Evolutionary Optimisation for Power Generation Unit Loading Application

[1]Lily D Li and [2]Jiping Zhou

[1]School of Information Communication Technology, Central Queensland University, Australia
l.li@cqu.edu.au
[2]Stanwell Cooperation Limited, Australia
jiping.zhou@stanwell.com

Abstract

Power generation unit loading optimisation is a practically viable tool for efficiency improvement. The objectives for the coal-fired power generation loading optimisation are to minimize fuel consumption and to minimize emissions for a given load demand. This paper presents two models for this significant industrial application. Depending on the environmental regulation, either a single objective constrained model or a multi-objective constrained model can be chosen in practice. A multi-objective constraint-handling method incorporating the constraint dominance concept via Particle Swarm Optimisation (PSO) algorithm has been adopted for problem solving. The simulation results based on a coal-fired power plant demonstrates the capability, effectiveness and efficiency of using the proposed approach in a large scale industrial application.

Keywords: *Power Generation; Evolutionary Optimisation; PSO Algorithm; Load Distribution;*

1. Introduction

Most power generation plants have a number of generating units. How to make the best use of the units directly affects a company's business bottom line. Increased pressures from environmental regulations, rising fuel costs, and green house gas emissions demand power generators to be more efficient and effective.

Power plant efficiency improvement activities can be classified into two categories – plant modification (often irreversible) and operational improvement (often reversible). Traditional performance improvement activities have been often linked to plant modifications and large capital investment. Those performance improvement based on large capital investment are not risk free. Even successfully done, they do not always materialize the promised benefits. For example, when a unit is upgraded to suit higher load, it will not be as efficient when load demand is low. Frequent changes in market strategy often require reversible changes such as operational changes rather than irreversible plant modifications. In contrast with plant modification, operational improvement is low-risk, low-cost and often with instant benefits. Power generation unit loading optimisation is a practically viable tool for operational improvement.

A power generation plant usually has a number of units that work together. Generally, a power generation company has a *m*-year (or *m*-month) overhaul system, that is, each time, a unit goes through a major overhaul in turn and every *m* years (or *m*-months) the plant completes an overhaul cycle. The unit which was overhauled most recently would have the highest thermal efficiency and the one close to an overhaul will have the lowest thermal efficiency. Units with higher thermal efficiency will consume less fuel and cause less environmental harm while units with lower thermal efficiency will consume more fuel and lead to higher environmental harm. In the normal operation range, unit thermal efficiency increases (or heat rate decreases) as load increases. The thermal efficiency for each unit is different depending on when the unit is last overhauled, what kind of problems it developed, what modifications it went through, and what operation mode a unit is operating under (such as mill pattern). The optimised loading can be achieved based on the units' thermal efficiency and emission characteristics, that is, heat rate/NO_x vs. load, for a given plant condition.

The generation of electricity from fossil fuel release several contaminants, such as SO_2, NO_x and CO_2, into the atmosphere. Among these contaminants, nitrogen oxides or NO_x are contributed largely by the power stations and therefore are heavily regulated by the environmental management such as Department of Energy and Resource management (DERM) in Queensland [1]. In this research, NO_x emission is taken as a selected index for environment conservation. The methodology can be extended to other emissions such as particulates or CO_2.

The unit loading optimisation problem has been studied in a branch called "Economic Load Dispatch" over the years. Many of the earlier models only consider one objective, that is, heat consumption (treated as production cost), and using traditional deterministic approaches [2-5] for solving the problem. These days, society demands adequate and secure electricity not only at the cheapest possible price but also at minimum levels of pollution [6]. Minimizing atmospheric pollution will be one of the major challenges for electricity utilities. Several studies have been conducted in using stochastic metaheuristic methods for power generation loading optimisation application [6-8]. Zhao and Cao [7] proposed to use Particle Swarm Optimisation [9] algorithm and some fuzzy rules to solve the multi-objective optimisation problem. They proposed to use two external repositories and to use a geographically-based approach to find the Pareto-optimal solutions. The method for constraint-handing is not mentioned in their paper. The two external repositories made the computation complicated and inefficient. Basu et al. [6, 8] reported their approach of using evolutionary programming (genetic algorithm) and fuzzy satisfying method for the problem. The constraint-handling method is not clearly mentioned in the papers. The focus is put on the interactive fuzzy method to select solutions. These approaches demonstrated the capabilities of using metaheuristics for the multi-objective load dispatch optimisation problem.

The scope of this study is focused on the thermal operation side.

