J Glob Optim (2008) 41:479–515
DOI 10.1007/s10898-007-9261-y

Scope of stationary multi-objective evolutionary optimization: a case study on a hydro-thermal power dispatch problem

Kalyanmoy Deb

Received: 15 November 2006 / Accepted: 19 October 2007 / Published online: 5 December 2007

Abstract Many engineering design and developmental activities finally resort to an optimization task which must be solved to get an efficient and often an intelligent solution. Due to various complexities involved with objective functions, constraints, and decision variables, optimization problems are often not adequately suitable to be solved using classical point-by-point methodologies. Evolutionary optimization procedures use a population of solutions and stochastic update operators in an iteration in a manner so as to constitute a flexible search procedure thereby demonstrating promise to such difficult and practical problem-solving tasks. In this paper, we illustrate the power of evolutionary optimization algorithms in handling different kinds of optimization tasks on a hydro-thermal power dispatch optimization problem: (i) dealing with non-linear, non-differentiable objectives and constraints, (ii) dealing with more than one objectives and constraints, (iii) dealing with uncertainties in decision variables and other problem parameters, and (iv) dealing with a large number (more than 1,000) variables. The results on the static power dispatch optimization problem are compared with that reported in an existing simulated annealing based optimization procedure on a 24-variable version of the problem and new solutions are found to *dominate* the solutions of the existing study. Importantly, solutions found by our approach are found to satisfy theoretical Kuhn–Tucker optimality conditions by using the subdifferentials to handle non-differentiable objectives. This systematic and detail study demonstrates that evolutionary optimization procedures are not only flexible and scalable to large-scale optimization problems, but are also potentially efficient in finding theoretical optimal solutions for difficult real-world optimization problems.

Kalyanmoy Deb, Deva Raj Chair Professor.

Currently a Finland Distinguished Professor, Department of Business Technology, Helsinki School of Economics, 00101 Helsinki, Finland.

K. Deb (✉)
Department of Mechanical Engineering, Indian Institute of Technology Kanpur, Kanpur 208016, India
e-mail: deb@iitk.ac.in
URL: http://www.iitk.ac.in/kangal/deb.htm

Keywords Multi-objective optimization · Kuhn–Tucker conditions · Evolutionary optimization · Robust optimization · Large-scale optimization

1 Introduction

Most real-world engineering optimization tasks for design, analysis and developmental purposes involve a variety of complexities:

- Objectives and constraints can be non-linear, non-differentiable and discrete.
- Objectives and constraints can be non-stationary.
- Objectives and constraints can be sensitive to parameter uncertainties near the optimum.
- Number of objectives, constraints, and variables can be large.
- Objectives and constraints can be expensive to compute.
- Decision or design variables can be of mixed type involving continuous, discrete, Boolean, and permutations.

Although classical optimization algorithms involving derivatives and/or deterministic transition rules for moving from one iteration to another are around for more than five decades [15,32,34], they sometimes get into difficulties in handling most of the above problems. When faced with a particular difficulty, an existing algorithm is modified to suit it to apply to the problem. One of the main reasons for this difficulty is that most of the classical methodologies use a point-by-point approach and are designed towards solving a particular structure of an optimization problem. With one point to search a complex search space, algorithms seem to run out of ways (degrees-of-freedom) to improve from its current solution and when faced with solving different problems one algorithm often does not work in many problems.

Evolutionary algorithms (EAs), suggested around 1960s [24,25] and applied to engineering design optimization problems from around 1980s [16,17,20–22], are population based procedures which are increasingly being found to be more suited to different vagaries of practical problems. For more than a decade now, the population-based approach of EAs are exploited to develop evolutionary multi-objective optimization (EMO) procedures and apply them to find a set of near Pareto-optimal solution simultaneously, instead of a single optimal solution [6,14,26,37,40]. In this paper, we consider a hydro-thermal power dispatch problem of generating and meeting power demand using two types of power generation modes: hydroelectric and thermal. This problem is particularly interesting for our study because of the following properties: (i) there are two conflicting and non-linear objective functions—the cost of production arising out of the interest of the power companies and NOx emission arising out of the interest of societal welfare (thereby making the conflict in two objectives), (ii) one of the objective functions is additionally non-differentiable, (iii) there are many quadratic equality constraints involving different sets of variables, thereby making the problem difficult to break into separable programming problems, (iv) every variable is restricted within a lower and an upper bound, thereby introducing two additional constraints for each variable, (v) the feasible search space is small with respect to the entire search space, (vi) the optimization problem is scalable to as many variables (multiples of six) as desired preserving the order of optimal objective values and solutions, and (vii) some parts of the Pareto-optimal frontier are sensitive to parameter uncertainties. In some sense, this problem represents most issues which other practical optimization problems usually possess. By considering one aspect at a time, we illustrate how a population-based EMO procedure can handle different complexities and produce useful solutions for practice. Problems having as many as 1,008 variables, 100s of equality constraints, and 1,000s of variable bounds (which must be treated as inequality

constraints using classical methods), requiring as large as about 6 h of computational time on a PC having a Pentium V processor, are handled in this paper.

Although handling practicalities in optimization problems is the main focus of this study, we also make an effort in verifying EMO-optimized solutions using theoretical Karush–Kuhn–Tucker conditions for their candidacy of an optimal or a near-optimal solution. To handle non-differentiable nature of one of the objective functions for the theoretical study, we use subdifferential concept suggested in the classical optimization literature [3,4] and demonstrate how a systematic verification of near-optimality property of EA-optimized solutions can be achieved even for non-differentiable multi-objective optimization problems.

In the remainder of this paper, we first introduce the hydro-thermal power dispatch optimization problem and discuss different difficulties offered by this problem to any optimization procedure. Thereafter, we describe the EMO solution procedures adopted in this study and present simulation results systematically and scientifically. The problem difficulties and their solution procedures adopted in this study clearly show power and flexibility of evolutionary search procedures. The systematic procedures followed in this paper should motivate researchers and applicationists to adopt a similar study for other practical problem-solving tasks.

2 Hydro-thermal power scheduling problem

In a hydro-thermal power generation system, both the hydroelectric and thermal generating units are utilized to meet the total power demand. A proper scheduling of the power units is an important task in a power system design. The optimum power scheduling problem involves the allocation of power to all concerned units, so that the total fuel cost of thermal generation is minimized, while satisfying all constraints in the hydraulic and power system networks [44]. To solve such a single-objective hydro-thermal scheduling problem, many different conventional techniques, such as Newton's method [46], Lagrange multiplier method [33], dynamic programming [45] and soft computing methodologies such as genetic algorithms [31], evolutionary programming [39], simulated annealing [43] etc., have been tried. However, the thermal power generation process produces harmful emission which must also be minimized for the environmental consideration. Unfortunately, an optimal economic power generation is not optimal for its emission properties and vice versa. Due to the conflicting nature of minimizing power generation cost (of interest to power companies) and emission characteristics (often imposed by governmental bodies for environmental safety), a multi-objective evolutionary algorithm seems to be the most suitable method for this problem [1]. Such an optimization task may find a of Pareto-optimal solutions with different trade-off conditions between two different objectives in a single simulation run, instead of artificially combining both objective somehow to suit an optimization procedure. Although the problem is of dynamic in nature, involving a time varying power demand term in the formulation, in this paper, we treat the problem as a static optimization problem with a known demand pattern over time. We are working on a sequel of this paper in which the demand is treated as a dynamic parameter which results in a dynamic multi-objective optimization involving on-line optimization and decision-making activities.

2.1 Optimization problem formulation

The original formulation of the problem was given in Basu [1], The hydro-thermal power generation system is optimized for a total scheduling period of T. However, the system is

assumed to remain fixed for a period of t_m so that there are a total of $M = T/t_m$ changes in the problem during the total scheduling period. In this off-line optimization problem, we assume that the demand in all M time intervals are known a priori and an optimization needs to be made to find the overall schedule before starting the operation. Let us also assume that the system consists of N_s number of thermal ($\mathbf{P}^{\mathrm{s}}$) and N_h number of hydroelectric ($\mathbf{P}^{\mathrm{h}}$) generating units sharing the total power demand. Thus, there are a total of $n = M(N_h + N_s)$ decision variables to the underlying optimization problem.

The previous studies [1] considered two objectives for scheduling such a power dispatch problem—(i) minimize the fuel cost for producing the thermal power and (ii) minimize the NOx emission from thermal power units. The first objective is of interest to power companies producing the power and the second objective is often desired for societal welfare and is often imposed by governmental bodies. It is usually the case that both these objectives are conflicting in nature. The fuel cost function of each thermal unit considering valve-point effects is expressed as the sum of a quadratic and a sinusoidal function and the total fuel cost in terms of real power output for the whole scheduling period can be expressed as follows [1]:

$$f_1(\mathbf{P}^{\mathrm{h}}, \mathbf{P}^{\mathrm{s}}) = \sum_{m=1}^{M} \sum_{s=1}^{N_s} t_m \left(a_s + b_s P^s_{\mathrm{sm}} + c_s (P^{\mathrm{s}}_{sm})^2 + \left|d_s \sin\left(e_s (P^{\mathrm{s}}_{s,\min} - P^{\mathrm{s}}_{sm})\right)\right|\right), \quad (1)$$

where a_s, b_s, c_s, d_s and e_s are parameters associated with s-th power generation unit and are described in the Appendix. Notice that the fuel cost is involved only with thermal power generation units and hydroelectric units do not contribute in the cost objective. This objective is non-differentiable due to the absolute function used in the right-most term.

The second objective—emission of total amount of nitrogen oxides—for the whole scheduling period T from all the thermal generating units can be obtained by adding up the individual emissions, expressed as follows:

$$f_2(\mathbf{P}^{\mathrm{h}}, \mathbf{P}^{\mathrm{s}}) = \sum_{m=1}^{M} \sum_{s=1}^{N_s} t_m \left(\alpha_s + \beta_s P^{\mathrm{s}}_{sm} + \gamma_s (P^{\mathrm{s}}_{sm})^2 + \eta_s \exp(\delta_s P^{\mathrm{s}}_{sm})\right), \quad (2)$$

where $\alpha_s, \beta_s, \gamma_s, \eta_s$ and δ_s are parameters associated with the s-th thermal power generation unit and also described in the Appendix. Like the cost objective, the emission objective does not involve hydroelectric power generation units. However, the effect of hydroelectric generation units to the optimization problem comes from the associated constraints, which we describe below.

