Multiobjective Genetic Algorithm for Demand Side Management of Smart Grid

Xiao-Min Hu, Zhi-Hui Zhan, Ying Lin, Yue-Jiao Gong, Wei-Jie Yu, Yao-Xiu Hu, and Jun Zhang
Department of Computer Science, Sun Yat-sen University
Key Laboratory of Intelligent Sensor Networks (Sun Yat-sen University), Ministry of Education
Key Laboratory of Software Technology, Education Department of Guangdong Province
P.R. China
xmhu@ieee.org; junzhang@ieee.org

***Abstract*—Demand side management is one of the most effective methods to control the usage of energy so as to achieve reliability and sustainability in the smart grid. Conventional methods for generating the management scheme are generally based on one objective, which represents only the requirements of energy suppliers or only the energy consumers. In this paper, a multiobjective genetic algorithm (GA) is proposed for extending the optimization problems by considering the objectives from the two conflicting groups and some compromise solutions are provided. The performance of the multiobjective GA is compared with its single objective version by three different cases. The results show that the solutions obtained by the multiobjective GA are more reasonable and better solutions for a single objective can even be found.**

Keywords- Genetic algorithms, multiobjective, demand side management, smart grid

I. Introduction

Smart grid [1][2] is a trend for replacing the existing electricity grid by using facilities such as intelligent meters, sensors, and other control techniques to manage the usage of energy for better efficiency, reliability, and sustainability. One of the distinguished features of smart grid is the interactions between the utility companies (suppliers) and the end users (consumers). The utility companies set the price of electricity based on the demand load. Higher demand load will induce a more expansive price for attracting the load demand of end users to shift to nonpeak hours [3].

The energy demand varies from seasons or even hours but most of the demand load cannot be scheduled to another time, especially for commercial areas and industrial areas, for example the economic markets open at the fixed time and some noisy or critical production processes can only work at daytime. Therefore, the energy consumed by controllable devices is generally the focus of smart grid management [4]. Demand side management is a way for controlling the usage of energy by the demand side, i.e. the utility companies and end users.

Demand side management in smart grid has the methods such as direct load control, smart pricing, and planned load shifting, etc [5]. Direct load control requires remote control to the operations of devices and cannot always be realized. Smart pricing dynamically adjusts the price of electricity based on the usage of energy but it may cause confusion to end users. Since part of the demand load is flexible to be scheduled to another time, a day-ahead planned load shifting technique is more reliable to reduce peak demand for utility companies or the electricity bill for end users.

Previous work of demand side management generally focuses on only a single objective such as minimizing the energy cost, minimizing the difference to the objective curve, or minimizing the peak-to-average ratio, etc [4]-[6]. However, in view of the electricity supplier, the demand load is better to be balanced to avoid a demand peak. In view of the consumers, a cheaper electricity bill is more preferred. These objectives may be conflicting with each other and the optimal solution to one of the objectives will lead to the unsatisfactory result to the other objectives. In this paper, a multiobjective genetic algorithm (GA) is proposed to address the above problem and provides a set of optimal solutions to both the suppliers and consumers.

Genetic algorithms have been successfully applied in solving various optimization problems [7]-[14]. Since a large number of optimization problems involve more than one objectives, which may be conflicting with each other, the research on solving the multiple objective optimization problems has became a hot spot. The motivation of this paper is to extend the traditional single objective optimization problems in the smart grid to a multiple objective optimization problem and provides more reasonable compromise solutions.

The remainder of this paper is organized as follows. Section II introduces the background and the definition of the

This work was supported in part by the National High-Technology Research and Development Program ("863" Program) of China under Grand No. 2013AA01A212, by the National Science Fund for Distinguished Young Scholars under Grant 61125205, by the National Natural Science Foundation of China under Grant 61070004, 61202130, by the NSFC Joint Fund with Guangdong under Key Project U1201258 and U1135005, Guangdong Natural Science Foundation S2012040007948, Fundamental Research Funds for the Central Universities No. 12lgpy47, Specialized Research Fund for the Doctoral Program of Higher Education 20120171120027.

978-1-4673-5909-2/13/$31.00 ©2013 IEEE

problem. The realization of the proposed algorithm is described in detail in Section III. A range of experiments are tested and the performance of the multiobjective GA is analyzed in Section IV. Section V concludes the paper.

II. Background

In order to provide optimal schemes to regulate controllable devices to meet some desired properties, the demand load is forecasted based on the previous usage. The period of estimation is a forecasted window. It can be one day, a week, or even longer. In that case, the algorithm can be run beforehand so as to generate a scheme for organizing the energy consumption of controllable devices in the window. In this paper, the proposed algorithm is designed for one day ahead planned load shifting, and it can be easily extended to the other sizes of forecasted windows.

A. Definition of the Optimization Problem

By collecting energy load demand and referring to the working pattern in the past, the usage of electricity in the forecasted window with m hours can be predicted. The forecasted load demand is denoted as Y_i, $i = 1,2,\ldots,m$. The devices which consume energy are divided into two groups, i.e. uncontrollable devices and controllable devices. The uncontrollable devices cannot be scheduled to start working earlier or later, so their energy consumption in each hour is fixed and is denoted as U_i, $i = 1,2,\ldots,m$. On the other hand, the controllable devices are more flexible to work at another time mainly due to their high tolerance to the operation time. For example, a washing machine in a residential house can be scheduled to work for convenience. Therefore, the scheduling of the controllable devices is the main focus in this paper.