There are two objectives in the power generation loading optimisation application. One is to minimize the total heat consumption (fuel consumption) and another is to minimize the total NO_x emission. It is desirable that the unit with higher thermal efficiency (lower heat rate) receives higher workload and the unit with lower thermal efficiency (higher heat rate) receives lower workload.

This paper proposes two optimisation models in order to provide flexibility for practical operation. The first model (Model 1) treats emission as an additional constraint and considers the application a single objective optimisation problem so that to simplify the problem solving. The second model (Model 2) treats emission as an additional objective and considers the application a multi-objective optimisation problem. The first model applies in the situation where emission value must be limited in the certain value by the environment regulation. The second model applies for the other situations. A multi-objective constraint-handling method incorporating with PSO-based approach is adopted in the study.

The rest of the paper is organized as following: The problem modelling and formulation are presented in the section 2. Section 3 presents the proposed approaches for the two models. Section 4 presents the simulation results and discussion. Section 5 concludes the paper.

2. Modelling and formulation

Table 1 introduces the notation of the power generation loading optimisation problem. The detailed definitions for these terms are followed.

Table 1. Notation of power generation loading optimisation

Symbol	Meaning
M_{total}	total power demand by the market, total workload (MW)
M_{min}	lowest workload (MW)
M_{max}	highest workload (MW)
Q	total NO_x emission for all units at a given load (g/m^3)
F	total units heat consumption (MJ / h)
a	coefficients of the polynomial to heat rate function
b	coefficients of the polynomial to emission curve function
f	unit heat rate, is the heat consumption for generating per unit electricity (KJ/KW.h)
g	output demand constraint function (MW)
h	heat consumption per hour to a unit at a given load (MJ / h)
i	generation unit index (subscript)
k	order of polynomial function (superscript)
n	number of generation unit
P	maximum NO_x emission license limit to each unit (g/m^3)
q	NO_x emission level to a unit at a given load (g/m^3)
r	NO_x emission constraint function (g/m^3)
x	workload allocated to a unit (MW)
δ	minimum error criterion for equality constraint
ε	tolerance allowed for feasibility

Term Definition:

- For a given condition, a unit's heat rate f_i is depending on the unit load x_i which can be expressed by a polynomial format. This function is obtained from field testing and unit modelling. The general expression for the heat rate function for unit i is

$$f_i(x_i) = a_{ik}\ x_i^{\ k} + a_{i(k-1)}\ x_i^{\ (k-1)} + \ldots + a_{i1}\ x_i + a_{i0} \tag{1}$$

- A unit's heat consumption h_i at a given load x_i is calculated by

$$h_i = x_i\ f_i(x_i) \tag{2}$$

- Each unit has its own NO_x emission curve q. It is generally a linear function in the normal operation range, which is obtained from the field testing and unit modelling.

$$q_i(x_i) = b_{i1}\ x_i + b_{i0} \tag{3}$$

- The total heat consumption is the sum of all units' heat consumption, which can be expressed as the following

$$F(x) = \sum_{i=1}^{n} h_i = \sum_{i=1}^{n} x_i f_i(x_i) \tag{4}$$

- The total workload is the total power generated by all units at a given time.

$$M_{total} = \sum_{i=1}^{n} x_i \tag{5}$$

- The NO_x gas emission for each unit has to be restricted within a license limit *P*.

$$q_i(x_i) \le P \quad (i = 1, 2, \ldots n) \tag{6}$$

- The total NO_x gas emission is

$$Q(x_i) = \sum_{i=1}^{n} q_i(x_i) \tag{7}$$

- For stable operation, the workload for each unit must be restricted within its lower and upper limits. This is the range where a unit load can be readily adjusted without excessive human intervention. For example, a unit is operating between 60% to 100% load without the need of mill change.

$$M_{i\min} \le x_i \le M_{i\max} \quad (i = 1, 2, \ldots n) \tag{8}$$

Several constraints should be taken into consideration.

- The first one is that the total power generated should meet the market demand at a given time. Considering that the data types have to be implemented in double precision, this constraint can be rewritten as

$$g(x) = \left| \sum_{i=1}^{n} x_i - M_{total} \right| < \delta \ . \tag{9}$$

- The second set of constraints is the NO_x gas emission. For some regions, there is an environmental licence limit applied in practice. The licence specifies the maximum amount of NO_x gas emission allowed for each thermal unit. In this case, the constraints can be written as

$$r_i(x) = q_i(x_i) - P \le 0 \quad (i = 1, 2, \ldots n) \ . \tag{10}$$

Note: If there is no environmental licence applied, these constraints in equation (10) can be disregarded.

- The third constraint is the unit capacity constraint which can be modelled as the boundary constraint in the optimisation.