The optimization problem has several constraints involving power balance for both thermal and hydroelectric units and water availability for hydroelectric units. In the power balance constraint, the demand term is time dependent, which we assume to be known in this static optimization study:

$$\left(\sum_{s=1}^{N_s} P^{\mathrm{s}}_{sm}\right) + \left(\sum_{h=1}^{N_h} P^{\mathrm{h}}_{hm}\right) - P_{Dm} - P_{Lm} = 0, \quad m = 1, 2 \ldots, M, \quad (3)$$

where the transmission loss P_{Lm} term at the m-th time period is given as follows:

$$P_{Lm} = \sum_{i=1}^{N_h+N_s} \sum_{j=1}^{N_h+N_s} \left(P_{im} B_{ij} P_{jm}\right). \quad (4)$$

Here, both types of power generation units are written in a combined form with $\mathbf{P}_m = (\mathbf{P}^{\mathrm{h}}_m, \mathbf{P}^{\mathrm{s}}_m)$. This constraint involves both thermal and hydroelectric power generation units.

The water availability constraint can be written as follows:

$$\left[\sum_{m=1}^{M} t_m \left(a_{0h} + a_{1h} P^{\mathrm{h}}_{hm} + a_{2h}(P^{\mathrm{h}}_{hm})^2\right)\right] - W_h = 0, \quad h = 1, 2, \ldots, N_h. \tag{5}$$

The variable bounds are given as follows:

$$P^{\mathrm{h}}_{h,\min} \le P^{\mathrm{h}}_{hm} \le P^{\mathrm{h}}_{h,\max}, \quad h = 1, 2, \ldots, N_h, m = 1, 2, \ldots, M, \tag{6}$$

$$P^{\mathrm{s}}_{s,\min} \le P^{\mathrm{s}}_{sm} \le P^{\mathrm{s}}_{s,\max}, \quad s = 1, 2, \ldots, N_s, m = 1, 2, \ldots, M. \tag{7}$$

The non-linear programming formulation of the complete problem is given below for the variable set $\mathbf{x} = (\mathbf{P}^{\mathrm{h}}, \mathbf{P}^{\mathrm{s}})$:

$$\begin{aligned}
&\text{Minimize } f_1(\mathbf{x}) = \sum_{m=1}^{M}\sum_{s=1}^{N_s} t_m \left(a_s + b_s P^{\mathrm{s}}_{sm} + c_s (P^{\mathrm{s}}_{sm})^2 + |d_s \sin(e_s (P^{\mathrm{s}}_{s,\min} - P^{\mathrm{s}}_{sm}))|\right),\\
&\text{Minimize } f_2(\mathbf{x}) = \sum_{m=1}^{M}\sum_{s=1}^{N_s} t_m \left(\alpha_s + \beta_s P^{\mathrm{s}}_{sm} + \gamma_s (P^{\mathrm{s}}_{sm})^2 + \eta_s \exp(\delta_s P^{\mathrm{s}}_{sm})\right),\\
&\text{subject to } \left(\sum_{s=1}^{N_s} P^{\mathrm{s}}_{sm}\right) + \left(\sum_{h=1}^{N_h} P^{\mathrm{h}}_{hm}\right) - P_{Dm} - P_{Lm} = 0, \quad m = 1, 2 \ldots, M,\\
&\quad \left[\sum_{m=1}^{M} t_m \left(a_{0h} + a_{1h} P^{\mathrm{h}}_{hm} + a_{2h}(P^{\mathrm{h}}_{hm})^2\right)\right] - W_h = 0, \quad h = 1, 2, \ldots, N_h.\\
&\quad P^{\mathrm{h}}_{h,\min} \le P_{hm} \le P^{\mathrm{h}}_{h,\max}, \quad h = 1, 2, \ldots, N_h, m = 1, 2, \ldots, M,\\
&\quad P^{\mathrm{s}}_{s,\min} \le P_{sm} \le P^{\mathrm{s}}_{s,\max}, \quad s = 1, 2, \ldots, N_s, m = 1, 2, \ldots, M.
\end{aligned} \tag{8}$$

The above two-objective optimization problem involves $(M(N_s + N_h))$ variables, two objectives, $(M + N_h)$ quadratic equality constraints and $(2M(N_s + N_h))$ variable bounds. The above problem has a number of complexities which may provide a stiff challenge to any optimization algorithm to find optimal solutions:

- Both objectives are non-linear to the decision variables and are conflicting to each other, thus generating a set of Pareto-optimal solutions, instead of a single optimal solution, to be the target of an optimization task.
- Constraints are all of quadratic equality type, thereby indicating that feasible solutions must lie on the intersection of all constraint surfaces.
- The first objective function is non-differentiable to the decision variables, thereby making it difficult for gradient-based optimization algorithms to be used to find the optimal solutions in this problem.
- The problem is scalable to an arbitrary number of decision variables by increasing the number of time periods (M), thereby allowing to perform a systematic study of the scalability property of an algorithm.

The specific stationary case considered here is involved with two hydroelectric ($N_h = 2$) and four thermal ($N_s = 4$)) power generation units. Thus, each time period contains six variables. For M time periods, there are a total of $n = 6M$ variables in this problem. An overall time span of $T = 48$ h is considered. Thus, for M time periods, each time period extends for $t_m = 48/M$ h. To start with, we consider only four time periods ($M = 4$). However, later we consider up to 168 time periods, thereby causing the underlying problem to have as large as $n = 6 \times 168$ or 1,008 variables. Other parameter values related to the problem are given in the Appendix and remain fixed for each power generation unit over time. For the $M = 4$ case, power demand values of 900, 1,100, 1,000 and 1,300 MW are

considered for the four time periods, respectively. For larger time period problems, we use a piece-wise change in demand values, as discussed later.

3 Proposed solution methodology using an EA

An EA procedure for solving the above optimization problem helps to reduce the complexity of the problem somewhat. We discuss these complexity reduction techniques in the following subsections.

3.1 Handling variables and variable bounds

The current problem involves $N_h = 2$ hydroelectric and $N_s = 4$ thermal power units. The variables are coded as a vector of real numbers in a subset of $(\mathbf{P}^{\mathrm{h}}, \mathbf{P}^{\mathrm{s}})$ for each time period (1 till M), as follows:

$$((\underbrace{P^{\mathrm{h}}_{11}\ P^{\mathrm{h}}_{21}}_{\mathcal{H}_1}\ \underbrace{P^{\mathrm{s}}_{11}P^{\mathrm{s}}_{21}P^{\mathrm{s}}_{31}P^{\mathrm{s}}_{41}}_{\mathcal{T}_1})(\underbrace{P^{\mathrm{h}}_{12}\ P^{\mathrm{h}}_{22}}_{\mathcal{H}_2}\ \underbrace{P^{\mathrm{s}}_{12}P^{\mathrm{s}}_{22}P^{\mathrm{s}}_{32}P^{\mathrm{s}}_{42}}_{\mathcal{T}_2})\ldots (\underbrace{P^{\mathrm{h}}_{1M}\ P^{\mathrm{h}}_{2M}}_{\mathcal{H}_M}\ \underbrace{P^{\mathrm{s}}_{1M}P^{\mathrm{s}}_{2M}P^{\mathrm{s}}_{3M}P^{\mathrm{s}}_{4M}}_{\mathcal{T}_M})).$$

The current optimization problem with $M = 4$ involves ($12M$ or 48) variable bounds. For many classical optimization method, these variable bounds must have to treated as constraints. However, in a real-parameter EA, they all can be easily taken care directly in the initialization procedure by making sure that solutions are always initialized within the specified bounds. Variable bounds are also respected while generating new solutions by means of recombination and mutation operators so that no solution outside the specified bounds are created [6,9]. The main difficulty arises in handling equality constraints, which we discuss in the following subsection.

3.2 Handling quadratic equality constraints

The optimization problem formulation given in equation 8 states that there are two sets of equality constraints—one that relates to only hydroelectric power generation variables and the other relating to all variables. It is also interesting to note that both constraint sets involve polynomial expressions of second order, thereby allowing us to use a *repair* mechanism to handle the equality constraints directly. Since EAs allow procedure-based techniques to handle constraints, objectives and decision variables, repair mechanisms are often employed to handle constraints in EAs [27,28]. In contrast, a previous study [1] used a penalty approach to handle the equality constraints. In this approach, if a solution did not satisfy any of the equality constraints, the solution is declared infeasible and is penalized. A large number of non-linear equality constraints can result in a tiny proportion of the feasible search space (and also often of non-convex nature) compared to the overall space for search by an optimization algorithm. In such problems, it is unlikely that a randomly created solution will satisfy all constraints and becomes feasible. But if a solution can be repaired using the constraints in a computationally viable manner, not only does the solution becomes feasible, the number of effective decision variables reduces, thereby reducing the complexity of solving the optimization problem.

Among the two sets of constraints we first handle the second set which deals with the water availability constraints (Eq. 5). We consider them first mainly due to their involvement of hydroelectric variables alone. Each equality constraint involves M different decision variables arising from a particular hydroelectric power generation unit ($P^{\mathrm{h}}_{h\mu}$) for M time periods. Although an EA population member prescribes all M values, when they are substituted to the equality constraint, they need not satisfy the constraint. To satisfy the equality constraint, we keep the $M-1$ variables (P^{h}_{hm}, $m=1,2,\ldots,M$ and $m\neq\mu$) fixed to their prescribed values and replace the μ-th variable $P^{\mathrm{h}}_{h\mu}$ by a suitable root of the quadratic equality constraint. To maintain the structure of the EA population member, we preserve their original ratios (marked as EA) with respect to the P^{h}_{hm} variables as follows: $P^{\mathrm{h}}_{hm} \leftarrow (P^{\mathrm{h}}_{hm}/P^{\mathrm{h}}_{h\mu})_{\mathrm{EA}} P^{\mathrm{h}}_{h\bar{\mu}}$, where $P^{\mathrm{h}}_{h\bar{\mu}}$ is the root of the following quadratic equation:

$$\left(a_{2h}\sum_{m=1}^{M} t_m \left(\frac{P^{\mathrm{h}}_{hm}}{P^{\mathrm{h}}_{h\mu}}\right)_{\mathrm{EA}}\right)\left(P^{\mathrm{h}}_{h\bar{\mu}}\right)^2 + \left(a_{1h}\sum_{m=1}^{M} t_m \left(\frac{P^{\mathrm{h}}_{hm}}{P^{\mathrm{h}}_{h\mu}}\right)_{\mathrm{EA}}\right) P^{\mathrm{h}}_{h\bar{\mu}} + (a_{0h}T - W_h) = 0. \tag{9}$$

Since $\left(\frac{a_{1h}}{a_{2h}}\right)$ is always positive, both roots of this equation cannot be positive. Thus, in two of the three cases, we choose the corresponding positive root and update all M variables. After the update, we check if all modified variables satisfy their specified variable bounds. If yes, we accept this modification. If not or in the third case (where both roots are negative), we declare a different variable as $P^{\mathrm{h}}_{h\mu}$, recompute the root, and follow the above checking procedure. If after M such iterations, a feasible set is still not found, we declare the EA population member to be infeasible.

