover Operation. Crossover is a process of taking two parent solutions to produce two children solutions from them. There are many techniques for crossover operation; see, for example, [39, 40]. The 2-point and 4-point crossover techniques are adapted in this research; see [33, 41]. To explain these two techniques, suppose two chromosomes (parents) are presented, as illustrated in Figure 6.

The 2-point crossover starts by selecting two crossover points on both parent chromosomes randomly. All data beyond the two selected points (called neighborhood (NB))

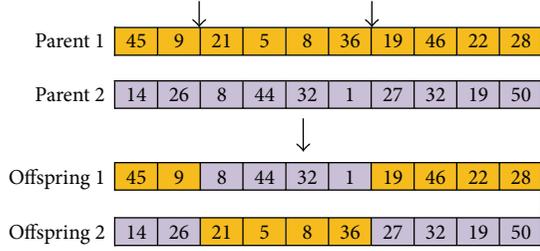


FIGURE 7: An illustration of 2-point crossover.

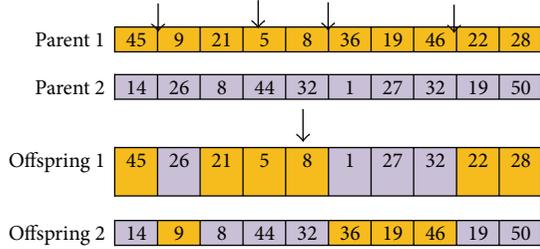


FIGURE 8: An illustration of 4-point crossover.

in each chromosome are swapped. This produces the children (offspring) shown in Figure 7. The size of the selected neighborhood NB varies but does not exceed 50% of the length of the chromosome. As an example of neighborhood size, in Figure 7 the size of neighborhood is 40%, 4 genes out of 10 the length of chromosome.

With the 4-point crossover four crossover positions are selected at random and the value of genes exchanged between the chromosomes along with these points. Then two new offspring are produced. The length between two selected points is not fixed, where it could bind only one gene, or it could be segment as in Figure 8.

After each crossover trial, all the following conditions must be satisfied:

$$WE_{ik} - \sum_{t=1}^g x_{tk} \geq OP_t, \quad (8)$$

$$WW_{ik} - \sum_{t=1}^g y_{dk} \geq OP_d, \quad (9)$$

$$fl_k \geq DE_k, \quad (10)$$

$$fw_k \geq DW_k, \quad (11)$$

$$W_{\min} \leq Res_k \leq W_{\max}. \quad (12)$$

Conditions (8) and (9) ensure that total number of units operating in week k is more than or equal to the minimum total number of units required to be operational in week k . Conditions (10) and (11) indicate that the available production in week k must be more than or equal to the demand. The last condition (12) ensures that the reservoir level amount of water remains within the boundaries.

Furthermore, starting a preventive maintenance task must be more than or equal to the earliest time. Meantime,

when the starting time for a specific unit takes place, the maintenance duration must not exceed the latest time. Therefore, any crossover trial for unit t must be within $[E_t, L_t]$ and must be within $[E_d, L_d]$ in case of unit d . The conditions are as follows.

If $x_{tk} = 1$, then the following conditions must be satisfied:

$$\begin{aligned} k &\geq E_t, \\ M_t + k &\leq L_t. \end{aligned} \quad (13)$$

And if $y_{dk} = 1$, then the following conditions must be satisfied:

$$\begin{aligned} k &\geq E_d, \\ M_d + k &\leq L_d. \end{aligned} \quad (14)$$

Because of the limitation of manpower required to serve units, the following conditions must be satisfied:

$$\sum_{t=1}^g \text{MANE}_t \cdot x_{tk} \leq \text{TCE} \quad \forall k \in T, \quad (15)$$

$$\sum_{d=1}^g \text{MANW}_d \cdot y_{dk} \leq \text{TCW} \quad \forall k \in T.$$

Conditions (15) state that no more than the available number of manpower is dedicated.