Depending on the environment regulation, the optimisation problem can take two different models – the single objective constrained model (SOCM) and the multi-objective constrained model (MOCM), as follows.

Model 1 – Single objective constrained model:

Find the optimal load distribution so as to minimize the total heat consumption *F*(*x*) subject to the total gas emission is restricted in the licensed limit and the output demand is satisfied, as described in Equation (11).

$$\begin{cases} \min & F(x) = \sum_{i=1}^{n} h_i = \sum_{i=1}^{n} x_i f_i(x_i) \\ \text{s.t.} & g_1(x) = \left| \sum_{i=1}^{n} x_i - M_{total} \right| - \delta \le 0 \\ & r_i(x) = q_i(x_i) - P \le 0 \quad (i = 1, 2, \ldots n) \\ \text{where} & f_i(x_i) = a_{ik} x_i^{k} + a_{i(k-1)} x_i^{(k-1)} + \ldots + a_{i1} x_i + a_{i0} \\ & q_i(x_i) = b_{i1} x_i + b_{i0} \\ & M_{i\min} \le x_i \le M_{i\max} \quad (i = 1, 2, \ldots n) \end{cases} \tag{11}$$

Model 2 – Multi-objective constrained model:

Find the optimal load distribution so as to minimize the total heat consumption $F(x)$ and the total NO_x gas emission $Q(x)$ subject to the output demand is satisfied, as described in Equation (12).

$$\begin{cases} \min & F(x) = \sum_{i=1}^{n} h_i = \sum_{i=1}^{n} x_i f_i(x_i) \\ & Q(x) = \sum_{i=1}^{n} q_i(x_i) \\ \text{s.t.} & g_1(x) = \left| \sum_{i=1}^{n} x_i - M_{total} \right| - \delta \le 0 \\ \text{where} & f_i(x_i) = a_{ik} x_i^{k} + a_{i(k-1)} x_i^{(k-1)} + \ldots + a_{i1} x_i + a_{i0} \\ & q_i(x_i) = b_{i1} x_i + b_{i0} \\ & M_{i\min} \le x_i \le M_{i\max} \quad (i = 1, 2, \ldots n) \end{cases} \tag{12}$$

3. The proposed approach

3.1. The PSO algorithm

Particle Swarm Optimisation is a stochastic metaheuristic method for optimising numerical functions on the metaphor of social behaviours of flocks of birds and schools of fish [10]. A PSO algorithm consists of individuals, called particles that form a swarm. Each particle represents a candidate solution to the problem. Particles change their positions by flying in a multi-dimensional search space looking for the optimal position. During flight, each particle adjusts its position according to its own experience and the experience from its neighbouring particles, making use of the best position encountered by itself and the best position in the entire population (global PSO) or its local neighbourhood (local PSO). The performance of each particle is measured by a predefined fitness function (objective function) which is problem-dependent.

Let *i*-th particle in a D-dimensional search space be represented as $x_i = (x_{i1}, x_{i2}, \ldots x_{iD})$. The best previous position of the *i*-th particle in the flight history is $pBest_i = (p_{i1}, p_{i2}, \ldots p_{iD})$. The position of the best particle of the neighbourhood is $lBest_i = (p_{g1}, p_{g2}, \ldots p_{gD})$. The velocity for particle *i* is $v_i = (v_{i1}, v_{i2}, \ldots v_{iD})$. In the PSO algorithm, the next position *(t+1)* of the particle *i* on the dimension *d* is manipulated by the following equations (*t* denote the iteration):

$$v_{id}^{(t+1)} = \chi[wv_{id}^{(t)} + c_1 r_{1id}^{(t)}(pBest_{id}^{(t)} - x_{id}^{(t)}) + c_2 r_{2id}^{(t)}(lBest_{id}^{(t)} - x_{id}^{(t)})] \quad (13)$$

$$x_{id}^{(t+1)} = x_{id}^{(t)} + v_{id}^{(t+1)} \quad (14)$$

where $d = 1,2,\ldots D$, D is the search dimension; $i = 1,2,\ldots N$, and N is the number of particles in the swarm; w is the inertia weight; χ is a constriction coefficient; c_1 and c_2 are two positive constants, called the cognitive and social parameters respectively; r_1 and r_2 are two random numbers within the range [0, 1].

The original PSO algorithm and its variations have no mechanism to handle constraints. In order to integrate constraints handling with PSO, we introduce the constraint handling methods in the following sections.