After a successful repair for one hydroelectric power unit for M time periods, we consider the next unit with the corresponding equality constraint and continue to execute above procedure till all hydroelectric units are considered. If all hydroelectric units are possible to be modified to satisfy N_h equality constraints, the power balance equality constraints (the first constraint set in Eq. 8) are used to repair the thermal power generation variables. We describe this repair procedure a little later. However, if the above repair mechanism for P^h_{hm} variables is not successful, the solution is declared as infeasible and no further computation of power balance constraints nor the computation of objective functions are performed. Instead, all original variables (hydroelectric and thermal) are substituted in the constraint equations and a normalized constraint violation value is computed:

$$\begin{aligned}\mathrm{CV} = &\sum_{m=1}^{M}\left|\left[\left(\sum_{s=1}^{N_s} P^{\mathrm{s}}_{sm}\right)+\left(\sum_{h=1}^{N_h} P^{\mathrm{h}}_{hm}\right)-P_{Lm}\right]/P_{Dm}-1\right| \\ &+\sum_{h=1}^{N_h}\left|\left[\sum_{m=1}^{M} t_m\left(a_{0h}+a_{1h}P^{\mathrm{h}}_{hm}+a_{2h}(P^{\mathrm{h}}_{hm})^2\right)\right]/W_h-1\right|. \end{aligned} \tag{10}$$

Thereafter, the following penalty-parameter-less procedure [5] is used to discourage such solutions to be propagated to the next generation of the algorithm. In the *tournament* selection operation involving two solutions, following three considerations are made [6]: (i) if one is feasible and other is not, we simply choose the feasible one, (ii) if both are feasible, the one having a smaller function value or having a better non-domination level is chosen, and (iii) if both are infeasible, the one with smaller normalized constraint violation (CV) is chosen. Thus, it is interesting to note that if a solution is infeasible, objective function computation is not required to be performed with the above procedure, thereby not requiring to specify

any user-defined penalty parameter. Since this constraint handling procedure can only be applied with a procedure dealing with more than one solution in an iteration, such a penalty-parameter-less strategy is an ideal procedure for a population-based optimization procedure, such as for an evolutionary optimization method.

Equation 3 involves M quadratic equality constraints, each for one time period. We follow a similar procedure as above and repair a particular thermal unit $P_{\psi m}$ of four thermal units for each time slot. The quadratic equation for this variable can be written as follows:

$$B_{\psi\psi} P_{\psi m}^2 + \left(2 \sum_{j=1}^{N_h+N_s-1} B_{\psi j} P_{jm} - 1 \right) P_{\psi m} + \left(P_{Dm} + \sum_{i=1}^{N_h+N_s-1} \sum_{j=1}^{N_h+N_s-1} P_{im} B_{ij} P_{jm} - \sum_{i=1}^{N_h+N_s-1} B_{\psi i} P_{im} \right) = 0. \quad (11)$$

Since the required hydroelectric power units (P_{hm}^{h}) are already available when considering these constraints, the above equation can be solved for $P_{\psi m}$. If the particular $P_{\psi m}$ value computed by finding the root of the above equation comes within the prescribed variable bounds, then the variable is accepted and we go for next constraint equation. Otherwise, another quadratic equation is formed with the next thermal unit and checked for its root to lie within its bounds. If for a time period, none of the N_s thermal units resulted in a successful replacement, the constraint violation is computed using the original thermal units and the repaired hydroelectric variables, and the solution is declared infeasible. Once again, the previously-discussed penalty-parameter-less constraint handling method is used for this purpose.

3.2.1 Time complexity estimate

The main computation involves in finding roots of above quadratic equations. First, we consider the water balance equations. There are N_h constraints, each involving all M hydroelectric power generation values for a particular unit. If P_{hm}^{h} (for $m = 1, 2, \ldots, M$) combinations are such that the solution of only one quadratic equation finds all M power generation values within the specified bounds for each unit, only N_h root-finding equations are required to be solved for each hydroelectric unit. However, the worst case scenario happens when all M power generation values for a fixed h (P_{hm}^{h}) are to be tried one after another to find a feasible solution. This will require a total of MN_h root-finding computations. Similarly, for finding a feasible set of thermal power units, M quadratic equations are to be solved, each requiring a maximum (worst case) of N_s sequential consideration of quadratic equations involving P_{sm}^{s} terms. Thus, in the worst case, there is a need of MN_s root-finding efforts. Thus, for each solution evaluation, in the worst case, a total of $M(N_s + N_h)$ root-finding computations are needed. For an EA with N population members and running for G generations, a total of $T = GNM(N_s + N_h)$ root-finding computations are necessary in the worst case. In the EA literature, studies exist in which different population sizing relationships (linear, square-root, and others) and different number of generations with number of decision variables were used [19,23,30]. However, it is a common practice to allocate an increasing number of overall function evaluations for an increase in number of decision variables. In this study, we choose a population size linearly proportional to the number of decision variables and the EA is run for a fixed number of generations, so that the number of function evaluations increase with the number of decision variables. However, for the current problem, the computational time of evaluating a solution also increases with the

number of decision variables, thereby increasing the overall time complexity estimate to be more than linear to the increase in decision variables. In the following paragraph, we account for the time complexity estimate of evaluating a solution.

In our problem, since the number of decision variables is $n = (N_h + N_s)M$ for a fixed number of hydroelectric and thermal power generations, the worst case time complexity is expected to be $O(M^2)$ or $O(n^2)$. However, with an increase in time periods (M), it is expected to be an increasingly difficult task to find a feasible solution with respect to power balance constraints. If a feasible root is not found in solving any one of the equality constraints, this solution will be declared infeasible. In an extreme situation, if in all cases, a feasible solution cannot be found for the very first constraint equation, the solution will be declared infeasible right way and no further root-finding efforts will be spent. In such a scenario, the overall time complexity will be $O(M)$ or $O(n)$. Thus, in practice, we would expect the time complexity of the above constraint-handling procedure for a fixed number of generations to vary between $O(n)$ and $O(n^2)$, thereby indicating a polynomial time computational procedure with an increase in number of variables.

3.3 Handling multiple objectives

Multi-objective optimization problems involving conflicting objectives give rise to a set of trade-off optimal solutions [6]. Each such solution is a potential candidate for implementation, but exhibits a trade-off between two objectives. Classical methods require a preference information a priori and then optimize a preferred single-objective version of the problem [29]. If different trade-off solutions are needed to investigate the effect of different preference values before choosing a final solution, such a classical approach requires to be applied again and again. A recent study has shown than such independent optimizations may be computationally expensive in complex problems [38].

However, EAs are ideal for such problem solving tasks. This is because the EA population can be used to store different trade-off optimal solutions obtained in a single simulation run. A number of efficient methodologies exist for this purpose [6], here we use a commonly-used procedure (NSGA-II [8]), which uses a non-domination sorting of population members to emphasize non-dominated solutions systematically, an elite-preserving procedure for faster convergence, and a diversity-preserving mechanism for maintaining a widely distributed set of solutions in the objective space. More about the NSGA-II procedure can be found in the original study [8]. A code in C programming language is freely available from http://www.iitk.ac.in/kangal/soft.htm.

4 Simulation results

The real-parameter NSGA-II is combined with the above-discussed constraint handling method for solving the hydro-thermal scheduling problem for $M = 4$, involving 24 real-valued variables. NSGA-II parameters used in this study are as follows: Population size is 100, SBX recombination probability is 0.9, polynomial mutation probability is $1/n$ (where n is the number of variables), and distribution indices for recombination and mutation are 10 and 20, respectively. More about these operators can be found in [6,7]. Figure 1 shows the obtained non-dominated front obtained using NSGA-II using small circles. Table 1 shows a few trade-off solutions obtained using the NSGA-II procedure.

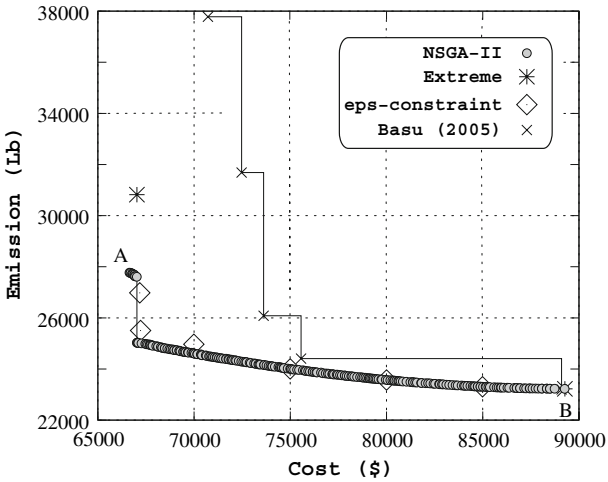

Fig. 1 Non-dominated front obtained by NSGA-II, verified by single-objective methods, and by a previous study

To gain confidence on the obtained NSGA-II solutions, we perform a number of single-objective optimization studies on the same problem. First, each objective is optimized independently by a real-parameter EA (with identical tournament selection, crossover and mutation operators, constraint-handling strategy, and EA parameter values as those used in the above NSGA-II study). Two obtained extreme solutions are also shown in Fig. 1 with a star. One of the extreme points (minimum-cost solution) is found to be *dominated* by a NSGA-II solution (marked as 'A'), and the minimum-emission solution is matched by the extreme NSGA-II solution (marked as 'B'). Due to lack of adequate diversity present in a single-objective EA, in some difficult problems, EAs may prematurely converge to a sub-optimal solution. However, the maintenance of multiple trade-off solutions in a multi-objective EA helps maintain adequate diversity in the EA population, thereby reducing the chance of such a premature convergence. We observe this phenomenon to occur here for finding the minimum-cost solution, because the cost objective is non-differentiable, non-linear, and periodic, thereby causing adequate difficulty to the single-objective optimization procedure used in this study.