If one value of gene k violates one of the previous conditions, then further action to address this situation should be taken. The operation is carried out by exchanging the value of gene k' with the opposite offspring which has the lowest gap ($\text{gap}l_{k'}$). If the previous conditions are satisfied, then the exchange succeeds; otherwise k'' will be selected until satisfying all conditions. This procedure continues until an exchange is implemented. On the other hand, if the trial failed, then the gene value remains unmodified (the same procedure is applied in case of distiller d). In addition, each offspring will be evaluated in terms of fitness value. If the fitness value is the best so far then it will be stored as the best chromosome. Meantime, both offspring will be used in the next step, applying mutation operation to each.

4.4. Mutation Operation. The mutation operator plays the next role in GA. The objective of the mutation is to interrupt the two offspring chromosomes slightly by swapping two gene values. There are many mutation operators for permutation representation; see, for instance, [39, 42].

Two mutation operators were applied, *exchange nodes* between two chromosomes and *swap nodes* for one chromosome, as shown in Figure 9. First, *exchange nodes* are applied. The performance is accomplished by selecting gene in offspring 1 with the largest gap, then selecting the gene in offspring 2 whose gap is the smallest. Finally, the values are exchanged. All conditions mentioned in the crossover operation must be satisfied. If an infeasible solution occurs, then *swap nodes* operation will be applied on the infeasible offspring. Application of this method is accomplished by searching for gene in this offspring with lowest gap, for

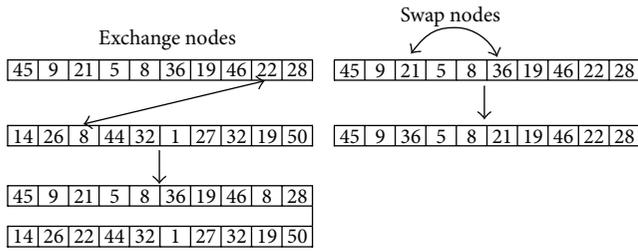


FIGURE 9: Several common mutation operations.

instance, gene k' in distiller; then the gene values will be swapped with gene k , where all conditions are satisfied. If operation failed, the second lowest gap of gene k'' will be selected and swapped with gene k . The process is repeated until reaching a feasible solution; otherwise the gene value of k remains unmodified.

Once the mutation is completed, fitness value will be evaluated. If the produced fitness value is the best then it will be stored as the best chromosome; otherwise it will be ignored.

These two offspring will be the new two parents for the next iteration. The same procedure explained previously will be applied and repeated for a number of ItIn iterations. However, if these two parents generate the best solution, in which the solution is improved, then iteration will start from zero and the procedure will carry on for a number of ItIn iterations. This means that these two chromosomes could lead to optimality or near optimality.

Once iteration has reached ItIn, then second new population will be generated and used in the next iteration of GA. This generational process is repeated for a number of ItOut iterations. Overall number of iterations ItAll = ItIn * ItOut. During each successive generation, best fitness solution will be stored, deleting the previous one. Figure 10 illustrates all GA operation.

5. Computational Study and Numerical Results

5.1. Description of the Test Study. The GA algorithm described above was coded in VB6 and can be run on Intel Core 2 Duo, 2.50 GHz CPU. To test the model described, a representative example is used to compare three Kuwaiti cogeneration plants. Each plant has seven units, where each unit consists of two pieces of equipment, boiler-turbine (capacity 150 MW) and distiller (capacity 10 MGID). The time-horizon is 52 weeks (one year).

The aim of this experiment is to search for the best fitness value, which means maximizing the availability number of operational units in each plant.

The input data for the model is as follows: maintenance window; it is assumed that the earliest starting time for maintenance is in week one ($E_t = 1$), while the latest starting time for maintenance is in week 52 ($L_t = 52$). The maintenance duration is taken to be between 2 and 4 weeks, depending on the unit type. The resource and maintenance crew constraints

are kept constant. The demand is taken from the *Statistical Year Book*, Ministry of Energy. Figure 11 illustrates two input demands; the first one is the electrical consumption in megawatts (MW), while the second is the water consumption in million imperial gallons per day (MIGD). The maximum total generator capacity is 3150 MW, and the maximum total distiller capacity is 210 MIGD.