3.2. Constraint handling

Multi-objective constraint-handling method has been studied in evolutionary algorithms [11-14], in which a single objective optimisation problem can be transformed into a bi-objective problem where the first objective is to optimise the original objective function and the second is to minimize

$$\Phi(x) = \sum_{i=1}^{m} \max(0, g_i(x)) \quad (15)$$

where $\Phi(x)$ is a total amount of constraint violations; m is number of constraints and $g_i(x)$ is the i-th constraint function. From the above equation, if a solution satisfies all constraints, that is, $g_i(x) \le 0$ for $i = 1, 2, \ldots, m$, $\Phi(x)$ returns a zero. Otherwise, it returns a positive number indicating the total amount of constraint violations. Thus, the smaller the constraint violation, the more feasible the solution is. The optimal value for constraint violations is 0.

Another concept adopted in the research is the constraint dominance concept, which can be described as following:

> *"A solution $x^{(1)}$ is said to 'constraint-dominate' a solution $x^{(2)}$, if any of the following conditions are true:*
> - *Solution $x^{(1)}$ is feasible and solution $x^{(2)}$ is not.*
> - *Solution $x^{(1)}$ and solution $x^{(2)}$ are both infeasible, but solution $x^{(1)}$ has a smaller constraint violation.*
> - *Solution $x^{(1)}$ and solution $x^{(2)}$ are feasible and solution $x^{(1)}$ dominates solution $x^{(2)}$ in the usual sense"* [15].

Constraint dominance concept indicates that non-dominated solutions are better than dominated solutions. This concept will be used in comparing particles for the PSO algorithm.

3.3. The modified constrained PSO algorithm

By adopting the multi-objective constraint-handling method, Model 1 in equation (11) can be transformed into:

$$
\begin{cases}
\min & F_1(x) = (F(x), \Phi(x)) \\
\text{where} & F(x) = \sum_{i=1}^{n} x_i f_i(x_i) \\
 & \Phi(x) = \max(0,\ g_1(x)) + \sum_{i=1}^{n} \max\ (0,\ r_i(x)) \\
 & g_1(x) = \ \left| \sum_{i=1}^{n} x_i - M_{total} \right| - \delta \\
 & r_i\ (x)\ =\ q_i(x_i)\ -\ P \quad (i = 1,\ 2, ...,\ n)
\end{cases} \tag{16}
$$

Model 2 in equation (12) can be transformed into:

$$
\begin{cases}
\min & F(x) = \sum_{i=1}^{n} x_i f_i(x_i) \\
 & Q(x) = \sum_{i=1}^{n} q_i(x_i) \\
\text{s.t.} & \Phi(x) = \max(0,\ g_1(x)) \le \varepsilon \\
\text{where} & g_1(x) = \ \left| \sum_{i=1}^{n} x_i - M_{total} \right| - \delta
\end{cases} \tag{17}
$$

The equation (16) is a bi-objective unconstrained optimisation model. The equation (17) is a bi-objective with one constraint.

Table 2 illustrates the proposed modified PSO algorithm. It integrates the multi-objective constraint-handling method and the constraint dominance concept into PSO algorithm.

Compared with the original PSO algorithm, the modified algorithm has the following features:

- When calculating fitness, the objective function F, constraint violation Φ and emission function Q (only for the Model 2) are calculated;
- A particle's personal best solution and the local best solution in its neighbourhood are determined by constraint-dominance concept;
- For the Model 1, the output is one single solution. The solution has a minimal F value and satisfies $\Phi \le \varepsilon$.
- For the Model 2, the output contains a set of non-dominated particles.

Table 2. Structure of the modified PSO algorithm for constrained optimisation problems

```
01    Uniformly initialize particles
02    Calculate fitness values F
03*   Calculate constraint violation Φ
04**  Calculate emission value Q (Model 2 only)
05    Set current locations as personal best locations for all particles
06    Set local best location for each particle according to constraint-dominance concept
07    Do
08        For each particle
09            Calculate new velocity by PSO formula
10            Calculate new location by PSO formula
11                If the new location is better than the personal best location (according to constraint-dominance concept)
12                Update the personal best location with the new location
13        End For
14        Set local best location for each particle (according to constraint-dominance concept)
15    While maximum iteration is reached
16    Output results
```

Note: * The formulas for calculating constraint violation Φ are different for Model 1 and Model 2.
** Step 04 is for Model 2 only, if Model 1 is taken, this step is discarded

4. Simulation Results and Discussion

4.1. Unit Heat Rates and Emission Functions

A local power plant has four 360MW generator units and a total 1440MW of generation capacity. The power plant has a four-year overhaul system. Each year, a unit goes through a major overhaul in turn and every four year the plant completes an overhaul cycle.