The objective for shifting the load demand can be various. In view of the electricity suppliers, the objective is to avoid the appearance of a high peak load demand and a long demand valley. Relatively steady energy requirements can increase the sustainability of the facility and reduce the cost for building new infrastructure. In view of the consumers, they want the electricity bill to be minimized. The objectives of different groups may be conflicting and the best way is to provide them a range of optimal solutions to these objectives and let the decision maker to decide.

1) Objective I – Fitting to the Objective Load Curve

For the utility companies, one of the objective is to bring the energy consumption load curve as close to the objective load curve as possible. In [4], the objective is to minimize the utility bill, so the objective load curve is chosen to be inversely propositional to the electricity market prices. Therefore, the objective curve for each time step t is set as

$$Objective(t) = \frac{1/price(t)}{\sum_{j=1}^{m} price(j)} \sum_{i=1}^{m} Y_i \tag{1}$$

where $price(t)$ is the price per Wh of electricity at time step t. The objective curve reflects the way that the electricity bill is balanced in every hour. If the price is high, the energy consumption will be small, and vice versa. The objective is to minimize the difference between the actual energy consumption curve and the objective curve and it is defined as

$$\text{Minimize } f = \sum_{t=1}^{m} (Pload(t) - Objective(t))^2 \tag{2}$$

where $Pload(t)$ is the load after scheduling at time step t, $Objective(t)$ is the load in the objective curve at time step t.

Before scheduling, the load is the forecasted load demand. Since the controllable devices can be arranged to operate at another time, some devices will be scheduled earlier or delayed. Therefore, the load after scheduling is calculated as

$$Pload(t) = Forcast(t) + Connect(t) - Disconnect(t) \tag{3}$$

The $Connect(t)$ is the energy which is shifted from the other time to the time step t, whereas the $Disconnect(t)$ is the energy shifted to the other time step from the time step t.

2) Objective II – Minimizing the Electricity Bill

Although Objective I has the intent to reduce the electricity bill, it is made based on the benefit of the energy suppliers. The optimal solution in view of the energy consumer is to minimize the total electricity bill, as defined as

$$\text{Minimize } f = \sum_{t=1}^{m} \cos t(t) \tag{4}$$

It can be observed that the optimal solution for this objective is that all demand load gathers at the time step with the cheapest price while no energy load at the other step time. Such an ideal solution is unrealistic because the demand load peak is maximized and it may exceed the supply capacity of the electricity grid.

This objective is considered because the optimal solution to the Objective I may not be satisfactory to the energy consumers. Compromise solutions are needed to be found.

III. Implementation of the Proposed Algorithm

In this paper, a multiobjective GA is used to address the optimization problem in smart grid by shifting the controllable devices to start working at appropriate time. In this section, the framework of the algorithm, the encoding of the chromosomes, and the modifications of genetic operations for the demand side management optimization problem will be described.

A. Framework of the Multiobjective Genetic Algorithm

In a traditional single objective GA, the operations of selection, crossover, mutation, and elitist are performed iteratively on a group of chromosomes to approach the optimal solution [7]. When solving a multiobjective optimization problem, more than one objective is considered simultaneously [15]. The comparison of the chromosomes is based on Pareto dominance.

Definition 1 (Pareto Dominance): Suppose there are two chromosomes a_1 and a_2, and n objectives $f_1, f_2, \ldots, f_n$, and smaller values are preferred to larger ones. We term chromosome a_1 dominates chromosome a_2 if $f_i(a_1) \le f_i(a_2)$ for all $i = 1,2,\ldots,n$ and for at least one $i \in \{1,2,...,n\}$ has $f_i(a_1) < f_i(a_2)$. The dominance is denoted as $a_1 \prec a_2$.

Definition 2 (Pareto Frontier): A Pareto frontier is formed by a set of chromosomes that are not dominated by any other chromosomes in the set. If there are no other solutions that can dominate solutions in the Pareto frontier, the frontier is termed the Pareto Optimal.

The target of a muliobjective GA is to find the Pareto Optimal, which is a set of solutions which cannot be dominated by any other solutions. During the iterations of GA, chromosomes are sorted and may form some Pareto frontiers. The chromosomes in different frontiers are comparable, but the ones in the same frontier cannot be compared directly. In order to maintain diversity in the population of chromosomes in each iteration, the crowding distance of a chromosome to all the other chromosomes in the same frontier is computed as an attribute. Besides the boundary chromosomes of the frontier, the chromosome with a larger distance is preferred to be preserved in the next generation than the smaller one. The computation of the crowding distance can be referred to [15].

The flowchart of the multiobjective GA is presented in Fig. 1. The initialization step initializes the parameters such as the number of chromosomes (population size), crossover rate, and mutation rate, etc, and the population of chromosomes. In each iteration, the population in the previous generation is reserved in a set Q_t, where t denotes the index of generation. Then the genetic operations are performed. After evaluating the new population P_t, the new population and the old population are combined and sorted by the Pareto dominance and the crowding distance. The first $|P_t|$ chromosomes are chosen as the solutions for the next generation.