5.2. Numerical Results. The performance of GA is implemented by exploring the quality of fitness value (lower standard deviation) in terms of the neighborhood size (NB) for each crossover operator. As mentioned previously, two types of crossover operators were implemented: 2-point and 4-point. So the algorithm was run two times. Neighborhood size NB for each was tested as percentage of chromosome size. The percentages are 10%, 20%, 30%, 40%, 50%, and a fixable NB size in the range [10%, 50%]. Therefore, a set of six runs for each NB size was performed. Additionally, seven different iterations ItIn (10, 20, 30, 40, 50, 60, and 100) were considered. The total number of iterations (ItAll) is fixed to 60000 iterations, where ItAll = ItIn * ItOut; this means if ItIn = 20 then ItOut = 3000. Moreover, to see the effect of the search space, the algorithm was run three times for each population size NM (100, 250, and 500). No further improvement was observed with NM > 500. Thus the total number of problems tested is equal to 126 (six runs for each NB size, seven different iterations ItIn, and three times for each population size NM).

A clear picture of the interactive effects of; two types crossover operator, inner iteration (ItIn), size of Neighborhood (NB) Portion of 16 chromosome, and population size (NM) is given in Table 1.

There are two types of experiments: 2-point and 4-point crossover operators. In each experiment, investigation to test the relationship between the number of iterations (ItIn), size of population (NM), and size of NB was implemented. Table 1 shows the results obtained using the two methods of crossover operators for each *Electricity* plants and *Distiller* plants. First column represents the three different population sizes NM neighborhood, while the second column shows the six neighborhood size NB. Due to limitation of this table, only best solution obtained using one of the seven ItIn (10, 20, 30, 40, 50, 60, and 100) was illustrated in column 3, while column 4 represents the best fitness. Computation time for each experiment is illustrated in column 5. Next columns are the same of columns 3, 4, and 5.

Since the standard deviation (fitness value) measures how far the numbers are from their average, it is obvious that most of the fitness values (best solution) are small: in electricity the fitness value lies between 97.639 and 105.849, while in water the fitness value is between 3.978 and 4.441. The best solution for electricity is equal to 97.639, which can be obtained using 4-point crossover with NM = 250 (size of population), NB = 10% (neighborhood size), and ItIn = 60 (number of inner iterations), where the solution time is 107 sec. Meantime, the best solution for water is equal to 3.978, which can be obtained using either NM = 250, NB size = 10%, and ItIn = 60, with solution time 90 min, or NM = 500, NB size = 10%,