The boundary constraints $M_{\min}$ and $M_{\max}$ for each unit are 220 (MW) and 360 (MW). The full load output ranges from $4\times 220=880$ (MW) as the minimum to $4\times 360=1440$ (MW) as the maximum. It would be better to simulate a series of output (a series of M_{total}) in order to allow the power plant to choose from the optimal results according to the market demand.

The heat rate functions and the NO_x emission functions for the four generator units are provided from a local power plant setting. The heat rate functions are in the polynomial format with the power of two. The NO_x emission functions are linear. Table 3 lists the sample functions. These functions can be modified when the units' performance are changed.

Table 3. Unit heat rate and NO_x emission functions*

Unit No.	Heat Rate	NO_x Emission
1	$f(x_1)=0.0023x_1^2-3.7835x_1+9021.7$	$q(x_1)=0.0036x_1-0.1717$
2	$f(x_2)=0.0238x_2^2-9.7773x_2+9432.6$	$q(x_2)=0.0031x_2-0.0226$
3	$f(x_3)=0.0187x_3^2-5.3678x_3+10240.0$	$q(x_3)=0.0036x_3-0.1252$
4	$f(x_4)=0.0120x_4^2-5.7450x_4+9231.7$	$q(x_4)=0.0039x_4-0.1706$

*Due to commercial reasons, the functions have been slightly modified.

4.2. Parameter Setting

The minimum error criterion for equality constraint is selected as $\delta=1.0E-03$. The NO_x license limits P is 1.3. PSO neighbourhood topology is set to static ring topology with the neighbour size of 2. PSO parameters are: $w=0$; $c_1=c_2=2$; $\chi=0.63$; $V_{\max}=0.5\times(M_{\max}-M_{\min})$; population size is 40 for Model 1 and 500 for Model 2; the maximum iteration is set to 10,000 for Model 1 and 1000 for Model 2. The feasibility tolerance allowed $\varepsilon=1.0E-08$, that is, if a total amount of constraint violation $\Phi\le\varepsilon$, the solution is considered feasible.

For each total load output M_{total} , the program runs ten times with the best solution recorded. For the Model 1, the best solution means the feasible solution that has the lowest heat consumption. For the Model 2, the best solution is a set of Pareto-optimal solutions that has a small "Spacing/Spread" value (see Appendix for the definition of Spacing/Spread).

4.3. Results

Table 4 and Figure 1 present the simulation results to the whole range of the generation capacity. For each total output demand, the optimal workloads to the four generators have been found based on their efficiency functions as listed in Table 3. After optimisation, the unit with higher thermal efficiency will receive a higher workload (such as Unit 1) while the unit with lower thermal efficiency will receive a lower workload (such as Unit 3). In practice, when the total output load changes, the optimal load allocation can be adjusted from these results.

The results for the Model 2 should have a series of figures corresponding to the different output capacities. That is, for a specific output demand, the optimisation result should be a set of Pareto-optimal solutions. The user needs to select a specific solution by compromising both objectives. The corresponding load distribution can then be found from the selected solution. How to compromise the objectives is not in the scope of this study.

Figure 2 and Figure 3 illustrate the simulation results for $M_{total} = 1000$ MW and $M_{total} = 1200$ MW.

It needs to be mentioned that the objective *F* (Heat Consumption) conflicts with and the objective Q (NO_x emission). NO_x is a product of combustion. NO_x increases as combustion temperature and oxygen increases. While the better combustion consumes less fuel and produces better efficiency and lower heat rate; and better combustion generates high NO_x levels as the flame would be hotter and oxygen would be more.

Table 4. Optimised workload distribution for Model 1

M_{total} (MW)	Unit 1 (MW)	Unit 2 (MW)	Unit 3 (MW)	Unit 4 (MW)
880	220.0000	220.0000	220.0000	220.000
900	235.1297	222.3830	220.0131	222.4746
950	273.1694	220.7023	220.0023	236.1254
1000	326.7896	230.9750	220.0002	222.2353
1050	359.6199	224.8249	221.3988	244.1572
1100	359.9975	227.8653	221.2147	290.9222
1150	359.8885	235.5493	221.1815	333.3800
1200	359.9999	269.2535	247.5140	323.2334
1250	359.9937	326.3954	221.1032	342.5064
1300	358.9863	339.9221	241.0938	359.9969
1350	359.9939	325.3112	305.7268	358.9679
1400	358.4181	353.5454	328.0362	359.9999
1440	360.0000	360.0000	360.0000	360.0000

