Improved Multi-objective Evolutionary Algorithm for Day-Ahead Thermal Generation Scheduling

Anupam Trivedi, N. M. Pindoriya, Dipti Srinivasan and Deepak Sharma
Department of Electrical and Computer Engineering
National University of Singapore (NUS), Singapore-117576
E-mail: eleat@nus.edu.sg, elepnm@nus.edu.sg, elesd@nus.edu.sg, eledeepa@nus.edu.sg

Abstract— **This paper presents a multi-objective evolutionary algorithm to solve the day-ahead thermal generation scheduling problem. The objective functions considered to model the scheduling problem are: 1) minimizing the system operation cost and 2) minimizing the emission cost. In the proposed algorithm, the chromosome is formulated as a binary unit commitment matrix (UCM) which stores the generator on/off states and a real power matrix (RPM) which stores the corresponding power dispatch. Problem specific binary genetic operators act on the binary UCM and real genetic operators act on the RPM to effectively explore the large binary and real search spaces separately. Heuristics are used in the initial population by seeding the random population with two Priority list (PL) based solutions for faster convergence. Intelligent repair operator based on PL is designed to repair the solutions for load demand equality constraint violation. The ranking, selection and elitism methods are borrowed from NSGA-II. The proposed algorithm is applied to a large scale 60 generating unit power system and the simulation results are presented and compared with our earlier algorithm [26]. The presented algorithm is found to outperform our earlier algorithm in terms of both convergence and spread in the final Pareto-optimal front.**

Keywords- Multi-objective generation scheduling; Evolutionary Algorithm; Unit Commitment

I. NOMENCLATURE

T_{max}	number of hours considered (scheduling horizon)
t	hourly time index
N	number of generating units
i	generating unit index
P_i^t	power generated by unit *i* at time *t*
$P_{max,i}$	rated upper limit generation of unit *i*
$P_{min,i}$	rated lower limit generation of unit *i*
u_i^t	binary commitment state of unit *i* at time *t* (= 1 if unit is committed at time *t*; = 0, otherwise)
f_i^t	fuel cost of *i*th unit at time *t* in $/h
SU_i^t	start-up cost of unit *i* at time *t*
SD_i^t	shut-down cost of unit *i* at time *t*
L^t	load demand at time *t*
R^t	system spinning reserve requirement at time *t*
MUT/MDT	minimum up/down time of unit *i*
a_i, b_i, c_i	fuel cost coefficients of the *i*th generator
a_{1i}, b_{1i}, c_{1i}	emission cost coefficients of the *i*th generator
HSC_i	Hot start-up cost of unit *i*
CSC_i	Cold start-up cost of unit *i*
$T_{ON,t}^i$ $/T_{OFF,t}^i$	continuously on/off time of unit up to hour *t*

II. INTRODUCTION

The thermal generation scheduling comprises of two tasks: one is the unit commitment, which determines the on/off schedules of thermal generators; other is the power dispatch which distributes the system load demand to the committed generators [1]. The optimal thermal generation scheduling requires effectively performing the above two tasks to meet the forecasted load demand over a particular time horizon, satisfying a large set of operating constraints and meeting certain objectives. Generally the only objective of the generation scheduling is to minimize the system operation cost and the problem is known as the classical Unit Commitment Problem (UCP).

Being a hard and challenging problem (it belongs to the set of NP-hard problems), and due to its economical importance (large operational costs are involved), the UC problem has for long been a matter of concern for the power generator companies. Over the years, a lot of research has focused on developing efficient UC algorithms which can be grouped as a) Deterministic techniques, b) Meta-heuristic techniques, c) Hybrid approaches based on deterministic and mete-heuristic techniques.

A good review on methods of generation scheduling in electric power systems is presented in [2]. Deterministic approaches such as priority list (PL) method [3, 4], dynamic programming (DP) [5], mixed-integer programming (MIP) [6], branch-and-bound method (BB) [7], and Lagrangian relaxation (LR) [8, 9] have been used for solving UCP. Among these methods, PL is simple and fast but achieves poor final solutions. DP method is flexible but computation time suffers from the "curse of dimensionality". MIP method fails when the number of unit increases because they require a large memory and suffer from great computational delay. The drawback of BB method is the exponential growth in the computation time for system of practical size. The LR method provides a faster solution but it suffers from numerical convergence and existence of duality gap. Recently, solution methods based on meta-heuristic techniques such as genetic algorithm (GA) [10-13], artificial neural network (ANN) [14], particle swarm optimization (PSO) [15-17] seem to provide promising results and are evolving to become the most widely used tools for solving the UCP. However, these meta-heuristic techniques require considerable amount of computational time to find near-optimal solutions for large scale UC problems. More recently, researchers have obtained even better solutions (i.e. solutions with improved objective) to UCP with hybrid techniques which are combination of meta-heuristic and deterministic techniques [18, 19].

However, the single economic objective in classical UCP can no longer be considered alone due to the environmental concerns that arise from the emissions produced by fossil-fuel based electric power plants. Generation of electricity from fossil fuel releases several contaminants, such as sulphur dioxides, nitrogen oxides and carbon dioxide into the atmosphere. Due to increasing public awareness of the environmental protection and the passage of the clean air act amendments of 1990 [20], the modern utilities have been forced to simultaneously optimize both economic and emission objectives.