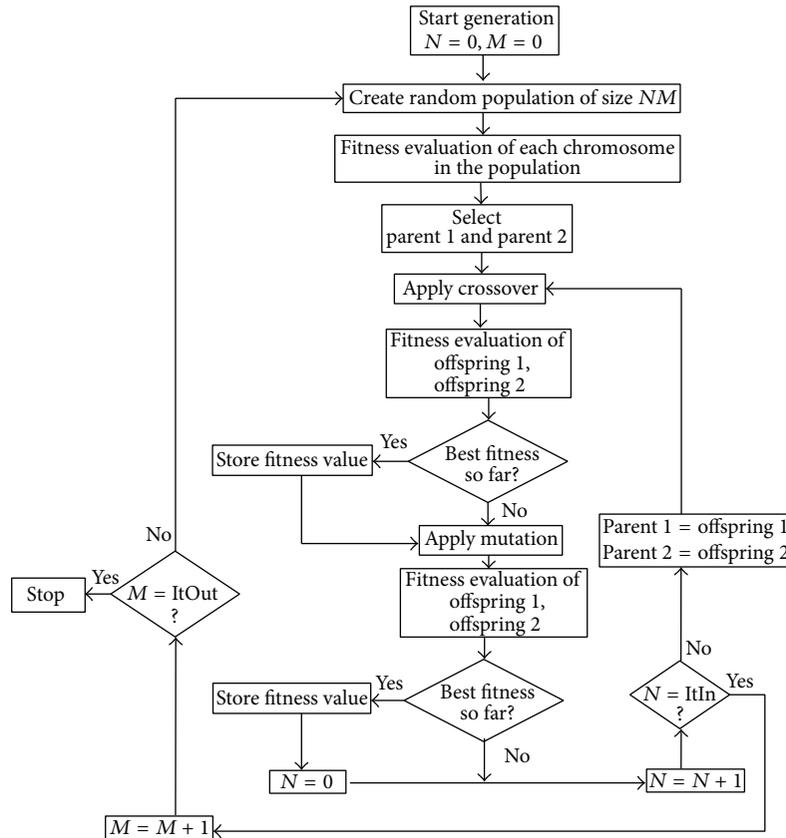


FIGURE 10: The flow chart of genetic algorithm operation.

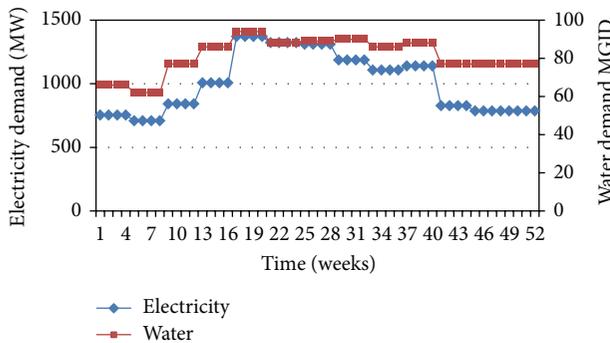


FIGURE 11: Input demand for the model.

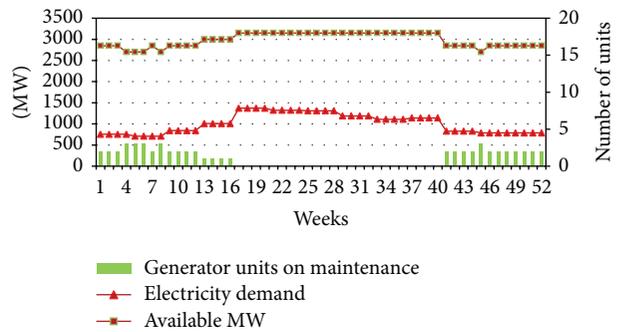


FIGURE 12: Unit system output loads and gross reserves for 21 units.

and $ItIn = 50$, with solution time 78 sec. Both are obtained using 4-point crossover.

It is obvious from this result that using 4-point crossover with $NM = 250$, NB size = 10%, and $ItIn = 60$ generates slightly best fitness value for both electricity and water units. The small value of standard deviation indicates the gaps between the demand and the available production are close to the average, which means that there is a sufficient production in any period in case of any breakdown occurred.

Figure 12 illustrates the available MW and the demand MW in each period of planning horizon (in week). Also, Figure 12 illustrates the number of units on maintenance

for each week during planning horizon. Figure 13 illustrates the same as Figure 12 but for water (MGID). It is clear—for Figures 12 and 13—that when there is increasing demand, there are fewer units taken for maintenance provided that the demand constrains are satisfied. On the other hand, when a generator is under maintenance, there is a decline in the total generator. The same thing is for water: when distiller is under maintenance, there is a decline in the total distiller capacity. Table 2 illustrates the generator and distiller capacity being maintained for each week. All constraints are satisfied; the table shows that when any preventive maintenance task starts it continues until it completes its duration, hence

TABLE I: Summary results of the computational experiments.