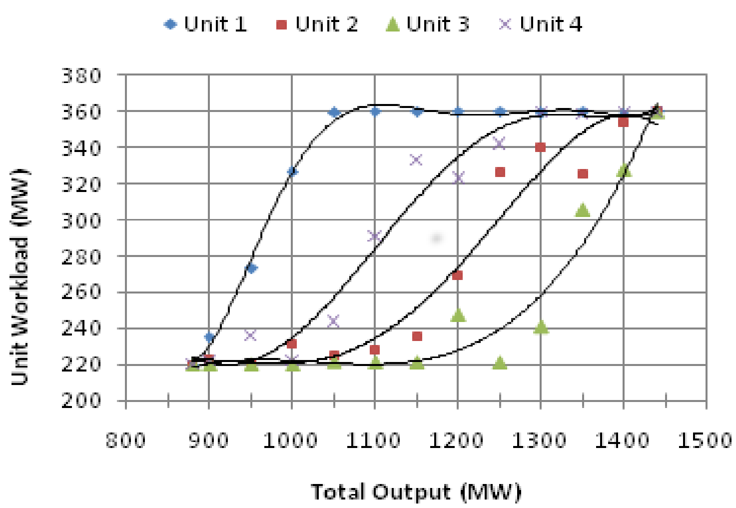

Figure 1. Optimal unit loading distribution for the whole range of the generation capacity for Model 1

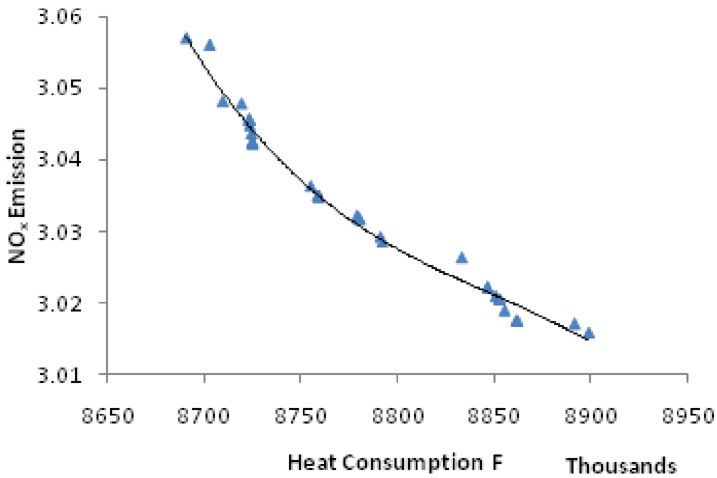

Figure 2. Simulated Pareto-front for Model 2 ($M_{total} = 1000\text{MW}$)

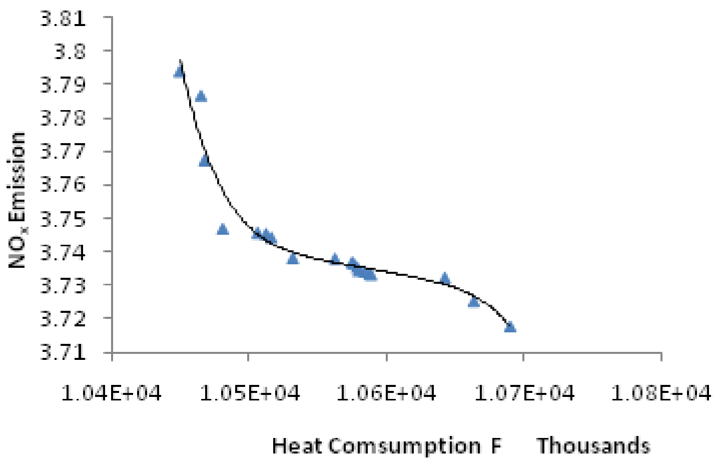

Figure 3. Simulated Pareto-front for Model 2 ($M_{total} = 1200\text{MW}$)

4.4. Discussion

Money saving from the optimisation is calculated in order to assess the significance of the loading optimisation. Without adopting the loading optimisation, the total market demand is averagely allocated to the four generator units. By adopting the optimisation, the total market demand is allocated to each generator unit based on each unit's thermal performance. The heat consumption saving is then calculated for comparing the difference between the two. The formula is:

$$H_{saving} = \sum_{i=1}^{4} x_{avg} f_i(x_{avg}) - \sum_{i=1}^{4} x_i f_i(x_i) \qquad (18)$$

where the $x_{avg} = M_{total} / 4$, the f_i is the heat rate curve listed in Table 3.

The annual money saving from the optimisation can be calculated by using the calorific value and the price of fuel. Suppose the fuel (coal) price is $\cos t = AU\$30$ per ton, the calorific value $c = 26$ MJ/kg,

$$M_{saving} = \frac{H_{saving}}{c \times 1000} \times \cos t \times 24 \times 360 \qquad (19)$$

The result for the annual money saving is illustrated in Figure 4.