Generation scheduling taking both system operation cost and emission into account is a constrained non-linear multi-objective optimization problem. In practice, finding trade-off solutions between economic and environmental objectives is a very difficult task. For this reason, in the past, instead of treating cost and emission as competing objectives, most of the approaches expressed the maximum allowable emission rates as constraints in the formulation of unit commitment and economic dispatch problems [21, 22]. The drawback in this approach is that no information about the trade-off solutions can be properly obtained. Another method for finding a compromised solution using conventional techniques converts the multi-objective problem into that with a single objective by assigning relative weights to each objective [23]. The drawback in this approach is that the trade-off solution obtained depends entirely upon the weights assigned to the two objectives and to obtain several trade-off solutions one needs to do multiple runs.

The recent direction is to handle both objectives simultaneously as competing objectives using evolutionary algorithms. A fuzzy multi-objective optimization technique developed in [24] handles the multiple objectives. But the solutions produced are suboptimal and moreover the algorithm does not provide a systematic framework for directing the search towards Pareto-optimal solutions. A heuristic guided evolutionary approach was presented in [25] to solve this problem. However, a large number of non-dominated solutions may get lost in this approach as the elitism property of evolutionary algorithms has not been included.

A recent approach for this problem is presented in [26]. Here the chromosome is formulated as a binary unit commitment matrix (UCM) for storing the generator on/off states. Lambda-iteration method is used to assign power dispatch to each chromosome (i.e, UCM) to form real power matrix (RPM). Problem specific genetic operators act on the binary unit commitment matrix. The shortcoming in this approach is that only the binary search space is effectively explored by the genetic operators but the real search space is not explored exhaustively by Lambda-iteration method. The consequence is that the Pareto-front obtained is discontinuous. The challenge in the existing generation scheduling methods and algorithms is to obtain trade-off solutions on the entire Pareto-optimal cost emission front. This was the motivation behind the conducted work in this paper; to obtain the entire Pareto-optimal front and to provide the decision maker with a wide range of solutions.

This paper proposes a multi-objective evolutionary algorithm (MOEA) for the day-ahead thermal generation scheduling problem. In the proposed algorithm, the chromosome is formulated as a binary unit commitment matrix (UCM) which stores the generator on/off states and a real power matrix (RPM) which stores the corresponding power dispatch. Here, binary genetic operators act on the binary UCM and real genetic operators act on the RPM to effectively explore the binary as well as real search space separately. Problem specific genetic operators which have been shown in literature to work well in UC problems have been adopted. The difference in the algorithm presented in this paper and the one presented in [26] lies in including the RPM as part of the chromosome on which genetic operators act unlike in [26] where the chromosome consists only of binary UCM and the power is dispatched using Lambda-iteration method. The aim of the presented algorithm is to exhaustively explore the real search space too by including the RPM as part of the chromosome so that a wider Pareto-optimal front can be obtained. The ranking, selection and elitism methods are borrowed from Non-dominated Sorting Genetic Algorithm – II (NSGA-II) [27]. Although many MOEAs exist in literature, but the ranking and crowding strategies of NSGA-II have been borrowed as NSGA-II is fast. NSGA-II has been shown to perform well to obtain good convergence and diversity on Pareto-optimal front for two-objective theoretical and real world problems. Heuristics are used in the initial population by seeding the random population with two Priority list based solutions. Intelligent repair operator based on Priority list has been designed to efficiently handle the load demand equality constraint. The proposed MOEA has been tested on large scale 60 generating unit power system and compared to the results obtained in [26].

In the remaining part of the paper, the multi-objective generation scheduling problem is formulated in section III. Thereafter, proposed algorithm is described in section IV. The results and discussions are presented in section V and the paper is concluded in section VI with future works.

III. MULTI-OBJECTIVE GENERATION SCHEDULING: PROBLEM FORMULATION

The problem of scheduling thermal power generators is formulated as a bi-objective optimization model. The two conflicting objectives are: 1) minimizing the system operation cost and 2) minimizing the emission cost, while satisfying all the equality and inequality constraints over the scheduling period. This problem is formulated, mathematically, in this section.

A. Objective Functions

1) Minimizing the system operation cost:

The total system operating cost includes the fuel cost of the scheduled generators and the transition cost over the entire scheduling horizon. The fuel cost at any given time interval is taken as quadratic function of the generator power output. The transition cost is the sum of the start-up cost and shut-down cost, which is associated with unit ON/OFF to OFF/ON transitions.

$$F_1 = \sum_{t=1}^{T_{max}} \sum_{i=1}^{N} f_i^t + SU_i^t + SD_i^t \quad (1)$$

The fuel cost f_i^t of ith generator unit at time t in \$/h is expressed as

$$f_i^t = a_i(P_i^t)^2 + b_i(P_i^t) + c_i \quad (2)$$

The start-up cost of a de-committed generating unit i at hour t depends on the off period of unit prior to start-up, $T_{OFF,t}^i$. In this paper, hot start-up costs are considered when the unit has been off for a number of periods smaller or equal to its cold start hours, and cold start start-up costs are considered otherwise. Shut-down costs have been set to zero for all instances.