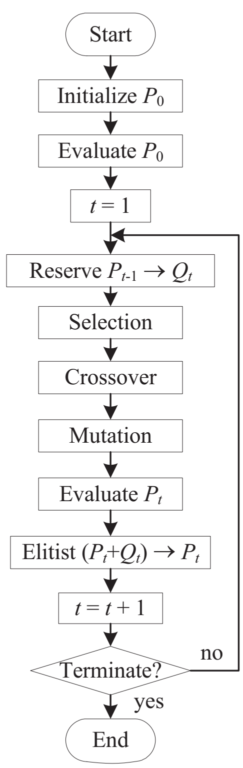


Figure 1. Frowchart of the multiobjective GA.

B. *Encoding of Chromosomes for the Problem*

Similar to the encoding methods used in [4], each gene represents the number of controllable devices which will be delayed to the corresponding time. The difference is that the genes are encoded in decimal values rather than binary bits. The controllable devices are divided into K types, and different types of devices have different working durations and energy load demands. Suppose the maximum delay of each type i of devices is d_i, the forecasted window is M hours ($M \geq d_i$), then the number of genes in each chromosome is

$$N = \sum_{i=1}^{K} [(M - d_i)d_i + \sum_{j=1}^{d_i - 1} j] \tag{5}$$

For example, suppose there are two types of devices, the forecasted window is 5 hours, and each device can be delayed to at most 3 hours. Fig. 2 illustrates an example of a chromosome (the genes are in the box).

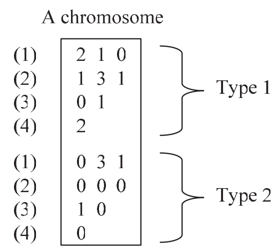


Figure 2. An example of a chromosome.

On the first line in Fig. 2, 2 devices with type 1 are shifted to start from time 1 to time 2, 1 device with type 1 is shifted to start from time 1 to time 3. No device is delayed to start at time 4. For the second line, 1 device is shifted to start from time 2 to time 3, 3 devices are shifted to start from time 2 to time 4, and 1 device is shifted to start from time 2 to time 5. For the third line, only 1 device is delayed from time 3 to time 5. Since the forecasted window is set as 5, no devices can be delayed to start later than time 5. Note that the total number of devices to be delayed to the other time cannot be larger than the number of controllable devices at the considered time step.

C. *Modifications to Genetic Operations for the Problem*

Based on the encoding of the chromosomes, traditional crossover and mutation operations such as the one-point crossover and random mutation may produce infeasible solutions that violate the constraints of a solution. In order to maintain the feasibility of the offspring, modified crossover and mutation operations are proposed for the algorithm.

1) *Modified Crossover*

After performing crossover to two chromosomes, the offspring will be adjusted if they are infeasible solutions. The pseudocode of the process of adjustment is shown in Fig. 3, where V_{ti} is the number of controllable devices with type i at time step t. The value of V_{ti} is determined by the forecasted load demand.

After the adjustment, the total number of delayed devices from a time step will not be larger than the original number of devices at the time step for a specific type of devices in a chromosome.

```
For i = 1 to K
    For t = 1 to M-1
        Calculate the total number of delayed devices with type i as S_ti
        If S_ti > V_ti      //violates the constraint
            randomly select a time step to reduce the number of delayed
            devices until the violation has been eliminated
        End If
    End For
End For
```

Figure 3. Pseudocode of adjustment for each offspring after crossover.

2) Modified Mutation

In the mutation operation, each gene has a probability of a mutation rate to be mutated. If a gene is to be mutated, the new value of the gene will be a random value within the range of the sum of the gene value and the rest number of controllable devices at the time for the type of devices.

IV. EXPERIMENTS

In this section, the algorithm is tested by three smart grid cases, i.e., in the residential area, the commercial area, and the industrial area. The forecasted load demands and wholesale energy prices can be referred to [4], in which the hourly consumption of devices in the three areas is also provided.

- In the residential area, 14 types of controllable devices with the total number 2604 are considered, in which only two types of devices need the second and the third hour respectively to complete their work and the others can finish in one hour.
- In the commercial area, the types of devices reduce to 6 with totally 808 devices but most of them work in two or three hours and their hourly energy consumption is larger than that in the residential area.
- In the industrial area, heavy energy consumed devices are often used and their working durations last the longest, with the maximum 6 hours. In the experiment, 6 types of devices with a total of 109 devices are tested in the industrial area.

For better illustrating the features of the three cases, Table I listed the attributes of the three cases. The first row is the average hourly consumption for all types, i.e. each type has only one device. The second row is the average hourly consumption by considering the number of devices with different types. If the value in the second row is smaller than the one in the first row, it means that the heavier energy consumed devices are fewer so that the average hourly energy consumption is below the average value. It can be observed that the energy consumption of the devices in the industrial area is the largest and lasts the longest.