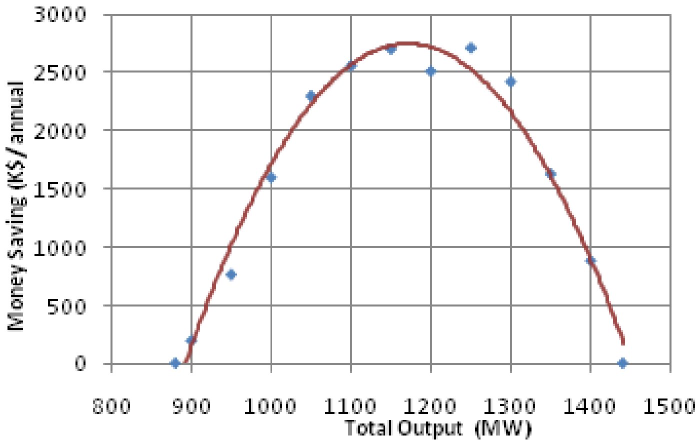

Figure 4. Annual money saving estimation from the optimisation (Calorific value = 26 MJ / kg, fuel price = $30 /per ton)

The curve in Figure 4 indicates that most benefits from load optimisation are made around 1100MW-1300MW in excess of annual fuel saving of two million dollars while no gain is obtained on minimum and maximum loading conditions, which is logical as no options for loading at both ends. In reality, it is impossible to always operate the plant in such a desirable way, that is, we cannot guarantee all four units keep running for a whole year without stopping. Assume there is a 50% chance of possible loading optimisation, the benefits will be halved and fuel savings will be around one million dollars per year.

In order to evaluate the efficiency of constraint-handling methods, two experiments have been conducted to evaluate the computation time for each individual run based on the Model 1. The first experiment adopted the multi-objective constraint-handling method as proposed in this paper. The second experiment adopted the preserving feasibility constraint-handling method as used in our previous work [16]. PSO parameters for both approaches are the same. The 40 particles, 10000 maximum iterations have been used for both experiments. Based on ten independent runs, the minimum time, maximum time and the average time spent for $M_{total} = 1000$ MW are listed in Table 5.

Table 5. Computation time spent for two constraint-handling approaches (Based on 10 independent runs)

CPU time spent	Method in this paper (ms)	Method in previous work [16] (ms)
Minimum	31	3016
Maximum	156	4204
Average	68.9	3925.3

Table 5 demonstrates that the multi-objective based constraint-handling method is much faster than the preserving feasibility method with PSO. The main reason is that the preserving feasibility approach assumes all particles starting at the feasible space which require a long initialization process. In other words, the evolution will not start until all particles are in the feasible space. It may be impractically too long or impossible for the problems that have large search spaces and with small feasible spaces. The multi-objective constraint-handling approach, however, does not require the particles to be in the feasible space at the beginning. The initialization does not need to check if the particles satisfy all constraints which make the initialization easier and faster.

Which model is to be used in the real world? It depends on the environment regulations. If there is an environment licence limit regulated, either Model 1 or Model 2 will suffice. If there is no environment license regulated, the Model 2 can be applied.

5. Conclusion

Power generation unit efficiency will be of greater practical importance in the coming pollution constrained economy in terms of fuel saving and minimizing environmental harm. Based on the problem description, the two optimisation models - the single objective constrained model (Model 1) and the multi-objective constrained model (Model 2) have been presented. The multi-objective constraint-handling method and the constraint dominance concept have been adopted incorporating with PSO algorithm for the unit loading optimisation application.

A simulation to a four-unit coal-fired local power plant has been conducted. The simulation results reveal the capability, effectiveness and efficiency of applying the proposed approach in the power industry. The simulation results have also demonstrated that the room for loading optimisation is significant.

In order to compare the two constraint-handling methods, two experiments have been conducted to evaluate the computation cost. The experiment results have demonstrated that the multi-objective constraint-handling based approach is more efficient than the preserving feasibility constraint-handling approach in terms of CPU time consumed. Since the multi-objective constraint-handling method has no problem-dependent parameters like those applied in the penalty function based constraint-handling approach, it makes it easier to extend to a wide variety of applications.

The power generation loading optimisation can take two models. If there is an environment licence limit applied in practice, the Model 1 is recommended. Otherwise the Model 2 can be adopted.

6. Appendix

Two metrics that can be used for performance evaluation in multi-objective optimisation are described as following.