2) Minimizing the emission cost:

The function representing the emission cost is similar to the function representing the fuel cost of generators and is expressed as

$$F_2 = \sum_{t=1}^{T_{max}} \sum_{i=1}^{N} a_{1i}(P_i^t)^2 + b_{1i}(P_i^t) + c_{1i} \quad (3)$$

B. Constraints

The solution must satisfy several system or unit related constraints as follows:

1. *System power balance:* the total power generation at time t must cover the total demand L^t. Hence,

$$\sum_{i=1}^{N} P_i^t = L^t, \quad t = 1,2,\dots,T_{max} \quad (4)$$

2. *System spinning reserve requirements:* a reserve is necessary to face in real-time possible sudden load increase due to a demand increase or to a failure of any of the working units. Hence,

$$\sum_{i=1}^{N} u_i^t P_{max,i} \geq L^t + R^t, \quad t = 1,2,\dots,T_{max} \quad (5)$$

3. *Unit minimum up/down times:* if a unit is off, it must remain off for atleast *MDT* periods of time. In the same way, if a unit is on it must remain on for atleast *MUT* periods of time.

$$\left[(u_i^t - u_i^{t-1})\left(T_{ON,t-1}^i - MUT\right)\right] \leq 0 \quad (6)$$

$$\left[(u_i^t - u_i^{t-1})\left(T_{OFF,t-1}^i - MDT\right)\right] \geq 0 \quad (7)$$

where the time counter for unit i which has been on or off at hour t, $T_{ON,t}^i$ or $T_{OFF,t}^i$ can be expressed as:

$$T_{ON,t}^i = \left(1 + T_{ON,t-1}^i\right)u_i^t$$

$$T_{OFF,t}^i = \left(1 + T_{OFF,t-1}^i\right)(1 - u_i^t)$$

4. *Unit generation limits:* for stable operation, the power output of each generator is restricted by lower and upper limits as follows:

$$P_{min,i} \leq P_i^t \leq P_{max,i}, \quad i = 1,2,\dots,N \quad (8)$$

IV. Proposed MOEA for Thermal Generation Scheduling

In this paper, the multi-objective thermal generation scheduling problem is solved using a MOEA with problem specific heuristics, crossover and mutation operators. The problem specific operators and the steps for the proposed algorithm are explained in this section.

A. Chromosome Formulation

The generation scheduling problem deals with discrete binary variables for the operating status (on/off) of the units and with continuous real variables for the hourly thermal power output of the units. For every chromosome, a $N \times T_{max}$ binary unit commitment matrix (UCM) represents the generator on/off schedule and a $N \times T_{max}$ real power matrix (RPM) represents the corresponding power dispatch.

B. Generation of initial population

The initial population is usually generated randomly. However, since this is a highly constrained problem, the algorithm always starts from the infeasible search space. A lot of time is wasted exploring the infeasible search space and the convergence is very slow. Including some priority list based solutions in the initial population provides some direction to the algorithm and increases the speed of convergence of the scheduling algorithm [28]. In this paper, the initial random population is generated randomly except for two solutions which are generated using Priority list (PL) and Lambda-iteration method. The UCM and RPM of one solution is generated using PL and Lambda-iteration method respectively based on fuel cost coefficients. Similarly, for the other solution, PL and Lambda-iteration method based on emission cost coefficients is used. The reason behind seeding the initial random population with only two PL based solutions is that these two solutions are enough to guide the algorithm towards the feasible space and more PL based solutions in initial population might lead to premature convergence.

C. Constraint handling and Function Evaluation

The UCM and the RPM of each chromosome are multiplied to form the resultant power matrix (Res.PM) which represents a chromosome's generation schedule. The spinning reserve constraint violation, minimum up-down time constraint violation and load demand equality constraint violation are calculated for each chromosome using the Res.PM. If a chromosome violates the load demand equality constraint, than an attempt is made to repair the chromosome modifying a strategy used in [29]. In this strategy the chromosomes are repaired using PL based on either fuel cost coefficients or emission cost coefficients (with equal probability). If the total power output of the committed units in an hour is less than the load demand for that hour then the power output of the units (in RPM) is increased in ascending order of the PL. The power output of the units (in RPM) is decreased in descending order of the PL if the total power output of the committed units in an hour is greater than the load demand for that hour. A check is always made to assure that the power output of the generators lies in their generation limits. After the repairing process is attempted, the UCM and the RPM of chromosome are again multiplied to form the resultant power matrix (Res.PM). The load demand constraint violation is re-calculated using the Res.PM. The spinning reserve constraint violation and load demand constraint violation are taken as the absolute values of constraint violation while for calculating the minimum up-down time constraint violation, a constant 100 is added for every time the constraint is violated. The spinning reserve constraint violation, minimum up-down time constraint violation and load demand constraint violation are then summed up to calculate the total amount of constraint violation. A solution is termed feasible if the total constraint violation is less than 10^{-5}. The objective functions F_1 and F_2 are then calculated for each chromosome using its Res.PM. The pseudo code for the load demand constraint violation repair operator is presented below. The tolerance value is set as 10^{-6}.

```
for chromosome = 1 : Popsize
  for time = 1 : Tmax
    for unit = 1 : N
      Ptotal = Ptotal + ResPM[unit][time]
    Pgap = LDemand[time] – Ptotal
    if(Pgap > tolerance)
      for priority = 1 : 1: 10
        for unit = 1 : N
          if(Priority[unit] == priority & ucm[unit][time] ==1)
            if(RPM[unit][time] < Pmax[unit])
              diff = Pmax[unit] - RPM[unit][time]
              if( Pgap >= diff )
                RPM[unit][time] = Pmax[unit]
                Pgap = Pgap – diff
              else
                RPM[unit][time] = RPM[unit][time] + Pgap
                Pgap = 0;
        if(Pgap < tolerance)
          break
    else if(Pgap <- tolerance)
      for priority = 10 : -1: 1
        for unit = 1 : N
          if(Priority[unit] == priority & ucm[unit][time] ==1)
            if(RPM[unit][time] > Pmin[unit])
              diff = RPM[unit][time] - Pmin[unit]
              if( abs(Pgap) >= diff )
                RPM[unit][time] = Pmin[unit]
                Pgap = Pgap + diff
              else
                RPM[unit][time] = RPM[unit][time] – abs(Pgap)
                Pgap = 0;
        if(abs(Pgap) < tolerance)
          break
```