Plant type		Electricity* (ave = 1974.04)						Water** (avw = 118.31)					
Crossover operator		2-point			4-point			2-point			4-point		
NM	NB size	ItIn	Best solution***	CPU	ItIn	Best solution	CPU	ItIn	Best solution	CPU	ItIn	Best solution	CPU
100	10%	60	101.334	118	10	100.401	93	20	4.07	108	50	3.98	87
	20%	100	101.305	71	10	102.518	119	10	4.07	83	40	4.07	75
	30%	40	105.403	153	10	101.624	79	20	4.348	111	40	4.348	79
	40%	50	104.281	75	100	102.66	67	100	4.206	80	60	4.25	57
	50%	10	104.422	112	30	104.591	145	40	4.116	74	100	4.348	106
	Rnd	20	104.422	83	30	103.63	83	40	4.206	73	30	4.116	74
250	10%	10	99.489	123	60	97.639	107	20	4.024	72	60	3.978	90
	20%	40	101.131	124	20	103.403	83	10	4.217	102	50	4.348	76
	30%	60	105.793	85	50	102.804	133	10	4.161	147	30	4.206	75
	40%	20	101.16	145	100	100.84	147	50	4.348	79	20	4.261	82
	50%	30	104.309	112	40	101.769	115	60	4.307	180	40	4.024	123
	Rnd	60	105.849	87	20	102.518	150	60	4.217	120	60	4.441	91
500	10%	10	102.288	119	100	101.827	114	40	4.024	139	50	3.978	78
	20%	40	103.8	98	100	101.595	82	10	4.345	164	50	4.07	80
	30%	20	102.23	147	100	101.335	81	40	4.024	131	60	4.206	84
	40%	30	102.66	107	20	104.984	88	60	4.348	87	100	4.39	75
	50%	50	103.772	82	10	102.058	129	30	4.426	201	10	4.305	122
	Rnd	40	105.292	98	100	103.061	71	30	4.116	105	40	4.116	94

*The gap average of electricity = 1974.04.

**The gap average of water = 118.31.

*** Best solution is related to fitness value.

CPU in seconds.

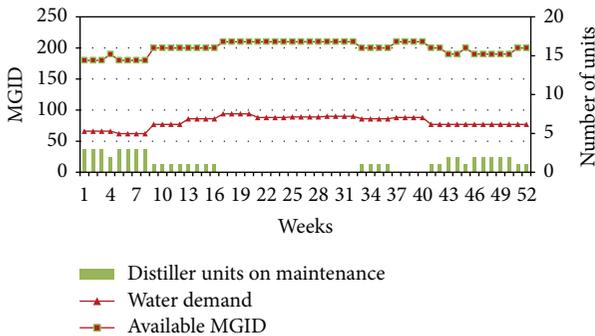


FIGURE 13: Distiller unit system output loads and gross reserves for 21 units.

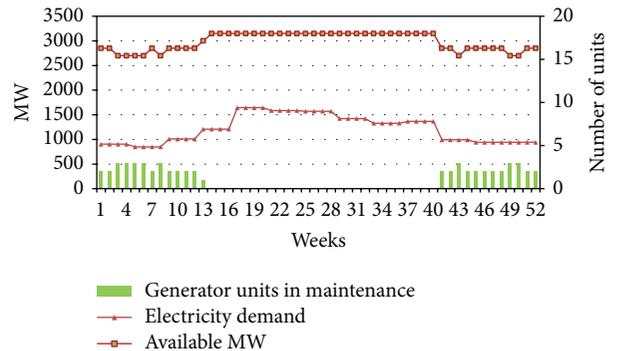


FIGURE 14: Generator unit system output loads and gross reserves for 21 units with 20 percent increase in demand.

satisfying the maintenance completion duration constraints. Crew constraints are kept constant; when these constraints are tightened, then the final schedule for the preventive maintenance tasks will change.

Since the temperature in Kuwait is very high in summer, this requires enormous consumption of electricity due to the need for air conditioners, while the opposite is in winter, and this explains why most of the scheduling of preventive

maintenance is in the winter as in Table 2. Meanwhile, the same reason applies on water preventive maintenance.

Sensitivity analysis was implemented on the model proposed to evaluate its robustness and to observe the impact when the demand is increased. The model proposed was run again with increasing by 20 per cent of the electricity and water demand. Figure 14 shows that when the demand

TABLE 2: Maintenance schedules for 21 units of electricity and 21 units of water.