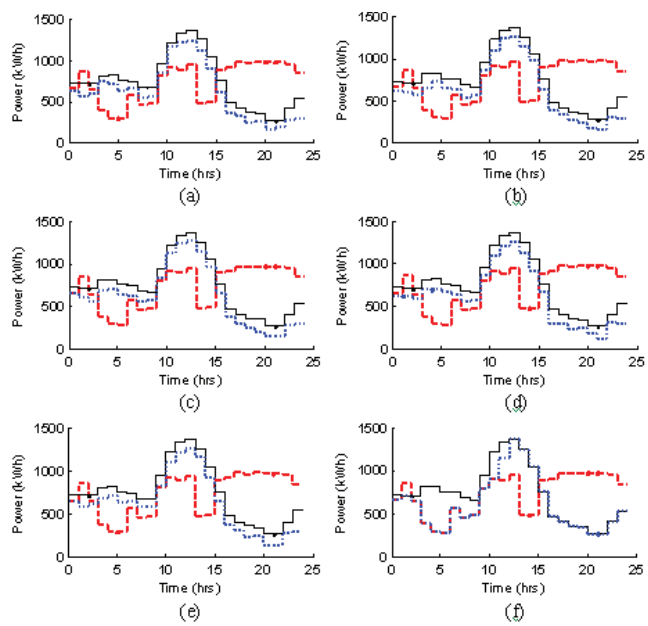
In the experiment, 10 randomly generated instances are tested for each case. The 10 instances have the same amount of total load demand in the same time step. The difference is their distributions of the load demand of uncontrollable devices and controllable devices. Moreover, the generation of the 10 instances for each case can be divided as two groups.

TABLE I. ATTRIBUTES OF THE THREE CASES

Attributes	**Cases**		
	1	**2**	**3**
Avg. Hourly Consump. (kW) for All Types	0.80	2.85	57.76
Avg. Hourly Consump. (kW) for All Devices	0.81	2.77	33.19
Workin Duration (Hour)	1~3	1~3	3~6

- Group 1: The controllable devices are randomly distributed in the time steps without exceeding the total load demand in the time step. The rest load demand in each time step is the consumption by the uncontrollable devices. Instances: Case 1 instances (a) to (e), Case 2 instances (a) to (e).
- Group 2: Randomly select the time step and then randomly select controllable devices to fill in the gap between the forecasted load demand and the objective curve so that the load demand by the uncontrollable devices will not exceed the load demand in the objective curve for the time step. Since the total amount of controllable devices is limited, some time steps cannot achieve the above situation. The instance generation method is for reducing the gap between the optimal curve for Objective I and the theoretical objective curve, which may not be applicable for the cases. Instances: Case 1 instances (f) to (j), Case 2 instances (f) to (j), Case 3 instances (a) to (j).

Fig. 4(a)-(j) illustrates the forecasted load and the objective curves for the ten instances in case 1 the residential area. It can be observed that for the first five instances the load demand by uncontrollable devices is randomly distributed, whereas for the last five instances the load demand by uncontrollable devices is forced to be smaller than the objective curve as possible. Among them, instance (f) is a quite special one for most the load demands by uncontrollable devices are no larger than the objective curve for the time steps.



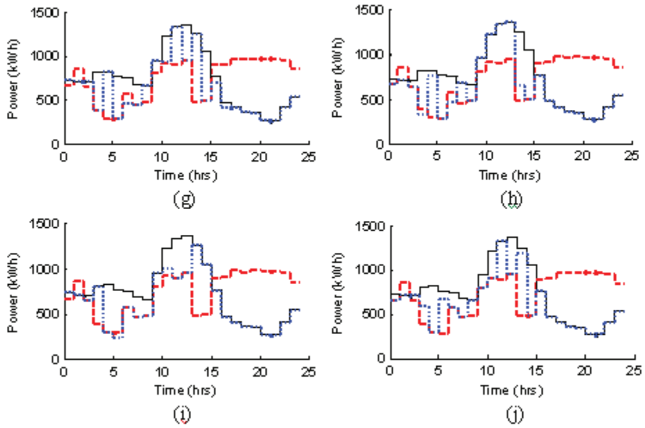


Figure 4. Illustration of the load curves for the ten instances in Case 1. The solid black line is the forecasted load curve. The red dashed line is the objective curve. The dotted blue line represents the distribution of load demand by the uncontrollable devices in the forecasted load demand.

The reason for setting those instances is for better comparison with the results in [4] and also for analyzing the performance of algorithms by fixing the total load demand. The optimal management scheme is different among the instances. The default parameter settings of GA are the population size 200, the crossover rate 0.9, and the mutation rate 0.1. The algorithm terminates after 5000 generations and the solutions forming the optimal frontier are obtained.

In the rest of this section, the results obtained by the multiobjective GA will be compared with the single objective GA in [4] for the three cases. For each instance the algorithm is run 50 independent trials for analyzing the average performance of the algorithms.

A. Case 1 – Residential Area

For the single objective GA, its objective is to satisfy the Objective I. The fitness function of the single objective GA is set as

$$Fitness = \frac{1}{1+\sum_{t=1}^{m}(Pload(t) - Objective(t))^2} \quad (6)$$

as in [4] and the larger the fitness value the better the solution for the objective. For the multiobjective GA, both Objectives I and II are considered simultaneously.