Second, to validate the optimality of some other intermediate NSGA-II solutions, we employ the same single-objective EA to solve several ϵ-constraint problems [29] by fixing function f_1 at different ϵ values:

$$\begin{array}{ll} \text{Minimize} & f_2(\mathbf{P}^{\mathrm{h}}, \mathbf{P}^{\mathrm{s}}), \\ \text{Subject to} & f_1(\mathbf{P}^{\mathrm{h}}, \mathbf{P}^{\mathrm{s}}) \leq \epsilon, \\ & (\mathbf{P}^{\mathrm{h}}, \mathbf{P}^{\mathrm{s}}) \in \mathcal{S}, \end{array} \tag{12}$$

where $\mathcal{S}$ is the feasible search space satisfying all constraints and variable bounds, presented in Eq. 8. The obtained solutions are shown in Fig. 1 with a diamond and it is observed that all these solutions (except a few near the minimum-cost solution, about which we discuss more in Sect. 4.3) match well with those obtained by NSGA-II. It is unlikely that so many different independent (single and multi-objective) optimizations will result in one particular trade-off frontier, unless the obtained frontier itself is close to the true optimal frontier. Although we perform a theoretical optimality check in Sect. 4.2, these multiple optimization procedures

Table 1 NSGA-II solutions for the original problem

Sol.No.	Time Pd.	P_1^h(MW)	P_2^h(MW)	P_1^s (MW)	P_2^s(MW)	P_3^s(MW)	P_4^s(MW)	Objectives f_1 (\$), f_2 (Lb)
1	1	168.70	314.62	72.98	134.30	135.92	91.64	$f_1 = 89257.42$
	2	246.07	409.03	80.33	143.16	148.12	101.52	$f_2 = 23223.18$
	3	210.98	357.24	77.03	139.22	142.25	96.13	
	4	*250.00*	*500.00*	104.10	169.04	185.58	130.84	
2	1	173.74	317.69	78.96	130.23	125.39	92.18	$f_1 = 85012.93$
	2	243.53	406.20	87.85	142.94	140.91	106.64	$f_2 = 23304.37$
	3	209.05	357.35	85.80	137.62	134.45	98.49	
	4	*249.88*	*499.99*	98.34	168.35	187.09	135.92	
3	1	171.77	318.08	85.21	123.42	124.98	94.67	$f_1 = 80056.08$
	2	242.38	406.72	93.82	139.42	125.75	119.85	$f_2 = 23564.85$
	3	212.24	356.42	91.04	129.09	125.41	108.49	
	4	*249.75*	*500.00*	98.49	166.36	187.66	137.30	
4	1	171.90	318.32	92.33	113.83	124.95	96.76	$f_1 = 75031.78$
	2	242.69	408.03	97.50	121.29	125.58	132.80	$f_2 = 24000.21$
	3	211.55	355.17	91.99	117.30	125.14	121.45	
	4	*249.99*	*499.69*	98.12	166.58	187.59	137.58	
5	1	191.62	324.84	97.51	112.87	124.99	66.64	$f_1 = 70039.35$
	2	232.71	420.03	98.49	112.90	125.05	138.83	$f_2 = 24589.13$
	3	203.06	350.87	97.32	113.55	124.91	132.68	
	4	*250.00*	487.05	98.81	166.28	197.45	139.65	
6	1	196.35	336.19	98.55	112.75	124.94	*50.00*	$f_1 = 66641.07$
	2	232.72	419.26	98.56	112.68	124.91	139.85	$f_2 = 27769.00$
	3	214.35	379.58	51.89	112.68	124.91	139.80	
	4	234.88	452.82	98.54	112.64	209.82	229.31	

Italized values are close to their respective upper bounds (refer Sect. 4.1 for a discussion)

give us confidence about the optimality of the obtained NSGA-II frontier. We believe that the absence of an EA's proof of convergence to any arbitrary problem for a finite number of evaluations, such a procedure of finding solutions by an EMO and verifying them with various optimization techniques (and, if possible, the theoretical study we performed in Sect. 4.2) is a reliable approach for practical optimization.

Basu [1] used a simulated annealing (SA) procedure to solve the same problem for four time periods ($M = 4$). That study used a root-finding approach for handling the power balance constraints and a naive penalty function approach for handling water resource constraints, in which if any SA solution if found infeasible, it is simply penalized. Moreover, a weight vector is used to constitute a single objective function from two conflicting objectives. For different weight vectors, the study used a goal programming methodology to find a set of optimized solutions. A comparison of the reported results (shown by 'x') is made with our NSGA-II approach in Fig. 1. It is observed that the front obtained by NSGA-II *dominates* that obtained in the previous SA study. Only the minimum emission solution found by both algorithms are identical. For trade-off solutions closer to the minimum-cost solution, the performance of the existing study is worse. This is mainly due to the fact that the previous study used (i) a naive constraint handling method despite the constraints being quadratic

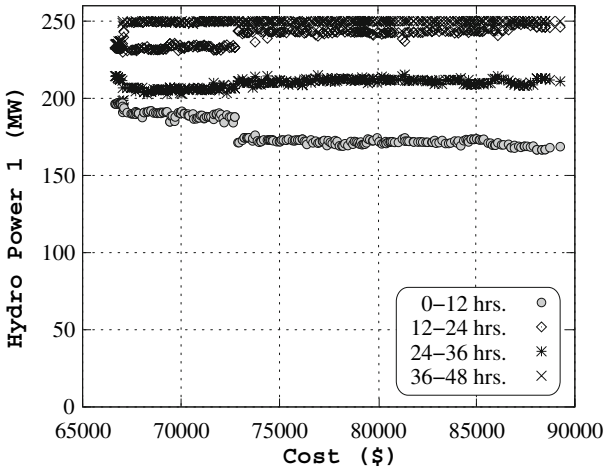

Fig. 2 Hydroelectric unit P_{1m}^{h} versus f_1 for the original problem

and (ii) a point-by-point search strategy. The repair mechanism used in our population-based approach helps find feasible solutions by the use of a mathematical root-finding strategy and in turn enables a better search to reach closer to the true optimal solutions.

4.1 Extending variable boundaries

Figures 2 and 3 show the variation of two hydroelectric power generation values for different trade-off solutions obtained by NSGA-II for $M = 4$ (24-variable) problem. These values are plotted against the corresponding cost objective value. The figure shows that for all trade-off solutions the hydroelectric power generation for the first unit in the fourth time period during 36–48 h (P_{14}^{h}) needs to be set at its upper limit of 250 MW and for most solutions the second hydroelectric unit at the same time period (P_{24}^{h}) needs to be set at its upper limit of 500 MW. Table 1 italicizes the power generation values which are assigned close to their lower or upper bounds in six well-distributed solutions from the entire Pareto-optimal front. These facts suggest that there is a scope of improving cost and emission values by possibly extending the upper bounds of these two variables. In the conceptual stage of design of a power generation system, such information is useful in designing an efficient yet optimal system and interestingly an optimization study is capable of making such vaulable recommendations. Based on this observation, we increase the upper limits of these two variables to 350 and 600 MW, respectively. We have also observed that one of the thermal generators (P_{24}^{s}) almost reaches its upper limit of 175 MW in the fourth time period and hence we also increase its upper limit to 200 MW. All other variable bounds are kept at their earlier values, shown in the Appendix.

Figure 4 shows the obtained front using these extended variable bounds. Individual optima and a number of ϵ-constraint single-objective minimizations on the modified problem suggest that the accuracy of the obtained NSGA-II front. Table 2 shows a few selected solutions from the NSGA-II frontier. We present the corresponding variable values later through Figs. 7–12. Both Table 2 and these figures demonstrate that in all cases the obtained variable values are well within their prescribed variable bounds. These solutions indicate their true optimality and are not affected by the specified variable bounds. An optimization study of this nature can help users in practice to analyze the trade-off between true optimal solutions and limiting

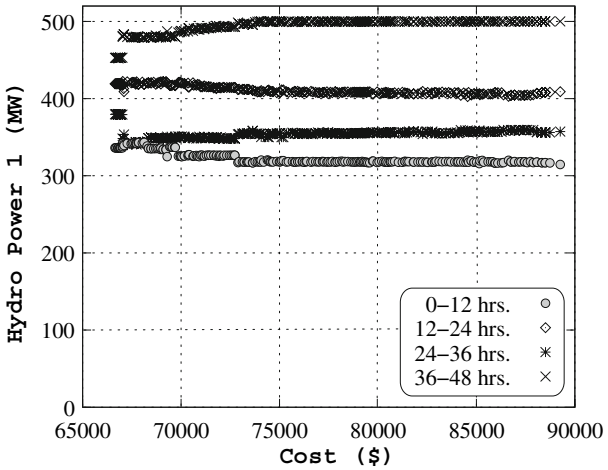

Fig. 3 Hydroelectric unit P_{2m}^{h} versus f_1 for the original problem

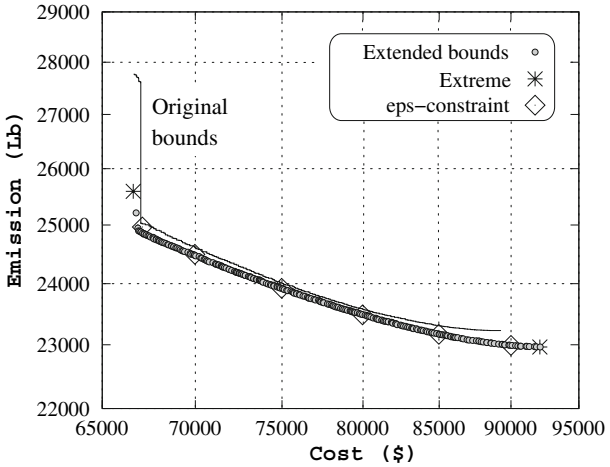

Fig. 4 NSGA-II frontier verified by single-objective optimizations for the extended-boundary problem. Original frontier is also shown in a solid line

bounds on variables. To illustrate this trade-off, the previous NSGA-II frontier (with original, strict bounds) is also shown in a solid line. It is clear that the extension of boundaries allowed a better front to be achieved. The new front dominates the previous front completely. In both objective axes, better individual optimal solutions are obtained, due to the enhancement of variable bounds. Interestingly, solutions near the minimum-cost solution is now much better behaved than in the original case.

4.2 Verifying solutions using theoretical Kuhn–Tucker optimality conditions

One of the criticisms of real-parameter evolutionary algorithms for solving optimization problems has been their lack of a proof of convergence to the true optimal solution in an arbitrary problem with a finite computational complexity. Although the iterative evolutionary

Table 2 NSGA-II solutions for the problem with extended bounds

Sol.No.	Time Pd.	P_1^h(MW)	P_2^h(MW)	P_1^s(MW)	P_2^s(MW)	P_3^s(MW)	P_4^s(MW)	Objectives f_1 (\$), f_2 (Lb)
1	1	151.45	310.40	76.90	139.03	143.01	97.18	$f_1 = 92120.45$
	2	227.59	406.19	84.71	147.93	155.15	106.38	$f_2 = 22963.09$
	3	187.36	360.18	80.66	143.85	148.94	101.69	
	4	302.90	503.43	92.34	157.00	168.09	116.62	
2	1	154.72	313.47	78.76	136.62	137.56	96.87	$f_1 = 89929.82$
	2	227.90	406.52	84.81	147.12	150.84	110.74	$f_2 = 22991.40$
	3	191.52	361.06	83.91	140.63	142.03	103.52	
	4	296.47	499.93	93.60	158.39	169.37	122.41	
3	1	155.53	316.56	85.65	134.61	*124.93*	100.72	$f_1 = 84997.68$
	2	231.57	404.91	94.49	141.20	140.55	115.11	$f_2 = 23169.00$
	3	191.25	363.75	87.59	138.59	133.37	108.13	
	4	292.71	496.62	*97.27*	156.93	166.94	129.50	
4	1	155.26	316.86	91.92	122.33	*124.95*	106.61	$f_1 = 79966.72$
	2	233.33	405.86	*96.56*	138.24	130.73	123.09	$f_2 = 23488.10$
	3	191.96	366.90	94.85	128.14	*125.67*	115.13	
	4	290.70	492.80	*96.41*	157.84	168.67	133.43	
5	1	157.48	312.69	92.77	114.57	*124.91*	115.43	$f_1 = 75023.50$
	2	231.00	407.55	*98.44*	133.62	*125.27*	131.88	$f_2 = 23920.95$
	3	189.80	363.79	94.71	124.51	*124.94*	124.79	
	4	292.89	497.31	*99.82*	166.75	143.27	*139.85*	
6	1	151.99	306.57	*96.98*	112.80	*124.73*	124.60	$f_1 = 70018.99$
	2	240.30	406.02	*98.61*	118.54	*124.85*	139.46	$f_2 = 24467.52$
	3	188.74	361.19	*97.07*	115.10	*125.55*	134.79	
	4	289.67	505.75	*98.56*	171.84	134.65	*139.60*	