Week number	Generating units scheduled for maintenance	
	Electricity	Water
1	14, 16	2, 13, 15
2	14, 16	2, 13, 15
3	4, 7	12, 13, 15
4	4, 7, 10	6, 12
5	7, 10, 12	6, 10, 21
6	7, 10, 12	8, 10, 21
7	12, 19	3, 8, 17
8	12, 17, 19	3, 8, 17
9	9, 17	7
10	9, 17	7
11	2, 3	14
12	2, 3	14
13	2	14
14	2	5
15	15	5
16	15	5
17	—	—
18	—	—
19	—	—
20	—	—
21	—	—
22	—	—
23	—	—
24	—	—
25	—	—
26	—	—
27	—	—
28	—	—
29	—	—
30	—	—
31	—	—
32	—	—
33	—	20
34	—	20
35	—	20
36	—	20
37	—	—
38	—	—
39	—	—
40	—	—
41	1, 18	18
42	1, 18	18
43	8, 11	1, 9
44	8, 11	1, 9

TABLE 2: Continued.

Week number	Generating units scheduled for maintenance	
	Electricity	Water
45	5, 11, 20	1
46	5, 20	1, 11
47	5, 13	4, 11
48	5, 13	4, 11
49	6, 21	16, 19
50	6, 21	16, 19
51	6, 21	19
52	6, 21	19

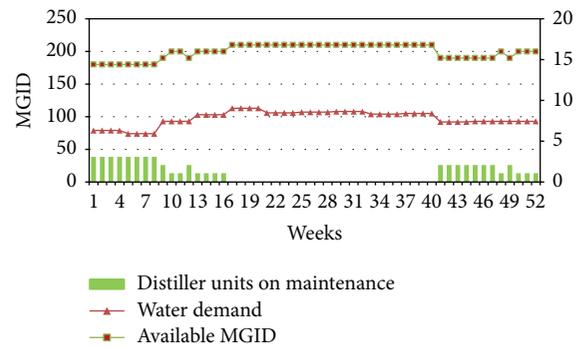


FIGURE 15: Distiller unit system output loads and gross reserves for 21 units with 20 percent increase in demand.

for electricity increases, the gap between the total available capacity and the demand curve reduces. The same thing is for water illustrated in Figure 15: it shows that when the demand for water increases, the gap between the total available capacity and the demand curve reduces. Thus the genetic algorithm proposed offers a promising viable approach for the MS of generators and distillers.

6. Conclusion

This paper proposed a GA method for solving the maintenance scheduling (MS) problem for cogeneration plants. The proposed approach was modeled to generate two preventing maintenance schedules, one for electricity and the other for water distiller. Two types of crossover operators were adopted, 2-point and 4-point. The objective function of the model is to maximize the available number of operational units in each plant. The results obtained were very close to the solution quality, where 4-point slightly produces better solution than 2-point ones for both electricity and water distiller. Genetic operators have been designed to incorporate further investigation in specific space before diversify to search in another solution space. The model has demonstrated that GA is a powerful technique for modeling the cogeneration plants preventing maintenance schedules. Moreover, this approach can be applied to any problem with any number of plants with any number of units. Meantime, user can reach better solution if he/she increases number of iterations.

The numerical analysis shows that genetic algorithm may hopefully be a new approach for kinds of difficult-to-solve problems.

For a future work, it is possible to extend the objective function to include the cost for the maintenance and the production of the units using an optimal approach (exact solution). The methodology implemented will first generate all feasible schedules then select the optimal solution using an optimal approach, such as integer programming (IP), where all constraints are satisfied.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