The best and the mean solutions for the two algorithms are tabulated in Table II, where the better values are bold. The values in the table are the fitness values of the solutions, whereas the values in brackets are the corresponding electricity bill (cost) of the solution with the best fitness value. For the multiobjective GA, the best and the mean values are the solutions with the largest fitness values from the Optimal Pareto Frontier in each trial.

It can be observed that in nine of the ten instances, the best and the mean solutions obtained by the multiobjective GA are better than the single objective GA for the Objective I. The electricity bill (cost) of these best solutions for the Objective I is also better than those by the single objective GA. Fig. 5(a)-(j) illustrate the locations of the solutions obtained by the two

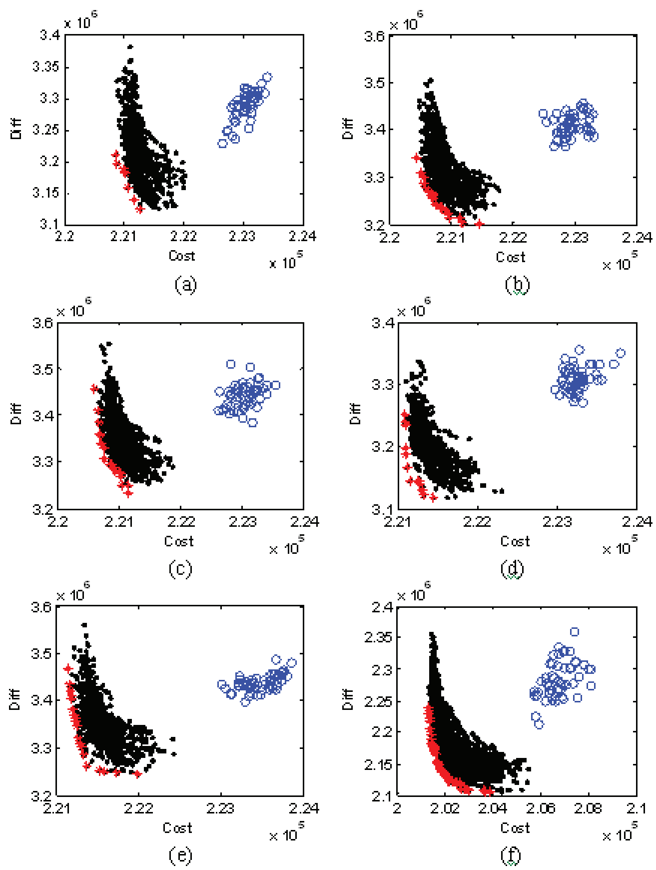
TABLE II. RESULTS OF CASE 1

	Single Objective GA		Multiobjective GA	
Instance	**Best**	**Mean**	**Best**	**Mean**
1	3.10E-07 (222658)	3.04E-07	**3.20E-07 (221261)**	**3.16E-07**
2	4.52E-07 (205917)	4.38E-07	**4.75E-07 (203874)**	**4.67E-07**
3	2.97E-07 (222658)	2.93E-07	**3.12E-07 (221440)**	**3.07E-07**
4	2.96E-07 (223163)	2.91E-07	**3.09E-07 (221160)**	**3.04E-07**
5	3.06E-07 (223324)	3.02E-07	**3.21E-07 (221453)**	**3.17E-07**
6	2.94E-07 (223300)	2.91E-07	**3.08E-07 (221987)**	**3.04E-07**
7	5.50E-07 (209938)	5.37E-07	**5.62E-07 (209498)**	**5.57E-07**
8	6.35E-07 (206694)	6.15E-07	**6.37E-07 (205678)**	**6.29E-07**
9	5.43E-07 (207859)	5.31E-07	**5.68E-07 (206229)**	**5.60E-07**
10	**6.81E-07** (210142)	**6.35E-07**	6.62E-07 **(209029)**	6.21E-07

algorithms in the 50 trials. For better showing the Pareto frontiers, the y-axis in the figures is

$$Diff = \frac{1}{Fitness} - 1 \quad (7)$$

which actually equals to the Objective I defined in (2) and its value is the smaller the better.



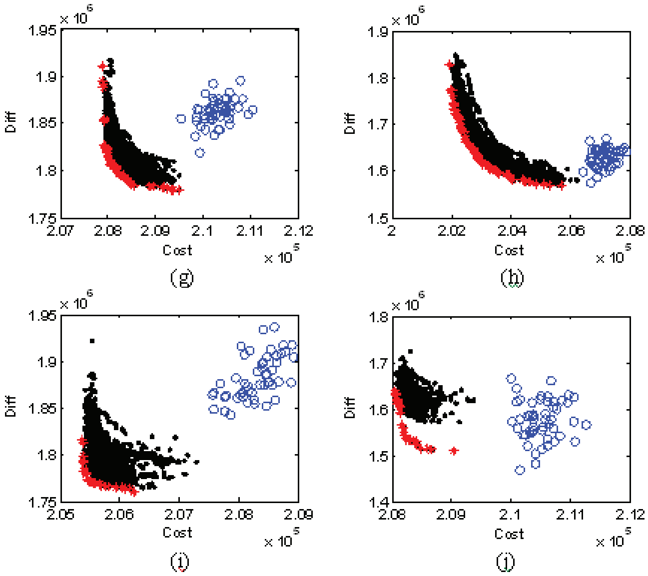


Figure 5. Comparisons of the solutions for the ten instances in Case 1 by the single objective GA and the multiobjective GA. The 50 blue hollow points represent the locations of the best solutions obtained by the single objective GA in the 50 trials. The solid black and red points are the solutions obtained by the multiobjective GA in the 50 trials, whereas the red points are the optimal frontier found by the algorithm.