Thermal power generation values near *kink* locations of objective f_1 are italized (refer to the final paragraph in Sect. 4.2 for a discussion)

process of updating a population of interacting solutions towards better search regions is an interesting concept, besides on a few specific test problems, evolutionary optimization algorithms (including multi-objective EAs) do not have a convergence proof with a finite time complexity estimate. It is important to note, however, that the so-called no-free-lunch (NFL) theory proved elsewhere [42] prohibits achieving a particular optimization algorithm, including any classical optimization algorithm and EAs, to solve *all* optimization problems efficiently within a finite number of evaluations. However, the greatest discomfort in adopting an EA procedure to a theoretical mind arises from EA's heuristic recombination and mutation operators. The use of gradients or other mathematical concepts such as conjugate directions [13] or feasible directions [47] in an algorithm provides some assurance that the resulting optimization algorithm is guided by a mathematical concept, despite the possibilities of a number of shortcomings of such algorithms such as getting stuck at local optimal solutions, getting cursed due to dimensionality in variables, objectives and constraints, and the need for using a host of heuristic approaches for handling constraints, discrete variables etc. Although

a time complexity estimate is difficult to obtain for any arbitrary optimization problem, an asymptotic convergence to the global optimal solution by specific EAs to any problem is proven elsewhere [36]. If viewed carefully and leaving aside its natural connection for a moment, an EA, through iterations, makes a constant emphasis on maintaining better solutions so that these better solutions can in turn find new better solutions by the act of recombination and blending, and local perturbation of existing solutions. EA researchers, over the years, have understood the importance of each component of such procedures and their interactions to make the complete procedure an algorithm for achieving optimal or near-optimal solutions in complex and difficult problem solving tasks [18].

What EA researchers who are interested in solving real-parameter optimization problems have not done enough is to verify the EA-found solution(s) for their theoretical optimality conditions.[1] However, there is an apparent reason for this act. The first-order Karush–Kuhn–Tucker (KKT) necessary optimality conditions have two bottlenecks to be applied in practice: (i) they are necessary conditions, meaning that a satisfaction of these conditions does not necessarily mean that the solution is an optimal solution to the underlying optimization problem, and (ii) they are mostly applicable to problems having objective and constraint functions which are differentiable. Although the first difficulty can be remedied by considering second-order sufficient conditions [2], they are usually computationally too demanding to apply them in practice. However, if the problem is differentiable, there is no harm in verifying whether an EA-optimized solution satisfies Karush–Kuhn–Tucker necessary conditions [34,35]. Because, if a solution does not satisfy these conditions, we can instantly rule out the solution from being an optimal solution. The second difficulty can also be remedied for certain problems by considering Clarke's subdifferential concept [3,4] for handling functions which are non-differentiable at optimal or other critical points. A recent study proposed an error metric based on the extent of satisfaction of KKT conditions as a measure for a solution being close to a KKT point. Here, we extent this approach for the hydro-thermal power dispatch problem.

4.2.1 Kuhn-Tucker (KT) conditions with subdifferentials

For a solution $\mathbf{x}$ to be Pareto-optimal for a multi-objective optimization problem with $\mathcal{M}$ differentiable objectives, K differentiable equality constraint functions ($h_k(\mathbf{x}) = 0, k = 1, 2, \ldots, K$) and J differentiable inequality constraint functions ($g_j(\mathbf{x}), \leq 0, j = 1, 2, \ldots, J$), it must satisfy the following Kuhn–Tucker optimality conditions:

$$\begin{aligned} \sum_{i=1}^{\mathcal{M}} \lambda_i \nabla f_i(\mathbf{x}) + \sum_{j=1}^{J} u_j \nabla g_j(\mathbf{x}) + \sum_{k=1}^{K} v_i \nabla h_k(\mathbf{x}) &= 0, \\ g_j(\mathbf{x}) &\leq 0, \\ h_k(\mathbf{x}) &= 0, \\ u_j g_j(\mathbf{x}) &= 0, \\ u_j &\geq 0. \end{aligned} \tag{13}$$

One other requirement is that not all λ_i can be zero simultaneously. There is no restriction on the value of v_k, since equality constraints can be either written as $h_k(\mathbf{x}) = 0$ or $-h_k(\mathbf{x}) = 0$,

[1] It is, however, important to mention that EAs are also applicable to many discrete and non-differentiable problems such as combinatorial optimization, problems having non-mathematical objective and constraint functions, etc., in which the above criticism does not hold and an EA's performance is usually measured based on whether the true optimal or the best-known solution is found or not with a reasonable number of function evaluations.

without changing the optimal solution. An important matter to note that variable bounds, if any, must be treated as inequality constraints in formulating the above Kuhn–Tucker conditions.

In the event of non-differentiable objective functions and constraints, the concept of Clarke subdifferentials [3,4] can be used to replace the exact differentials. The Clarke subdifferential is derived from the Clarke directional derivative at $\mathbf{x} \in R^n$ and in the direction $\mathbf{v} \in R^n$ for a locally Lipschitz function f, as follows:

$$f^o(\mathbf{x}, \mathbf{v}) = \lim_{\mathbf{y} \to \mathbf{x}} \sup_{t \to 0+} \frac{f(\mathbf{y} + t\mathbf{v}) - f(\mathbf{y})}{t}, \tag{14}$$

where $\|\mathbf{v}\| = 1, \mathbf{y} \in R^n$ and $t > 0$. For locally Lipschitz function, the right-hand side is bounded and the limit is finite. The Clarke subdifferential is then defined as follows:

$$\partial^c f(\mathbf{x}) = \{\zeta \in R^n : f^o(\mathbf{x}, \mathbf{v}) \geq \langle \zeta, \mathbf{v} \rangle, \forall \mathbf{v} \in R^n\}. \tag{15}$$

In other words, the Clarke subdifferential is a set of all vectors whose component along any direction $\mathbf{v}$ is smaller than or equal to the Clarke directional derivative defined above. For locally Lipschitz functions, an important result is that the Clarke subdifferential is a convex and compact set made up with limiting derivatives $\lim_{i \to \infty} \nabla f(\mathbf{x}^{(i)})$ at neighboring points $\mathbf{x}^{(i)} \to \mathbf{x}$. For example, the function $f(x) = |x|$ is not differentiable at $x = 0$. However, the limiting derivatives for neighboring solutions $x^{(i)} > 0$ is 1 and for $x^{(i)} < 0$ is -1. Thus, any real value in the range $[-1, 1]$ is a Clarke subdifferential of $f(x)$ at $x = 0$. It is interesting to note that Clarke subdifferential contains the $\nabla f(\mathbf{x})$ at a point $\mathbf{x}$, if the function is continuously differentiable.

Thus, there are a couple of changes needed in writing KT conditions for non-differentiable objective functions. The first change must be made in replacing the exact derivative values with the corresponding subdifferentials. This will lead the left side expression of the first KT equation above to represent a convex set (in R^n). The second change must be made in checking if the zero-vector (or origin) is a member of the above convex set.

In our approach of checking if a NSGA-II solution $\mathbf{x}$ is indeed a KT point, we know the exact derivatives of differentiable functions at $\mathbf{x}$ and we also know the constraint values at $\mathbf{x}$. For non-differentiable functions (say $f(\mathbf{x})$), we also know the left side (l_j^{∂}) and right side (u_j^{∂}) limits of the subdifferential values with respect to x_j. We can now associate an unknown $s_j \in [-1, 1]$ with j-th variable and compute the subdifferential with respect to j-th variable, as follows:

$$\partial^c f^j(\mathbf{x}) = 0.5(u_j^{\partial} + l_j^{\partial}) + 0.5 s_j (u_j^{\partial} - l_j^{\partial}). \tag{16}$$

This way, the left side of the first KT condition (we call it the 'KT vector') becomes dependent on the chosen $\mathbf{s}$ vector. Since the zero vector must be a member of the convex set spanned by the left side expression, we can formulate a minimization problem of choosing an appropriate set of ($\mathbf{s}$, $\mathbf{u}$, $\mathbf{v}$, and $\boldsymbol{\lambda}$) vectors so that a norm of the KT vector is zero. If at this condition, every u_j is non-negative and $\boldsymbol{\lambda}$ is a non-zero vector with non-negative components, the NSGA-II solution is indeed a KT point. We describe about the chosen norm a little later. First, we formulate the KT vector for the hydro-thermal dispatch problem.

4.2.2 Derivatives and subdifferentials for hydro-thermal power dispatch problem

The hydro-thermal power dispatch problem has one non-differentiable objective ($f_1(\mathbf{x})$) and one differentiable objective ($f_2(\mathbf{x})$), $n = M(N_h + N_s)$ decision variables, $(M + N_h)$

equality constraints and $2M(N_h + N_s)$ inequality constraints arising due to the use of two variable bounds for each decision variable. We can write the Kuhn–Tucker conditions by separating the equality and inequality constraints into two sets, each with $u = (u^{(L)}, u^{(U)})$ and $v = (v^{(1)}, v^{(2)})$. Power balance constraints given in Eq. 3 are valid for each time period and involve all decision variables involving hydroelectric and thermal power generation units. Water balance constraints given in Eq. 5 involve each hydroelectric power unit for all time periods:

$$\begin{aligned} \mathbf{0} \in\ & \lambda_1 \partial^c f_1(\mathbf{x}) + \lambda_2 \nabla f_2(\mathbf{x}) + \sum_{k=1}^{M} v_k^{(1)} \nabla h_k^{(1)}(\mathbf{x}) + \sum_{l=1}^{N_h} v_l^{(2)} \nabla h_l^{(2)}(\mathbf{x}) \\ & + \sum_{j=1}^{M(N_h+N_s)} u_j^{(L)} \nabla g_j^{(L)}(\mathbf{x}) + \sum_{j=1}^{M(N_h+N_s)} u_j^{(U)} \nabla g_j^{(U)}(\mathbf{x}), \\ h_k(\mathbf{x}) &= 0, \\ g_j(\mathbf{x}) &\le 0, \\ u_j g_j(\mathbf{x}) &= 0, \\ u_j &\ge 0. \end{aligned} \tag{17}$$