Spacing (S) [17] measures how well distributed (spaced) the solutions in the non-dominated set found. The formula is:

$$S = \sqrt{\tfrac{1}{n}\sum_{i=1}^{n}(d_i - \overline{d})^2} \tag{20}$$

where n is the number of solutions in the obtained non-dominated set, $d_i = \min_{k\in n \wedge k\neq i} \sum_{m=1}^{M} | f_m^{\,i} - f_m^{\,k} |$ and $\overline{d}$ is the mean value of the above distance measure $\overline{d} = \sum_{i=1}^{n} d_i / n$; f_m is the m-th objective function value; M is the number of objective functions. When the solutions are near uniformly spaced, the corresponding distance measure will be small. Thus, an algorithm finding a set of non-dominated solutions having a smaller spacing S is better.

Maximum Spread [18] gives a value which represents the maximum extension between the farthest solutions in the non-dominated set found. The formula is:

$$D = \sqrt{\sum_{m=1}^{M} (\max_{i=1}^{n} f_m^{\,i} - \min_{i=1}^{n} f_m^{\,i})^2} \tag{21}$$

A larger D value indicates better performance.

7. References

[1] Queensland Government, ed: http://www.epa.qld.gov.au/, Retrieved on 2011-3-16.
[2] A. Farag, et al., "Economic Load Dispatch Multiobjective Optimization Procedures using Linear Programming Techniques", IEEE Transactions on Power System, vol. 10, No. 2, 1995.
[3] I. N. d. Silva and L. Nepomuceno, "An Efficient Neural Approach to Economic Load Dispatch in Power Systems", In Proceedings of the IEEE PES Summer Meeting, CD-Rom, 2001.
[4] I. N. d. Silva, et al., "An Efficient Hopfield Network to Solve Economic Dispatch Problems with Transmission System Representation", International Journal of Electrical Power & Energy Systems, vol. 26, pp. 733-738, 2004.
[5] R. Danraj and F. Gajendran, "An Efficient Algorithm to Find Optimal Economic Load Dispatch for Plants having Discontinuous Fuel Cost Functions", IE(I) Journal - EL, vol. 85, 2004.
[6] M. Basu, et al., "Economic Emission Load Dispatch with Non-smooth Fuel Cost and Emission Level Functions through an Interactive Fuzzy Satisfying Method and Evolutionary Programming Technique", IE(I) Journal - EL, vol. 87, 2006.
[7] B. Zhao and Y.-j. CAO, "Multiple Objective Particle Swarm Optimization Technique for Economic Load Dispatch" , Journal of Zhejiang University SCIENCE, pp. 420-427, 2005.
[8] M. Basu, "Dynamic Economic Emission Dispatch Using Evolutionary Programming and Fuzzy Satisfying Method", International Journal of Emerging Electric Power Systems, vol. 8 (2007), Issue 4, 2007.
[9] R. Eberhart, et al., Computational Intelligence PC Tools. San Diego: Academic Press Professional, 1996.
[10] J. Kennedy and R. Eberhart, "Particle Swarm Optimization", In Proceedings of IEEE International Conference on Neural Networks, Perth, Australia, 1995.
[11] E. Camponogara and S. N. Talukdar, "A Genetic Algorithm For Constrained and Multiobjective optimization" ,In Proceeding of 3rd Nordic Workshop on Genetic Algorithms and Their Applications, Vaasa, Finland, 1995.
[12] E. Mezura-Montes and C. A. C. Coello, "Multiobjective-Based Concepts to Handle Constraints in Evolutionary Algorithms", In Proceedings of the Fourth Mexican International Conference on Computer Science, Apizaco, MEXICO, 2003.
[13] L. D. Li, et al., "A Modified PSO Algorithm for Constrained Multi-objective Optimization" , In Proceedings of 3rd IEEE International Conference on Network and System Security, Gold Coast, Australia, 2009.
[14] L. D. LI, et al., "A Dynamic Neighbourhood Particle Swarm Optimisation Algorithm for Comstrained Optimisation", In Proceddings of the 37th Annual Conference of the IEEE Industial Electronics Society, Melbourne, 2011.
[15] K. Deb, Multi-Objective Optimization using Evolutionary Algorithms: John Wiley & Sons, 2001.
[16] L. D. Li, et al., "Constrained Power Plants Unit Loading Optimization using Particle Swarm Optimization Algorithm", WSEAS Transactions on Information Science and Applications, vol. 4, pp. 296-302, 2007.
[17] J. R. Schott, Fault Tolerant Design Using Single and Multi-Criteria Genetic Algorithms vol. Masters Thesis: Massachusetts Institute of Technology, 1995.
[18] E. Zitzler and L. Thiele, "Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach", IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, vol. 3 (4), pp. 257-271, 1999.