Since the Diff and the Cost in Fig. 5 are the smaller the better, the figures show that the solutions obtained by the multiobjective GA are much better than those by the single objective GA for both objectives in the first nine instances. For the tenth instance, some solutions obtained by the single objective GA are better than the multiobjective GA for the Objective I, but the cost values of the solutions are larger.

B. *Case 2 – Commerical Area*

The best and the mean solutions for the two algorithms for Case 2 are tabulated in Table III. In this case, for five of the ten instances the multiobjective GA can get better solutions than the single objective GA. The solutions that the values for Objective I by the multiobjective GA are not better than the single objective GA still have better cost values.

TABLE III. RESULTS OF CASE 2

	Single Objective GA		Multiobjective GA	
Instance	**Best**	**Mean**	**Best**	**Mean**
1	9.04E-08 (352014)	8.95E-08	**9.41E-08 (349345)**	**9.27E-08**
2	1.33E-07 (316956)	1.32E-07	1.33E-07 **(315481)**	1.32E-07
3	9.31E-08 (351318)	9.20E-08	**9.65E-08 (349368)**	**9.55E-08**
4	8.94E-08 (351327)	8.85E-08	**9.26E-08 (349216)**	**9.17E-08**
5	9.28E-08 (350712)	9.17E-08	**9.59E-08 (349327)**	**9.49E-08**
6	9.06E-08 (350691)	8.96E-08	**9.40E-08 (349011)**	**9.32E-08**
7	**1.58E-07** (326967)	**1.57E-07**	1.57E-07 **(326945)**	1.56E-07
8	**1.63E-07** (335808)	**1.62E-07**	1.62E-07 **(335706)**	1.61E-07
9	1.44E-07 (340740)	1.43E-07	1.44E-07 **(340628)**	1.43E-07
10	**1.46E-07** (321026)	**1.45E-07**	1.45E-07 **(320824)**	1.44E-07

Fig. 6 shows the solutions obtained by the two algorithms for the ten instances in Case 2 in the 50 trials. Most of the solutions obtained by the multiobjective GA can have smaller values for Objective I than the single objective GA. For the instances that the values for Objective I by the multiobjective GA are larger, the cost values are still smaller than the single objective GA.

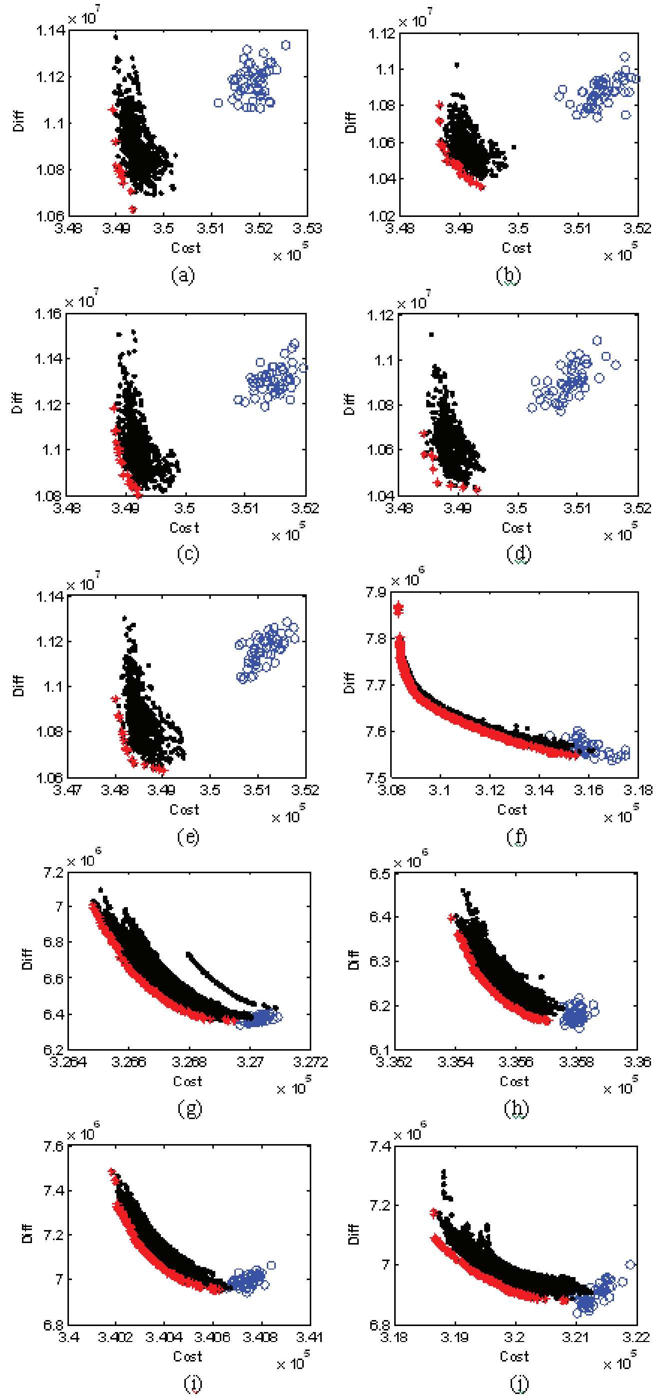


Figure 6. Comparisons of the solutions for the ten instances in Case 2 by the

single objective GA and the multiobjective GA. The 50 blue hollow points represent the locations of the best solutions obtained by the single objective GA in the 50 trials. The solid black and red points are the solutions obtained by the multiobjective GA in the 50 trials, whereas the red points are the optimal frontier found by the algorithm.