Since $g_j^{(L)} = x_j^{(L)} - x_j$, the corresponding derivative is as follows: $\nabla g_j^{(L)} = \{-\delta_{ij} | \forall i = 1, 2, \ldots, n\}$, where $\delta_{ij} = 1$ for $i = j$ and zero otherwise. Similarly, $g_j^{(U)} = x_j - x_j^{(U)}$ and $\nabla g_j^{(U)} = \{\delta_{ij} | \forall i = 1, 2, \ldots, n\}$. For the first set of equality constraints, the i-th component of the derivative vector at k-th time period $(\nabla h_{ki}^{(1)}, i = 1, 2, \ldots, n)$ can be written as follows:

$$\nabla h_{ki}^{(1)}(\mathbf{x}) = \begin{cases} 1 - \sum_{j=1+(k-1)(N_h+N_s)}^{k(N_h+N_s)} & \text{if } i \in [1 + N_h + N_s)(k-1), (N_h + N_s)k], \\ \quad (B_{j'i'} + B_{i'j'}) x_j, & \\ 0, & \text{otherwise}, \end{cases} \tag{18}$$

where $j' = j \mod (N_h + N_s)$ and $i' = i \mod (N_h + N_s)$. Similarly the i-th component of the derivative vector $(\nabla h_{li}^{(2)}, i = 1, 2, \ldots, n)$ for the second set of constraints and for l-th hydroelectric unit is as follows:

$$\nabla h_{li}^{(2)}(\mathbf{x}) = \begin{cases} t_m \left(a_{1l'} + 2a_{2l'} x_i\right), & \text{if } i = \{l + (j-1)(N_h + N_s) | j = 1, 2, \ldots, M\}, \\ 0, & \text{otherwise}, \end{cases} \tag{19}$$

where $l' = l \mod (N_h + N_s)$. Likewise, the i-th component of the second objective can be computed and is nonzero for thermal power units ($\mathcal{T} = \cup_{m=1}^{M} \mathcal{T}_m$) as follows:

$$\nabla (f_2)_i(\mathbf{x}) = \begin{cases} t_m \left(\beta_s + 2\gamma_s x_i + \delta_s \eta_s \exp(\delta_s x_i)\right), & \text{for } i \in \mathcal{T}, \\ 0, & \text{otherwise}. \end{cases} \tag{20}$$

Parameters $\beta_s, \gamma_s, \delta_s$ and η_s are corresponding values for the i-th thermal unit given in the Appendix.

The final term in the first objective function is non-differentiable (due to the presence of the absolute function) and therefore prohibits us to write $\nabla f_1(\mathbf{x})$ term explicitly. Every component of this derivative vector will involve two terms—one for the differentiable part $(\nabla (f_1)_i^1(\mathbf{x}))$ and other for the non-differentiable part $(\nabla (f_1)_i^2(\mathbf{x}))$. The first part is given as follows:

$$\nabla (f_1)_i^1(\mathbf{x}) = \begin{cases} t_m \left(b_s + 2c_s x_i\right), & \text{for } i \in \mathcal{T}, \\ 0, & \text{otherwise}. \end{cases} \tag{21}$$

It is interesting to note that the objective function $f_1(\mathbf{x})$ is a periodic function and the minimum of the function is likely to take place for values for which this periodic function is

close to its minimum. For the absolute sine function, this happens for values for which the argument of the sine function is multiple of π, or importantly the solutions for which the derivative does not exist. We call these locations as 'kinks'. The second term can be written in terms of Clarke's subdifferential, as follows:

$$\nabla(f_1)_i^2(\mathbf{x}) = \begin{cases} -s_i \left(t_m e_s d_s \cos(e_s(x^{\min} - x_i))\right), & \text{for } i \in \mathcal{T}, \\ 0, & \text{otherwise.} \end{cases} \tag{22}$$

In the event of x_i not falling near a kink location, we use $s_i = 1$ if the sine term is positive and $s_i = -1$ if the sine term is negative. However, for x_i falling near a kink, any value of s_i within the range $[-1, 1]$ will qualify it as a representative value, as the derivative of $|x|$ for $x < 0$ is -1 and for $x > 0$ is 1.

4.2.3 Error metric derivation from KT conditions

Having computed all the derivatives, we now discuss the procedure for verifying if a particular solution ($\mathbf{x}^*$) satisfy Kuhn–Tucker conditions numerically. We use the following procedure. For every thermal and hydroelectric power units x_i which does not lie on the lower or the upper bounds ($x^{\min}$ or $x^{\max}$), the corresponding Lagrange multiplier $u_j^{(L)}$ or $u_j^{(U)}$ will be zero due to the presence of $u_j g_j(\mathbf{x}) = 0$ constraint. Thus, we can eliminate all such inequality constraints for further consideration. For the case of extended bounds considered in Sect. 4.1, it turns out that all obtained NSGA-II solutions (see Figs. 7–12) have power units well inside the two bounds and never lie on the bounds. Thus, in our computations here, $u_j = 0$ for all variables.

For every thermal power unit $i \in \mathcal{T}$, we first check to see its position with respect to kink locations. If $\sin(e_s(x_i^{\min} - x_i^*))$ is within $[-0.1, 0.1]$ (bounds are arbitrarily chosen here), we declare it close to the kink and treat it as a non-differentiable case. The derivative value of f_1 with respect to this variable is then declared as $\nabla(f_1)_i = \nabla(f_1)_i^1 + s_i D_i$, where $D_i = -t_m e_s d_s \cos(e_s(x^{\min} - x_i))$ and s_i is an unknown value, but lie within $[-1, 1]$, to agree with the definition of subdifferential for an absolute function. If the value x_i does not lie near a kink by the above definition, we simply use Eq. 22 with s_i value as 1 or -1, depending on the sign of sine term, as discussed above. All other derivatives can be exactly computed and are replaced with their exact values, as given in Eqs. 18–20.

With no inequality constraints contributing to the Kuhn–Tucker conditions, we are now left with two λ parameters and $(M + N_h)$ parameters involving v, as unknowns. However, every derivative vector has $M(N_h + N_s)$ components and this number is much more than number of unknowns. For $M = 4$, $N_h = 2$ and $N_s = 4$, there are eight unknowns and 24 equality constraints given by the Kuhn–Tucker conditions. The Kuhn–Tucker condition set is reduced to the following form:

$$\mathbf{0} \in \lambda_1 \left(\nabla f_1^1(\mathbf{x}) + s_i D_i\right) + \lambda_2 \nabla f_2(\mathbf{x}) + \sum_{k=1}^{M} v_k^{(1)} \nabla h_k^{(1)}(\mathbf{x}) + \sum_{k=1}^{N_h} v_k^{(2)} \nabla h_k^{(2)}(\mathbf{x}). \tag{23}$$

Since we only consider feasible solutions for this analysis, we need not consider $h_k(\mathbf{x}) = 0$ conditions again here. As mentioned earlier, since any value of $s_i \in [-1, 1]$ is allowed here, we can try to find a $\mathbf{s}$ vector for which the KT vector is a zero vector, thereby reducing the above task to a solution of a linear system of equations. To impose non-zero values of both λ_1 and λ_2, we divide the right side expression with λ_2 so that the second vector term is free from unknowns and move the second vector term to the right side to construct a linear system of equations to solve: $A\mathbf{y} = b$. In the specific case with $M = 4$, $N_h = 2$ and $N_s = 4$, $\mathbf{y}$

has seven variables representing the ratio of unknowns, such as $\mathbf{y} = (\lambda_1/\lambda_2, v^{(1)}/\lambda_2, \ldots)^T$ and $b = -\nabla f_2(\mathbf{x})$. Since all terms except the first one in $\mathbf{y}$-vector correspond to equality constraints thereby allowing any sign on their values, only the first term λ_1/λ_2 needs to be checked for its non-negativeness. The matrix A is constructed using the derivative values which are determined exactly, except the first column of A which involves a unknown vector $\mathbf{s}$, representing the cases for which x_i lies near kinks of the absolute sine term. Thus, we can write the first column of A is a function of a number of unknowns lying within $[-1, 1]$, or $A = A(\mathbf{s})$. The number of unknowns are as many as the number of cases ($|\mathbf{s}|$) of x_i lying near a kink. It is important to note that any solution to this linear system will cause the obtained solution $\mathbf{x}^*$ to satisfy the Kuhn–Tucker conditions [3] and is therefore is a likely candidate to the theoretical optimal solution of the original problem.

However, to solve the above vector equation, we have a difficulty. As discussed, the number of unknowns is less than the number of equations. In general, such a system does not have a solution, but we shall try to find an approximate solution to the system by solving a least-square minimization of an error using $\bar{y} = (A^T A)^{-1}(A^T b)$ [41]. Substituting this solution to the original linear system, we have an error vector $e = b - A\bar{y}$. We compute a normalized error metric from this error vector as $\tilde{e}(\mathbf{s}) = \|b - A(\mathbf{s})\bar{y}(\mathbf{s})\|/\|b\|$. Since $\bar{y}$ and A are functions of $\mathbf{s}$, we can now try to minimize this normalized error metric and find optimal value of $\mathbf{s}$:

$$\begin{array}{ll} \text{Minimize} & \tilde{e}(\mathbf{s}) = \|b - A(\mathbf{s})\bar{y}(\mathbf{s})\|/\|b\|, \\ \mathbf{s} & \\ \text{subject to} & s_i \in [-1, 1], \quad \forall i. \end{array} \tag{24}$$

The `fminsearch` procedure of MATLAB optimization toolbox is used to solve the above problem. If the feasible optimal solution to the above problem is close to zero, the original obtained solution $\mathbf{x}^*$ may be considered to have solved the Kuhn–Tucker condition and is a likely candidate to be a Pareto-optimal solution. If $|\mathbf{s}| = 0$, the above optimization is not needed and the normalized error metric value is simply computed by the above objective function ($\tilde{e}(\emptyset)$) equation.