- [1] M. Alardhi, R. G. Hannam, and A. W. Labib, "Preventive maintenance scheduling for multi-cogeneration plants with production constraints," *Journal of Quality in Maintenance Engineering*, vol. 13, no. 3, pp. 276–292, 2007.
- [2] A. M. El-nashar and M. S. Khan, "Economic scheduling of the UAN cogeneration plant: a preliminary optimization study," *Desalination*, vol. 85, no. 1, pp. 93–127, 1991.
- [3] M. Y. El-Sharkh, R. Yasser, and A. A. El-Keib, "Optimal maintenance scheduling for power generation systems—a literature review," in *Proceeding of the Maintenance and Reliability Conference*, Knoxville, Tenn, USA, 1998.
- [4] K. Wangnick, *IDA Worldwide Desalting Plants Inventory*, Produced by Wangnick Consulting for the International Desalination Association, Gnarrenburg, Germany, 2002.
- [5] C. Zhao and H. Tang, "A note to due-window assignment and single machine scheduling with deteriorating jobs and a rate-modifying activity," *Computers and Operations Research*, vol. 39, no. 6, pp. 1300–1303, 2012.
- [6] C. M. Joo and B. S. Kim, "Genetic algorithms for single machine scheduling with time-dependent deterioration and rate-modifying activities," *Expert Systems with Applications*, vol. 40, no. 8, pp. 3036–3043, 2013.
- [7] V. S. Gordon and A. A. Tarasevich, "A note: common due date assignment for a single machine scheduling with the rate-modifying activity," *Computers and Operations Research*, vol. 36, no. 2, pp. 325–328, 2009.
- [8] C.-Y. Lee and V. J. Leon, "Machine scheduling with a rate-modifying activity," *European Journal of Operational Research*, vol. 128, no. 1, pp. 119–128, 2001.
- [9] M. Ben-Daya, S. O. Duffuaa, and Abdul Raouf, *Maintenance, Modeling, and Optimization*, Kluwer Academic Publishers, 2000.
- [10] Z. A. Yamayee, "Maintenance scheduling: description, literature survey and interface with overall operations scheduling," *IEEE Transactions on Power Apparatus and Systems*, vol. 101, no. 8, pp. 2770–2779, 1982.
- [11] H. H. Zuern and V. H. Quintana, "Generator maintenance scheduling via successive approximations dynamic programming," *IEEE Transactions on Power Apparatus and Systems*, vol. 94, no. 2, pp. 665–671, 1975.
- [12] R. Dekker and P. A. Scarf, "On the impact of optimisation models in maintenance decision making: the state of the art," *Reliability Engineering and System Safety*, vol. 60, no. 2, pp. 111–119, 1998.
- [13] Y. S. Sherif, "Reliability analysis: optimal inspection and maintenance schedules of failing systems," *Microelectronics Reliability*, vol. 22, no. 1, pp. 59–115, 1982.
- [14] C. Valdez-Flores, "Survey of preventive maintenance models for stochastically deteriorating single-unit systems," *Naval Research Logistics*, vol. 36, no. 4, pp. 419–446, 1989.
- [15] T. Aven and R. Dekker, "A useful framework for optimal replacement models," *Reliability Engineering and System Safety*, vol. 58, no. 1, pp. 61–67, 1997.
- [16] R. Dekker, "Integrating optimisation, priority setting, planning and combining of maintenance activities," *European Journal of Operational Research*, vol. 82, no. 2, pp. 225–240, 1995.
- [17] L. M. Pintelon and L. F. Gelders, "Maintenance management decision making," *European Journal of Operational Research*, vol. 58, no. 3, pp. 301–317, 1992.
- [18] R. Dekker, "Applications of maintenance optimization models: a review and analysis," *Reliability Engineering and System Safety*, vol. 51, no. 3, pp. 229–240, 1996.
- [19] P. A. Scarf, "On the application of mathematical models in maintenance," *European Journal of Operational Research*, vol. 99, no. 3, pp. 493–506, 1997.
- [20] British Standards Institution, *Glossary of General Terms Used in Maintenance Organization*, 1984.
- [21] M. L. Baughman, N. A. Eisner, and P. S. Merrill, "Optimizing combined cogeneration and thermal storage systems: an engineering economics approach," *IEEE Transactions on Power Systems*, vol. 4, no. 3, pp. 974–980, 1989.
- [22] M. F. J. Bos, R. J. L. Beune, and R. A. M. van Amerongen, "On the incorporation of a heat storage device in Lagrangian Relaxation based algorithms for unit commitment," *International Journal of Electrical Power and Energy Systems*, vol. 18, no. 4, pp. 207–214, 1996.
- [23] H. B. Puttgen and P. R. MacGregor, "Optimum scheduling procedure for cogenerating small power producing facilities," *IEEE Transactions on Power Systems*, vol. 4, no. 3, pp. 957–964, 1989.
- [24] A. K. Coleman, "Optimization of a single effect, multi-stage flash distillation desalination system," *Desalination*, vol. 9, no. 4, pp. 315–331, 1971.
- [25] K. Moslehi, M. Khadem, R. Bernal, and G. Hernandez, "Optimization of multiplant cogeneration system operation including electric and steam networks," *IEEE Transactions on Power Systems*, vol. 6, no. 2, pp. 484–490, 1991.
- [26] J. F. Dopazo and H. M. Merrill, "Optimal generator maintenance scheduling using integer programming," *IEEE Transactions on Power Apparatus and Systems*, vol. 94, no. 5, pp. 1537–1545, 1975.
- [27] K. W. Edwin and F. Curtius, "New maintenance-scheduling method with production cost minimization via integer linear programming," *International Journal of Electrical Power and Energy Systems*, vol. 12, no. 3, pp. 165–170, 1990.
- [28] S. P. Canto, "Application of Benders' decomposition to power plant preventive maintenance scheduling," *European Journal of Operational Research*, vol. 184, no. 2, pp. 759–777, 2008.
- [29] K. S. Moghaddam and J. S. Usher, "Preventive maintenance and replacement scheduling for repairable and maintainable systems using dynamic programming," *Computers and Industrial Engineering*, vol. 60, no. 4, pp. 654–665, 2011.
- [30] K. S. Moghaddam and J. S. Usher, "Sensitivity analysis and comparison of algorithms in preventive maintenance and replacement scheduling optimization models," *Computers & Industrial Engineering*, vol. 61, no. 1, pp. 64–75, 2011.