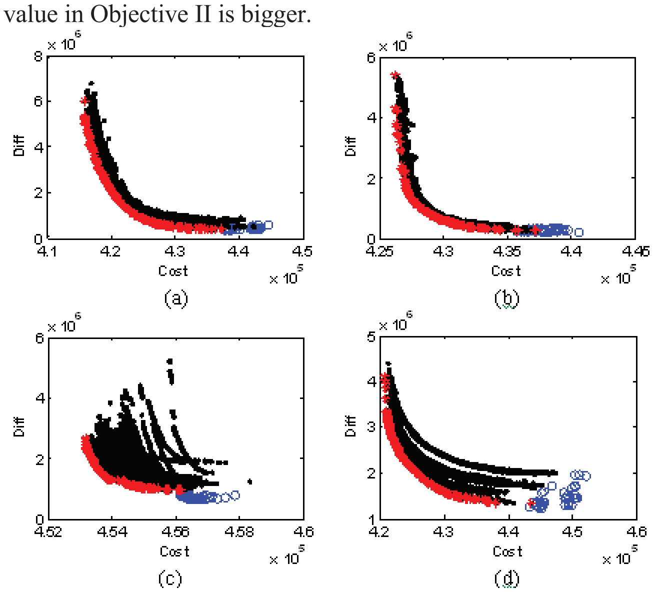
C. *Case 3 – Industrial Area*

The best and the mean solutions for the two algorithms for Case 3 are tabulated in Table IV. In this case, the solutions obtained by the single objective GA have better fitness values for Objective I than those obtained by the multiobjective GA for the ten instances. But for the Objective II, the solutions by the multiobjective GA are always better.

TABLE IV. RESULTS OF CASE 3

	Single Objective GA		**Multiobjective GA**	
Instance	**Best**	**Mean**	**Best**	**Mean**
1	**2.77E-06** (437733)	**2.27E-06**	2.38E-06 **(437125)**	1.62E-06
2	**4.86E-06** (440604)	**3.80E-06**	3.66E-06 **(437358)**	2.81E-06
3	**1.57E-06** (456825)	**1.38E-06**	1.06E-06 **(456188)**	8.32E-07
4	**8.21E-07** (444876)	**7.10E-07**	7.40E-07 **(443490)**	5.44E-07
5	**3.09E-07** (482050)	**3.01E-07**	2.89E-07 **(480823)**	2.60E-07
6	**1.46E-06** (452140)	**1.37E-06**	1.24E-06 **(451443)**	9.51E-07
7	**8.67E-07** (462028)	**8.13E-07**	7.61E-07 **(460623)**	6.46E-07
8	**5.23E-07** (472917)	**5.13E-07**	4.40E-07 **(471103)**	4.01E-07
9	**3.97E-07** (484394)	**3.79E-07**	3.65E-07 **(479894)**	3.34E-07
10	**3.79E-07** (444357)	**3.66E-07**	3.39E-07 **(442534)**	3.02E-07

For better investigating the performances of the two algorithms, the solutions of the 50 trials for Case 3 by the algorithms are presented in Fig. 7. The solutions obtained by the single objective GA always gather at the lower right corner, where the values for Objective I may be smaller but the cost value in Objective II is bigger.