4.2.4 Results for six NSGA-II solutions

To investigate the near-optimality of obtained solutions, we apply the above procedure to six widely separated NSGA-II solutions shown in Table 2, which also lie near the solutions marked in Fig. 4 with a diamond. Since these six solutions are chosen from the entire Pareto-optimal front, satisfaction of KT conditions in these six solutions and the apparent smoothness of the obtained non-dominated front give us confidence about the same for other Pareto-optimal solutions. The results of the above procedure and a single-objective optimization using the `fminsearch` procedure are summarized in Table 3. In all cases, the optimization run is started with an initial solution ($\mathbf{s} = \mathbf{0}$) of all zeros and the procedure is run till no further improvement is found. In all six cases, a small normalized error value (up to 2.41%) is obtained. Moreover, λ_1/λ_2 value in each case is also found to be positive, thereby indicating that all 24 Kuhn–Tucker equality conditions are satisfied adequately. The first solution is close to the minimum-emission solution; hence λ_1 value is found to be zero, thereby indicating no importance to the cost objective and 100% importance to the emission objective provided by this solution. The Pareto-optimal front (Fig. 4) seems to have a zero-slope at this point, thereby agreeing with the 0–100% trade-off corresponding to this minimum-emission solution. On the other hand, the minimum-cost solution (solution 6 in the table) seems to correspond to 12.5–87.5% trade-off between cost and emission objectives. Figure 4 clearly shows that

Table 3 Results of Kuhn–Tucker analysis of NSGA-II solutions at different generations

Soln.	(f_1(\$), f_2(Lb))	|**s**|	i Near kink	**s**	$\tilde{e}$(**s**)	λ_1/λ_2
Feasible solutions at final generation						
1	(92120.45, 22963.09)	0	–	–	0.0024	0.000
2	(89929.82, 22991.40)	0	–	–	0.0169	0.019
3	(84997.68, 23169.00)	2	(5,21)	(0.99, −0.60)	0.0141	0.051
4	(79966.72, 23488.10)	4	(5,9,17,21)	(0.69, −0.92, 0.87,−0.38)	0.0126	0.077
5	(75023.50, 23920.95)	7	(4,5,9,11, 17,21,24)	(0.88, 0.45, −0.86, 0.88, 0.56, −0.63, −0.90)	0.0207	0.099
6	(70018.99, 24467.52)	11	(3,4,5,9, 11,12,15,16, 17,21,24)	(−0.85, 0.92, 0.45, −0.52 0.89, −0.92, −0.86, 0.90 0.57, −0.20, −0.55)	0.0241	0.125
Feasible solutions at generation 10						
1	(83900.92, 24760.04)	0	–	–	0.2915	0.066
2	(82273.16, 25924.39)	1	(12)	(−1)	0.3469	0.051
3	(81281.44, 26298.55)	4	(5,6,9,22)	(−1,1,1,1)	0.3512	−0.078
Feasible solutions at generation 30						
1	(83145.39, 23610.54)	1	(6)	(−1)	0.1265	0.056
2	(73140.35, 24720.39)	4	(3,6,12,18)	(−0.57,1.00,1.00,1.00)	0.1931	0.097
3	(71869.35, 25234.02)	7	(5,6,11,12 16,18,23)	(1.00, −0.640.31, −1.00 1.00,−1.00,1.00)	0.2338	0.105
Feasible solutions at generation 70						
1	(81952.12, 23425.20)	2	(3,23)	(−0.42, 1.00)	0.0515	0.065
2	(74994.34, 24045.11)	4	(6,11,15,17)	(−1.00,0.42,−0.81,0.88)	0.0713	0.096
3	(70424.45, 24922.91)	8	(5,6,11,16 17,18,23,24)	(−0.93,−0.46,0.52,1.00 1.00,−1.00,1.00,−1.00)	0.1459	0.107

the slope of the Pareto-optimal front at the minimum-cost solution is not infinite, thereby suggesting a non-zero weight value for the second objective. Interestingly, there is a steady increase in λ_1 value (meaning more importance to cost objective), as solutions move from the minimum-emission solution (soln. 1) to minimum-cost solution (soln. 6).

To evaluate how the error metric value for these final solutions compare with that computed for solutions in early generations, we have tabulated the error metric values for three feasible solutions chosen from the best non-dominated front of populations at generations 10, 30 and 70 in the table. These solutions are not close to the Pareto-optimal front and, therefore, are not supposed to be expected to satisfy the Kuhn–Tucker conditions. In all but one case, although a positive value of λ_1/λ_2 is obtained, the error value is significantly high. It is interesting to note how the error metric values, as large as 29%–35% at generation 10 reduce with generation to 0.24%–2.41% at the final generation. Some of these solutions reside in the kink regions, thereby causing an non-empty **s** vector. There are two main observations from this study.

1. A feasible solution early in the simulation does not satisfy the Kuhn–Tucker conditions, thereby indicating that these solutions are not close to Pareto-optimal solutions.
2. The gradual decrease in the error metric value with generations suggests that this metric can be used as a termination criterion in an evolutionary optimization simulation.

The eventual reduction of the error metric value close to zero also gives us confidence in the working principle of the proposed evolutionary algorithm. It is also interesting to note that solutions near minimum-emission solution do not involve the non-differentiable terms, as the effect of cost objective (which is the only non-differentiable function in this problem) is contradictory to the emission objective. As the chosen solutions get closer to the minimum-cost solution, more and more variables take values closer to kink locations (column 4 shows the variable numbers for such cases), thereby supporting our argument in Sect. 4 about the relative difficulty in solving the cost function in this problem. For the sixth solution obtained at the final generation, there are 11 (out of 16) thermal power units are near kink locations. For the six final solutions, the variables lying near the kink locations are shown italicized in Table 2. Column 5 in Table 3 shows the obtained **s** values for all the subdifferentials. Since all these values are within $[-1, 1]$, these are acceptable subdifferential values. Thus, we can conclude that these widely-distributed set of solutions on the obtained NSGA-II front are all found to be very closely satisfying the theoretical Kuhn–Tucker conditions. These results give us confidence in our proposed approach and the near-optimality property of the entire obtained NSGA-II frontier.

4.3 Detecting robust regions of optimality

The single-objective optimization results found near the minimum-cost solution (in Fig. 1) indicate that this region is difficult to optimize. Somewhat larger normalized error metric values computed for near minimum-cost solutions in Table 3 also indicate this fact. The cost objective involves a periodic term with the thermal generating units P^s_{sm}. This objective along with quadratic equality constraint makes the optimization task difficult. To investigate the sensitivities of solutions in this region, we perform a *robust* optimization [10] in which variables are perturbed within ± 5 MW around each variable using a uniform distribution in this range. Each solution is then evaluated for 50 perturbations and a normalized change in i-th function value for j-th variable perturbation $(\Delta f_i^{(j)}/f_i)$ is noted. If the maximum normalized change in function values between two neighboring solutions is more than a user-defined threshold value (η), the solution is declared infeasible. Thus, we add a new constraint

$$\max_{i=1}^{2} \max_{j=1}^{50} \frac{\Delta f_i^{(j)}}{f_i} \leq \eta \tag{25}$$

to the original two-objective optimization problem and optimize the problem to find the robust frontier, instead of the Pareto-optimal frontier. This constraint is normalized by bringing η on the left side and dividing the expression the term by η. The constraint violation, if any, is then added to the normalized CV computed by Eq. 10. Unfortunately, the added constraint is non-differentiable and may cause difficulty to a classical gradient-based optimization procedure. But this non-differentiable constraint does not cause much difficulty to an evolutionary optimization procedure, due to its direct use of constraint values as a constraint violation in EA's selection operator.

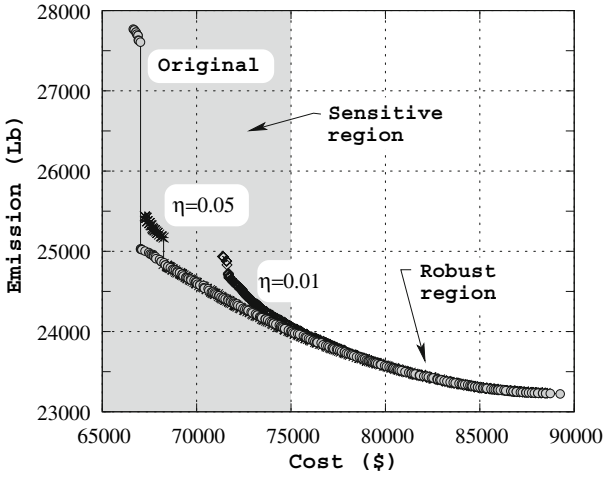

Fig. 5 Robust frontier for the original problem

We use two different values of η and plot the obtained robust frontiers in Fig. 5 along with the original NSGA-II frontier (from Fig. 1). It can be clearly seen that as the threshold (η) is reduced, thereby restraining the normalized change to a small value, the robust front deviates more from the original non-robust frontier, only in the area of minimum-cost solution. The figure shows that solutions costing around 75,000 units or less is sensitive to the variable uncertainties. However, the minimum-emission region remains relatively unperturbed, meaning the near minimum-emission region is robust. A similar analysis on the modified problem with extended bounds is performed and the results are shown in Fig. 6. It can be clearly seen from the figure that the trade-off frontier for the extended problem is fairly robust except a small portion near the minimum-cost region. Interestingly, the robust multi-objective NSGA-II simulation finds a new trade-off robust frontier in the sensitive region. These new solutions, despite being close to the minimum-cost region, pass the sensitivity test (through constraint given in Eq. 25) and are acceptable robust solutions. It is also interesting to note that in the presence of uncertainties in power generation units, the minimum-cost solution is not robust in both cases.

Such a robust optimization strategy is of utmost importance in practice, as in the event of uncertainties in achieving decision variables and problem parameters, engineers and designers are in most situations interested in finding a robust solution which is relatively insensitive to such uncertainties. The above study also reveals that in this problem the minimum-cost solution is relatively more sensitive to such uncertainties than the minimum-emission solution. Due to a large sensitivity of minimum-cost solutions with respect to variable uncertainties, single-objective optimizations shown for cost objective in Fig. 1 were also relatively difficult to attain to optimality.

4.4 Unveiling common principles of operation

To understand better the optimal operating principles of the hydro-thermal power dispatch problem considered here, we analyze the optimized solutions taken from the trade-off frontier (Fig. 4 described in Sect. 4.1) obtained using extended bounds on variables. The procedure of deciphering important design or operating principles in a problem from such *optimal*

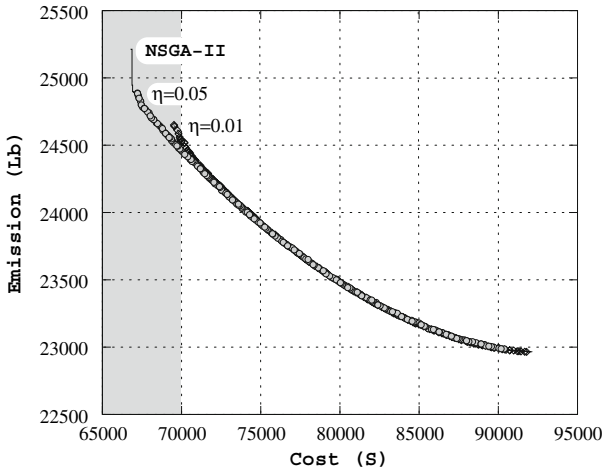

Fig. 6 Robust frontier for the modified problem with extended bounds

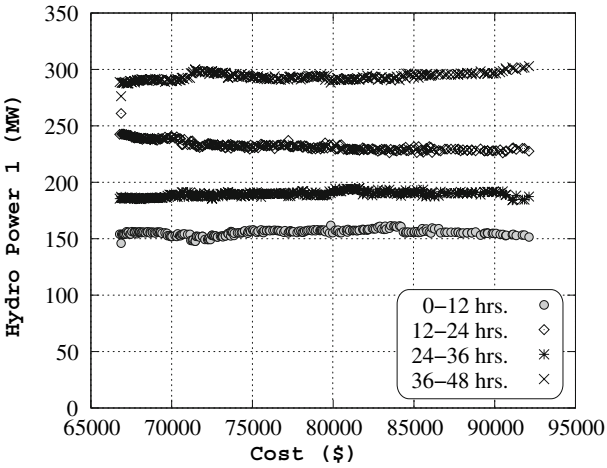

Fig. 7 Hydroelectric unit P_{1m}^{h} versus f_1

trade-off solutions is discussed in detail in another study by the author [12] and was termed as a task of *innovization*—innovation through optimization. Since these solutions are close to KT solutions, it is expected that they will possess some interesting relationships which, when satisfied, would make a solution qualify to a near-optimal solution. Since these solutions are already shown to near-KT solutions, the deciphered common principles are likely to be free from any biases or artifacts. An analysis for deciphering such relationships from obtained near-optimal solutions is one way of unveiling such important properties of optimal solutions in a problem. We perform such an innovization study for the hydro-thermal power dispatch problem here.