D. Ranking and Selection

The chromosomes are ranked using the constrained-domination principle [30]. A solution x^i is said to constrain-dominate a solution x^j if any of the following conditions is true:

1. Solution x^i is feasible and solution x^j is not.
2. Solutions x^i and x^j are both infeasible, but solution x^i has a smaller constraint violation.
3. Solutions x^i and x^j are both feasible, but solution x^i pareto-dominates solution x^j.

Crowding distance is calculated only for the feasible solutions. Constrained-binary tournament method [30] is used as the selection method and is based on the same principles as the constrained-domination condition. Out of two parent solutions x^i and x^j, solution x^i wins the tournament and goes into the mating pool if x^i constrain-dominates x^j. If x^i and x^j are feasible and belong to the same front, then the one having larger crowding distance goes into the mating pool.

E. Crossover

The generation scheduling problem involves both discrete binary variables and continuous real variables. Hence, a binary and a real crossover are employed to explore both the search spaces. Standard crossover operators do not work well on the binary variables in the unit commitment problem. In this paper, a slightly modified window crossover operator as mentioned in [31] is used as the binary crossover. It works by randomly selecting two parents from the mating pool and then randomly selecting a window size. The unit commitment entries within the window portion are exchanged between the UCM of the two parents to generate the UCM of the two offsprings. SBX crossover [30] is applied on the RPM of the two parents to obtain the RPM of the two offsprings.

F. Mutation

1) Swap-window operator: The binary and real versions of this operator are applied separately on UCM and RPM of a chromosome respectively. It works by randomly selecting two units u_1, u_2, a time window of width *w* (hours) between 1 and T_{max} and a random window position. Then the entries of the two units u_1, u_2 included in the window are exchanged. The operator acts like a sophisticated mutation operator [10].

2) Window Mutation Operator: This operator works on the UCM of a chromosome by randomly selecting a unit, a time window of width *w* (hours) between 1 and T_{max} and a random window position. Then it mutates all the bits included in the window turning, all of them to either 1s or all of them to 0s with an equal probability [10].

3) Real Mutation operator: This operator is applied to the RPM of the chromosomes. It has been shown by Coello et al. in [32] to enrich the exploratory capabilities of the real PSO algorithm and prevent the algorithm from premature convergence. The mutation operator tries to explore with all the chromosomes at the beginning of the search. The percentage of the population that is affected by mutation decreases rapidly with generations according to a non-linear function. The enhanced exploratory capability of this real mutation operator lies in the fact that it is applied not only to the chromosomes

(RPM) but also to the range of each design variable (i.e the generation limits of units). The operator works on the range of power variable of the units by covering the full range of power variables at the beginning of the search and narrowing down the search using the same non-linear variation function. The real mutation operator presented in [32] is slightly modified in this paper to enhance the exploratory capability of mutation operator. The pseudo code of the real mutation operator employed is given below. In this code, *Var* represents the non-linear function mentioned above. The constant *alpha* is a fraction of the total generations and constant *beta* lies between *0-1*.

```
Var = (1 − currentgen/alpha)^(3/beta)
for each chromosome
    if(rand < Var)
        time = random(1,T_max)
        for each unit
            mutrange =(Pmax[unit] − Pmin[unit]) * Var
            ub = RPM[unit][time] + mutrange
            lb = RPM[unit][time] – mutrange
            if(ub > Pmax[unit] then ub = Pmax[unit]
            if(lb < Pmin[unit] then lb = Pmin[unit])
    RPM[unit] [time]  = realrandom(lb,ub)
```

The steps of the proposed MOEA are depicted in Fig. 1 and are described as follows:

1. Parameter-setting input: population size, generation number, crossover probability, distribution index for SBX crossover, swap-window operator probability, window mutation operator probability, alpha and beta.
2. In the initial population, the UCM and RPM of all chromosomes are randomly generated except for two chromosomes. The UCM and RPM of the two heuristic based solutions are generated using PL and Lambda iteration method respectively.
3. Repair operator is used to repair the chromosomes for load demand equality constraint violation. Constraint violation (spinning reserve, min up-down time, load demand) is calculated and summed up for each chromosome.
4. The UCM and the RPM of each chromosome are multiplied to form the resultant power matrix (Res.PM). The objective functions F_1 and F_2 are then calculated using the Res.PM.
5. Constrain-domination condition is used to rank the population into non-dominated fronts. Crowding distance is calculated for the chromosomes lying in the feasible search space.
6. Constrained-binary tournament method is used as the selection operator to form the mating pool.
7. Crossover works by randomly selecting two parents from the mating pool. Window crossover is applied on the UCM and SBX crossover on the RPM of the two parents to form the UCM and RPM of the two offspring chromosomes.
8. Swap-window operator, window-mutation operator and real mutation operator is then applied on the offspring chromosomes.
9. Steps 3-5 are performed for offspring chromosomes.
10. The parent chromosomes and offspring chromosomes are then combined and sorted in different non-domination levels according to constrain-domination condition.
11. The next generation is formed using the elitism principle as used in NSGA-II.
12. If termination condition is satisfied then obtain Pareto-optimal solutions else steps (5) to (11) are continued.

V. RESULTS AND DISCUSSIONS

To demonstrate the effectiveness of the proposed MOEA for day-ahead thermal generation scheduling problem, this paper considers the case study of a 60 unit test system. The 60 unit test system data was generated by reproducing the data for 10 unit system [10] six times. The algorithm was run for the following parameters. These parameters were selected through experiments.

- Population Size = 300
- Generation number = 50000
- Crossover probability = 0.6
- Variable crossover probability in SBX = 1
- Distribution index in SBX = 2
- Swap-window operator probability = 0.25
- Window mutation operator probability = 0.25
- Alpha = 5000
- Beta = 0.9

Fig. 2 shows the progress of the non-dominated front over the number of generations. The test system being very large, each chromosome has a 60 by 24 unit commitment matrix and a 60 by 24 real power matrix. Hence, the algorithm has to deal with a large number (1440) of both binary and real variables. Moreover, the problem being multi-objective, the search space is very vast and therefore a large population size is chosen. The vastness of the search space is the reason for algorithm to take 50000 generations to achieve convergence. The problem being highly constrained, the algorithm actually starts from infeasible search space and initially all the solutions are infeasible. The heuristic based intelligent solutions in the initial population and the problem specific genetic operators help move the algorithm into feasible search space quite rapidly (in around 70 generations). The heuristic repair operator based on PL, efficiently handles the load demand equality constraint. The window crossover applied on the UCM and the SBX crossover applied on the RPM help in achieving smooth convergence. The swap window operator, window mutation operator and real mutation operator prevent the algorithm from premature convergence and help in obtaining a wide spread in the final Pareto-optimal front.