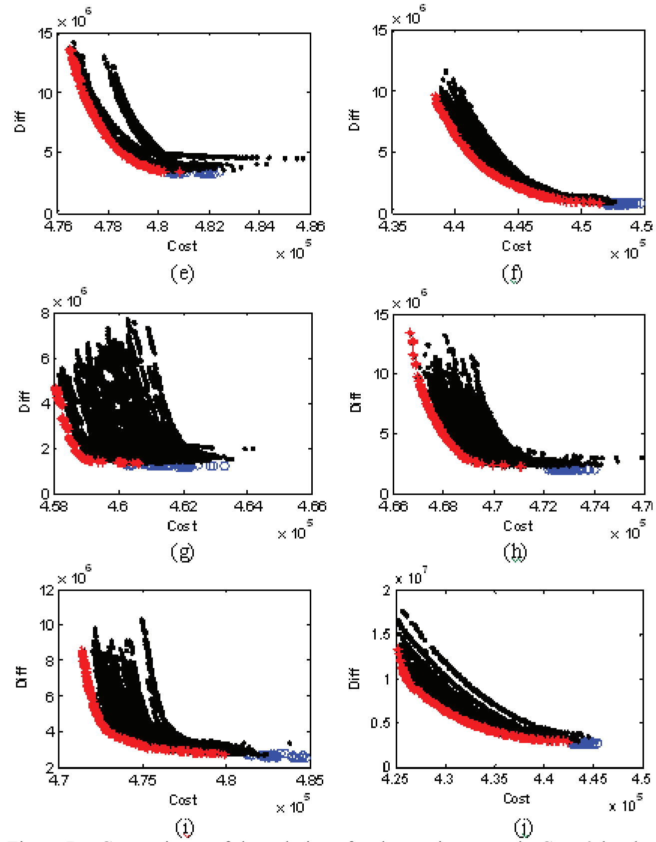


Figure 7. Comparisons of the solutions for the ten instances in Case 3 by the single objective GA and the multiobjective GA. The 50 blue hollow points represent the locations of the best solutions obtained by the single objective GA in the 50 trials. The solid black and red points are the solutions obtained by the multiobjective GA in the 50 trials, whereas the red points are the optimal frontier found by the algorithm.

From Fig. 7, it can be observed that in the frontier many solutions have similar small values for Objective I but the corresponding cost values in Objective II are smaller. Based on the results, utility companies can choose an acceptable solution with smaller cost values rather than the one with a larger cost value but the values for Objective I are slightly better. For solving the smart grid problem, multiobjective solutions are more applicable and appropriate for making decisions.

V. CONCLUSION

In this paper, the demand side management optimization problem is addressed as a multiobjective optimization problem by considering multiple objectives for satisfying the needs of different interest groups, e.g. the utility companies (suppliers) and the end users (consumers). A multiobjective genetic algorithm is proposed for solving the problem. The performance of the algorithm has been compared with the solutions obtained by the single objective problems in three sets of instances. The results show that the solutions found by the multiobjective algorithms are more reasonable and can better balance the needs of different interest groups.

REFERENCES

[1] H. Farhangi, "The path of the smart grid," IEEE Power and Energy Magazine, vol. 8, no. 1, pp. 18-28, 2010.

[2] I. Koutsopoulos and L. Tassiulas, "Challenges in demand load control for the smart grid," IEEE Network, vol. 25, no. 5, pp. 16-21, 2011.

[3] T. T. Kim and H. V. Poor, "Schdeuling power consumption with price uncertainty," IEEE Trans Smart Grid, vol. 2, no. 3, pp. 519-527, 2011.

[4] T. Logenthiran, D. Srinivasan, and T. Z. Shun, "Demand side management in smart grid using heuristic optimization," IEEE Trans Smart Grid, vol. 3, no. 3, pp. 1244-1252, 2012.

[5] A.-H. Mohsenian-Rad, V. W. S. Wong, J. Jatskevich, R. Schober, and A. Leon-Garcia, "Autonomous demand-side management based on game-theoretic energy consumption scheduling for the future smart grid," IEEE Trans Smart Grid, vol. 1, no. 3, pp. 320-331, 2010.

[6] P. Du and N. Lu, "Appliance commitment for household load scheduling," IEEE Trans Smart Grid, vol. 2, no. 2, pp. 411-419, 2011.

[7] J. H. Holland, Adaptation in Natural and Artificial Systems, Ann Arbor, MI: The University of Michigan Press, 1975.

[8] X-M. Hu and J. Zhang, "Ant colony optimizatin for enhancing scheduling reliability in wireless sensor networks," in Proc. 2012 IEEE Intern. Conf. Systems, Man, and Cybernetics, pp. 785-790, 2012.

[9] X.-M. Hu, J. Zhang, Y. Yu, H. S.-H. Chung, Y.-L. Li, Y.-H. Shi, and X.-N. Luo, "Hybrid genetic algorithm using a forward encoding scheme for lifetime maximization of wireless sensor networks," IEEE Trans. Evol. Comput., 766-781, vol. 14, no. 5, 2010.

[10] X.-M. Hu, J. Zhang, et al, "SamACO: variable sampling ant colony optimization algorithm for continuous optimization," IEEE Trans. Systems, Man, and Cybernetics--Part B, vol. 40, no. 6, pp.1555-1566, 2010.

[11] Z.H. Zhan, Jun Zhang, et al, "Multiple populations for multiple objectives: A co-evolutionary technique for solving multi-objective optimization problems ", IEEE Trans. Systems, Man, and Cybernetics--Part B: Cybernetics. (in press)

[12] W.-N. Chen, Jun Zhang, "Ant colony optimization for software project scheduling and staffing with an event-based scheduler," IEEE Trans. Software Engineering, (in press)

[13] X.-M. Hu, Jun Zhang, H. Chung, and J. Xiao, "An intelligent testing system embedded with an ant-colony-optimization-based test composition method," IEEE Trans. Systems, Man, and Cybernetics--Part C, Vol. 39, No. 6, pp. 659-669, 2009.

[14] J. Zhang, H. S.-H. Chung, and W.-L. Lo, "Clustering-based adaptive crossover and mutation probabilities for genetic algorithms," IEEE Trans. Evol. Comput., vol. 11, no. 3, pp. 326-335, Jun. 2007.

[15] K. Deb, A. Pratap, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," IEEE Trans. Evol. Comput., vol. 6, no. 2, pp. 182-197, 2002.