- [31] I. El-Amin, S. Duffuaa, and M. Abbas, "Tabu search algorithm for maintenance scheduling of generating units," *Electric Power Systems Research*, vol. 54, no. 2, pp. 91–99, 2000.
- [32] J. T. Saraiva, M. L. Pereira, V. T. Mendes, and J. C. Sousa, "A Simulated Annealing based approach to solve the generator maintenance scheduling problem," *Electric Power Systems Research*, vol. 81, no. 7, pp. 1283–1291, 2011.
- [33] K. Al-Hamad, M. Al-Ibrahim, and E. Al-Enezy, "A genetic algorithm for ship routing and scheduling problem with time window," *American Journal of Operations Research*, vol. 2, no. 3, pp. 417–429, 2012.
- [34] T.-K. Liu, J.-T. Tsai, and J.-H. Chou, "Improved genetic algorithm for the job-shop scheduling problem," *The International Journal of Advanced Manufacturing Technology*, vol. 27, no. 9-10, pp. 1021–1029, 2006.
- [35] <http://waterwebster.org/Desalination.htm>.
- [36] K. Wangnick, *IDA Worldwide Desalting Plants Inventory*, Wangnick Consulting for the International Desalination Association, Gnarrenburg, Germany, 1990.
- [37] MEW, *Statistical Year Book Kuwait*, Ministry of Electricity, State of Kuwait, 2010.
- [38] M. Gen and R. Cheng, *Genetic Algorithms and Engineering Design*, John Wiley & Sons, New York, NY, USA, 1997.
- [39] J. Berger and M. Barkaoui, "A parallel hybrid genetic algorithm for the vehicle routing problem with time windows," *Computers & Operations Research*, vol. 31, no. 12, pp. 2037–2053, 2004.
- [40] C. Prins, "A simple and effective evolutionary algorithm for the vehicle routing problem," *Computers & Operations Research*, vol. 31, no. 12, pp. 1985–2002, 2004.
- [41] B. M. Baker and M. A. Ayechev, "A genetic algorithm for the vehicle routing problem," *Computers and Operations Research*, vol. 30, no. 5, pp. 787–800, 2003.
- [42] H.-S. Hwang, "An improved model for vehicle routing problem with time constraint based on genetic algorithm," *Computers & Industrial Engineering*, vol. 42, no. 2–4, pp. 361–369, 2002.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