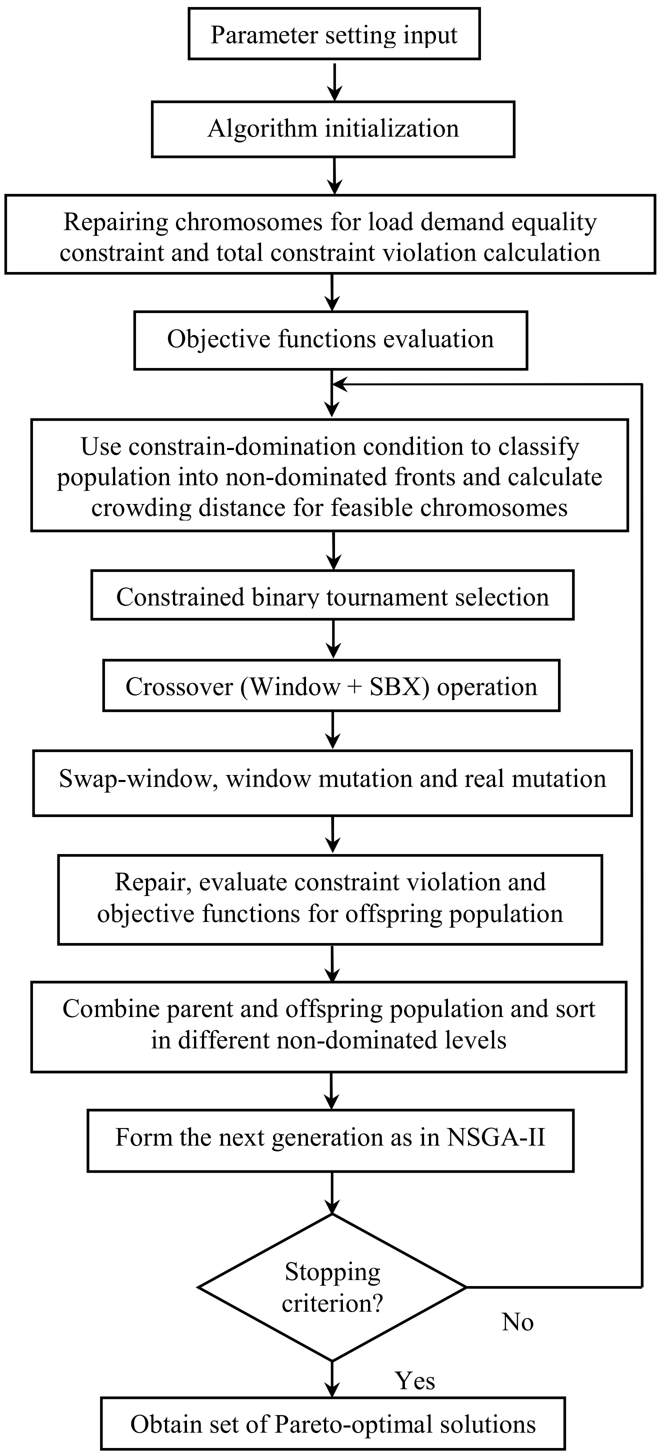

Figure 1: Flowchart of proposed approach

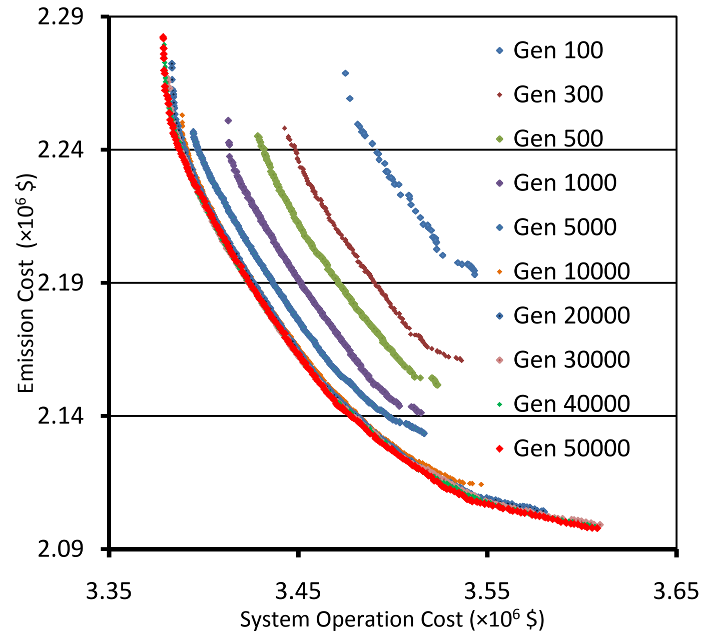

Figure 2: Improvement of non-dominated front over generations

The same 60 unit test system was solved using our earlier algorithm in [26]. As mentioned above, in this algorithm the chromosome was a binary unit commitment matrix and for every chromosome (i.e, UCM), a real matrix was used to store the corresponding power dispatch obtained using Lambda-iteration method. In this paper, the proposed MOEA considers both binary and continuous variables. The UCM represents the generator on/off schedules and the RPM represents the power dispatch of the corresponding units. The results of the proposed MOEA have been benchmarked against the results obtained in [26].

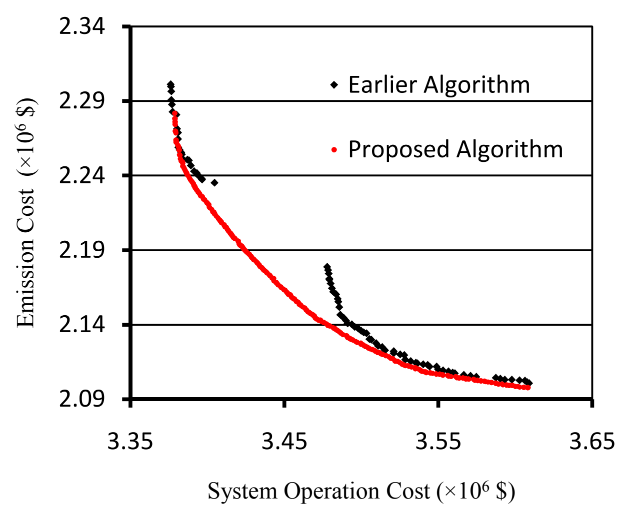

Figure 3: Comparison of results

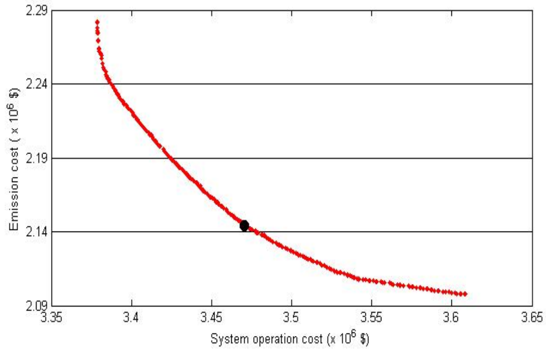

Figure 4: Solution chosen on Pareto-front using fuzzy membership function

Fig. 3 shows the comparison of the final Pareto-optimal fronts obtained using our earlier algorithm [26] and using proposed algorithm. It can be clearly seen that the Pareto-optimal front obtained using earlier algorithm is discontinuous while the one obtained using proposed algorithm is smooth and continuous. This demonstrates that by including RPM as part of the chromosome for handling power dispatch (instead of Lambda iteration method as used in [26]), we have been able to effectively explore the real search space (and thus objective space too) and obtain a wider spread of Pareto-optimal solutions.

After the entire Pareto-front is obtained, the Decision Maker has to choose a single trade-off solution from the final non-dominated solutions. One of the methods of choosing one single solution is to use linear fuzzy membership functions [33]. Fig. 4 represents the single trade-off solution (in black color) chosen from the wide range of Pareto-optimal solutions using linear fuzzy membership functions.

In this paper, the quantitative performance assessment of the proposed algorithm is done. The proposed MOEA is executed for 20 different runs and hypervolume indicator is used [34] that signify the proximity and spread of non-dominated front with respect to the reference set. The reference set of non-dominated solutions is obtained by choosing the best non-dominated solutions over 20 different runs. The statistical value of this indicator lies between -1 and 1, where -1 represents the best performance and 1 represents the worst performance [34]. As Table I shows, the mean value of hyeprvolume indicator is close to zero which shows a good performance of the proposed MOEA. The difference between the best and the worst values of hypervolume indicator is small that leads to a smaller value of standard deviation. This signifies the consistent performance of proposed MOEA for a large 60 unit system over 20 different runs.

All the simulations were performed using C platform on a PC with Intel Xeon 2.53GHz processor and 12 GB memory. The computational time of the algorithm for the parameters listed above was around 75 minutes.

Mean	Median	Best	Worst	Std.Dev
0.0305	0.0283	0.0133	0.0648	0.0146

Table I. Hypervolume indicator values

VI. Conclusion

A multi-objective evolutionary algorithm for day ahead thermal generation scheduling problem was proposed in this paper and solved for cost and emission objectives. In the proposed algorithm, the chromosome was formulated as a binary unit commitment matrix (UCM) which stored the generator on/off states and a real power matrix (RPM) which stored the corresponding power dispatch. The algorithm was tested on a large scale 60 unit system and the results were compared with the results obtained using our earlier algorithm [26]. It was observed that the proposed algorithm's performance is better in terms of both convergence and spread in the final Pareto-front. The hypervolume indicator was used for the qualitative performance assessment of the proposed algorithm. The statistical analysis showed that the proposed MOEA's performance was consistent with respect to the convergence and spread in the final Pareto-front. It is concluded that by including RPM as part of the chromosome along with UCM (instead of Lambda iteration method as used in [26]), the real search space could be explored effectively (and thus objective space too) to obtain a wider and well distributed Pareto-optimal front. The trade-off solutions were obtained along the entire Pareto-optimal cost emission front which provides the flexibility to decision maker to choose from a wider range of options.

At present this paper considered only thermal generators; however, the algorithm can be further modified for hybrid power systems which may include renewable energy sources along with thermal generators.

VII. Acknowledgement

This work is supported by National Research Foundation, NRF-2007EWT-CERP01-0954 (R-263-000-522-272).

References

[1] A. J. Wood and B. F. Woolenberg, *Power Generation, Operation, and Control*. New York: John Wiley & Sons, 1996.

[2] H. Y. Yamin, "Review on methods of generation scheduling in electric power systems," *Electric Power Systems Research,* vol. 69, pp. 227-248, 2004.

[3] T. Senjyu*, et al.*, "A fast technique for unit commitment problem by extended priority list," *IEEE Transactions on Power Systems,* vol. 18, pp. 882-888, 2003.

[4] T. Senjyu*, et al.*, "Emerging solution of large-scale unit commitment problem by stochastic priority list," *Electric Power Systems Research,* vol. 76, pp. 283-92, 2006.

[5] Z. Ouyang and S. M. Shahidehpour, "An intelligent dynamic programming for unit commitment application," *IEEE Transactions on Power Systems,* vol. 6, pp. 1203-9, 1991.

[6] H. Daneshi*, et al.*, "Mixed integer programming method to solve security constrained unit commitment with restricted operating zone limits," in *2008 IEEE International Conference on Electro/Information Technology, IEEE EIT 2008 Conference, May 18, 2008 - May 20, 2008*, Ames, IA, United states, 2008, pp. 187-192.

[7] A. I. Cohen and M. Yoshimura, "A branch and bound algorithm for unit commitment," *IEEE Transactions on Power Apparatus and Systems,* vol. PAS-102, pp. 444-51, 1983.

[8] S. J. Wang*, et al.*, "Short-term generation scheduling with transmission and environmental constraints using an augmented Lagrangian relaxation," *IEEE Transactions on Power Systems,* vol. 10, pp. 1294-301, 1995.

[9] W. Ongsakul and N. Petcharaks, "Unit commitment by enhanced adaptive Lagrangian relaxation," *IEEE Transactions on Power Systems,* vol. 19, pp. 620-8, 2004.

[10] S. A. Kazarlis*, et al.*, "A genetic algorithm solution to the unit commitment problem," *IEEE Transactions on Power Systems,* vol. 11, pp. 83-92, 1996.

[11] D. Dasgupta and D. R. McGregor, "Thermal unit commitment using genetic algorithms," *IEE Proceedings-Generation, Transmission and Distribution,* vol. 141, pp. 459-65, 1994.

[12] G. Dudek, "Unit commitment by genetic algorithm with specialized search operators," *Electric Power Systems Research,* vol. 72, pp. 299-308, 2004.

[13] I. G. Damousis*, et al.*, "A solution to the unit-commitment problem using integer-coded genetic algorithm," *IEEE Transactions on Power Systems,* vol. 19, pp. 1165-72, 2004.

[14] S. P. Valsan and K. S. Swarup, "Hopfield neural network approach to the solution of economic dispatch and unit commitment," in *Proceedings of International Conference on Intelligent Sensing and Information Processing, 4-7 Jan. 2004*, Piscataway, NJ, USA, 2004, pp. 311-16.

[15] G. Zwe-Lee, "Discrete particle swarm optimization algorithm for unit commitment," in *2003 IEEE Power Engineering Society General Meeting, 13-17 July 2003*, Piscataway, NJ, USA, 2003, pp. 418-24.

[16] L. Jin*, et al.*, "An improved binary particle swarm optimization for unit commitment problem," in *2010 Asia-Pacific Power and Energy Engineering Conference (APPEEC 2010), 28-31 March 2010*, Piscataway, NJ, USA, 2010, p. 4 pp.

[17] T. O. Ting*, et al.*, "Solving unit commitment problem using hybrid particle swarm optimization," *Journal of Heuristics,* vol. 9, pp. 507-20, 2003.

[18] C. Chuan-Ping*, et al.*, "Unit commitment by Lagrangian relaxation and genetic algorithms," *IEEE Transactions on Power Systems,* vol. 15, pp. 707-14, 2000.

[19] H. H. Balci and J. F. Valenzuela, "Scheduling electric power generators using particle swarm optimization combined with the Lagrangian relaxation method," *International Journal of Applied Mathematics and Computer Science,* vol. 14, pp. 411-21, 2004.

[20] A. A. El-Keib*, et al.*, "Economic dispatch in view of the Clean Air Act of 1990," *IEEE Transactions on Power Systems,* vol. 9, pp. 972-978, 1994.

[21] K. S. Swarup*, et al.*, "Genetic algorithm approach to environmental constrained optimal economic dispatch," *Engineering Intelligent Systems for Electrical Engineering and Communications,* vol. 4, pp. 11-23, 1996.

[22] J. H. Talaq*, et al.*, "A summary of environmental/economic dispatch algorithms," *IEEE Transactions on Power Systems,* vol. 9, pp. 1508-16, 1994.

[23] J. Nanda*, et al.*, "Economic-emission load dispatch through goal programming techniques," *IEEE Transactions on Energy Conversion,* vol. 3, pp. 26-32, 1988.

[24] Y. S. Brar*, et al.*, "Fuzzy satisfying multi-objective generation scheduling based on simplex weightage pattern search," *International Journal of Electrical Power & Energy Systems,* vol. 27, pp. 518-27, 2005.

[25] D. Srinivasan and A. Tettamanzi, "Heuristics-guided evolutionary approach to multiobjective generation scheduling," *IEE Proceedings-Generation, Transmission and Distribution,* vol. 143, pp. 553-9, 1996.

[26] A. Trivedi*, et al.*, "Modified NSGA-II for Day-Ahead Multi-objetive Thermal Generation Scheduling," *International Power Engineering Conference,* 2010.

[27] K. Deb*, et al.*, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation,* vol. 6, pp. 182-97, 2002.

[28] D. Srinivasan and J. Chazelas, "Heuristics-based evolutionary algorithm for solving unit commitment and dispatch," in *The 2005 IEEE Congress on Evolutionary Computation, 2-5 Sept. 2005*, Piscataway, NJ, USA, 2005, pp. 1547-54.

[29] S. N. Singh*, et al.*, "Application of advanced Particle Swarm Optimization techniques to wind-thermal coordination," in *2009 15th International Conference on Intelligent System Applications to Power Systems (ISAP), 8-12 Nov. 2009,* Piscataway, NJ, USA, 2009, p. 6 pp.

[30] K. Deb, *Multi-objective Optimization using Evolutionary Algorithms*. Singapore: John Wiley & Sons, 2001.

[31] J. Valenzuela and A. E. Smith, "A seeded memetic algorithm for large unit commitment problems," *Journal of Heuristics,* vol. 8, pp. 173-95, 2002.

[32] C. A. C. Coello*, et al.*, "Handling multiple objectives with particle swarm optimization," *IEEE Transactions on Evolutionary Computation,* vol. 8, pp. 256-79, 2004.

[33] M.Sakawa, "Fuzzy sets and interactive multiobjective optimization," *Fuzzy Sets and Systems,* vol. 59, pp. 324-325, 1993.

[34] E. Zitzler*, et al.*, "Performance assessment of multiobjective optimizers: An analysis and review," *IEEE Transactions on Evolutionary Computation,* vol. 7, pp. 117-132, 2003.