Figures 7–12 show the variation of decision variables as they differ with the cost objective. Several interesting behaviors (innovizations) can be observed from these figures:

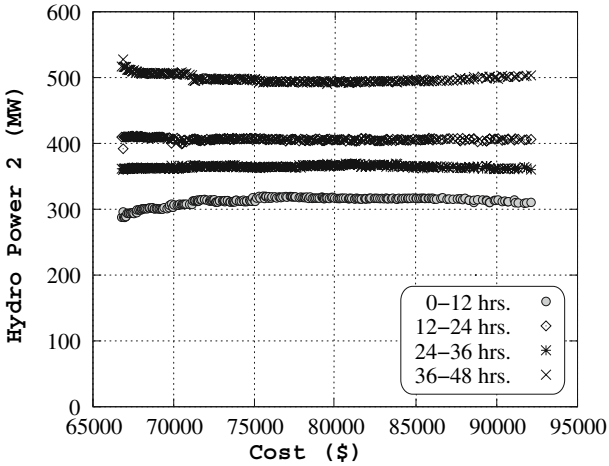

Fig. 8 Hydroelectric unit P_{2m}^{h} versus f_1

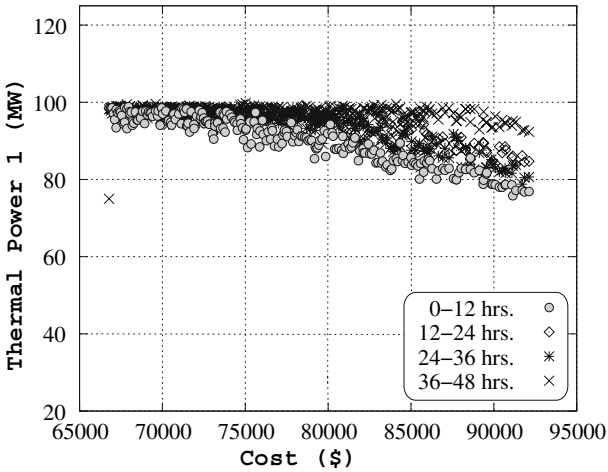

Fig. 9 Thermal unit P_{1m}^{s} versus f_1

1. Output powers for both hydroelectric units are more or less constant for all trade-off optimal solutions for all time slots. The average values are shown below:

	0–12 h	12–24 h	24–36 h	36–48 h
Demand P_{dm} (MW)	900.00	1100.00	1000.00	1300.00
P_{1m}^{h} (MW)	155.45	232.70	189.29	293.39
P_{2m}^{h} (MW)	311.68	406.30	364.37	498.69

From an introspection to the NLP problem given in Eq. 8, such a phenomenon is not obvious. Since these solutions are close to the true optimal solutions, these solutions are special to the objectives considered here and it is not quite surprising that some common principles among such solutions will exist, in general. Analyzing the trade-off solutions obtained using an EMO procedure and deciphering such important principles common

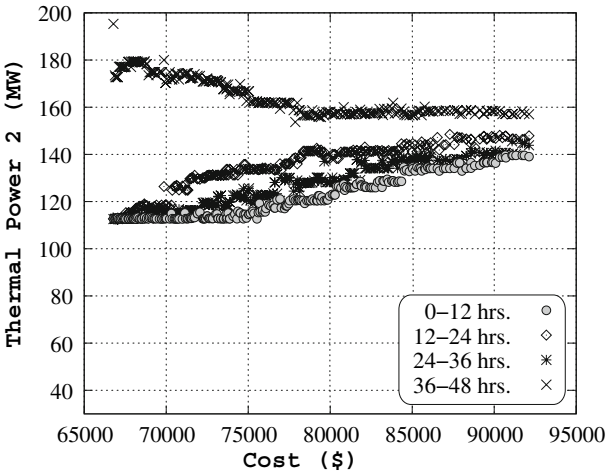

Fig. 10 Thermal unit P^{s}_{2m} versus f_1

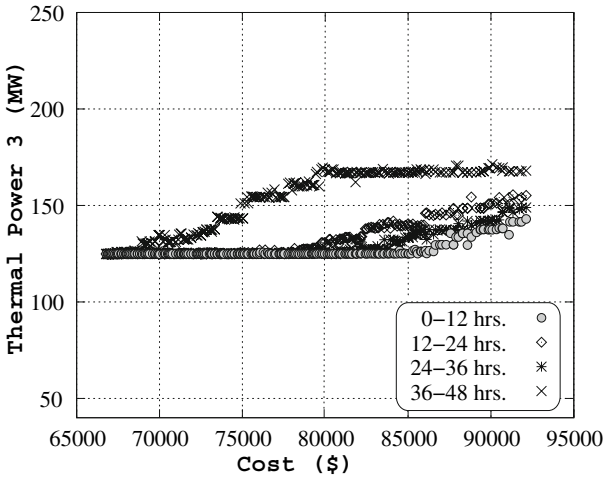

Fig. 11 Thermal unit P^{s}_{3m} versus f_1

to most trade-off solutions stands as a systematic computational procedure for such an important task.

2. The hydroelectric power output values shown above are related to the variation in power demand, as can be seen from the above table. If the demand is more in a time period, the optimal strategy is to increase the hydroelectric power generation for that time period and vice versa.
3. The combined hydroelectric power generation (with only two units) takes care of more than 50% of the total demand in all time periods in this problem.
4. Power output for both hydroelectric units for a particular time period is related to the amount of water availability for each case. For the second hydroelectric unit, the water availability is more, hence the power generation from it in the trade-off solutions also turns out to be more.

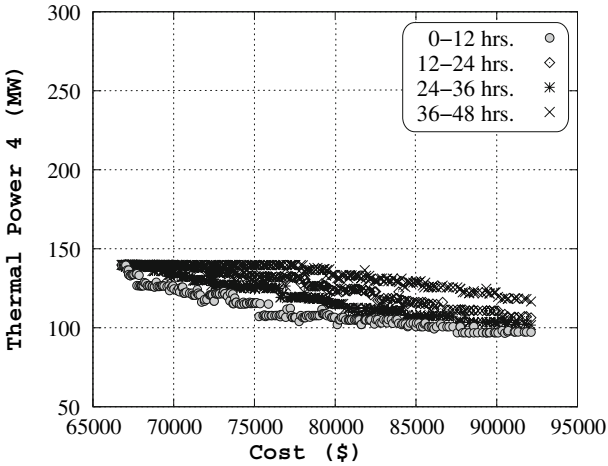

Fig. 12 Thermal unit P^{s}_{4m} versus f_1

5. Like in the case of hydroelectric units, for all four thermal units, the power generation must be increased with the demand in that interval.
6. For better cost solutions, the thermal power generation of units 1 and 4 must be increased. This is because a_s and b_s values for these two units are lower compared to other two thermal units, thereby causing a comparatively smaller increase in the fuel cost with an increase in power generation in these two units. However, the power generation of units 2 and 3 must be decreased (with an exception at the fourth time period for unit 2) due to the opposite reason to that above.
7. For smaller cost solutions, the thermal power generation is more or less the same for all time periods, except the fourth time period in unit 2. This is because due to handling a large demand in this time period, the emission-effective way of reducing the cost is to increase the thermal power generation in unit 2 (due to large negative β_s value associated with this unit).
8. Although a large range of power generation is allowed, the optimal values of these power generation units are concentrated within a small region in the search space. Figures 7–12 are plotted on the entire range of allowable values for each power generation unit, but all trade-off solutions seem to concentrate on a smaller portion of the allowable range.

The above properties of the optimized trade-off solutions are interesting and are, by no means, trivial. Finding the optimized trade-off solutions and then analyzing the solutions for common properties (the so-called innovization process [12]) is a viable procedure of deciphering such important information in complex optimization tasks.

4.5 Scaling-up to larger number of variables

All the above simulations are confined to the $M = 4$ case involving 24 variables and six quadratic equality constraints. To investigate the ability of the proposed optimization procedure in handling a large number of variables and to test its computational efficacy, we increase the number of time periods (M) systematically keeping the overall time window of $T = 48$ h, thereby increasing the number of variables in the resulting multi-objective optimization problem. The number of variables increases as $n = 6M$. We keep $N_h = 2$

and $N_s = 4$ as before, so that the existing data for the hydro-thermal scheduling problem (given in the Appendix) can be still used here. As M increases, the number of quadratic equality constraints increase as $M + 2$, but the two constraints given in Eq. 5 now involves more (M) variables. The number of variable bounds increase as $12M$. The demand in power consumption is assumed to vary in a piece-wise linear manner between the following control points: (Time (h), Demand (MW)): (0, 1,300), (12, 900), (24, 1,100), (36, 1,000) and (48, 1,300). Since all constraints are of equality type, the feasible search space and hence the Pareto-optimal solutions must lie on the intersection of these equality constraints. Moreover, such intersecting points must also lie within the allowable lower and upper bounds of each variable. Although feasible solutions may exist, a satisfaction of all such constraints with an increase in variables becomes a challenging task for any optimization algorithm.

We consider the problem data used in Sect. 4.1. Since a large-variable problem demands a large population size for an evolutionary algorithm starting with a random initial population [19], we use a linear population sizing rule here: $N = 8n$, where n is the number of variables in the problem. All NSGA-II simulations are run for 2,000 generations, to make sure a good convergence, although in all cases population usually comes close to the final trade-off frontier in about 200 generations. Other NSGA-II parameters are the same as those used in Sect. 4.1. Figure 13 shows the obtained fronts up to 480 variables. The figure depicts how better frontiers are found with an increase of number of variables up to 96 variables and then the performance remains more or less the same till about 288-variable problem and then the performance deteriorates with further increase in number of variables. The piece-wise linear demand pattern with time gets better satisfied by considering more time periods (by the use of more variables). Although, a coarse approximation of demand obtained by considering fewer time periods may cause a larger or a smaller cost and emission values compared to that of a finer approximation, in this problem, a finer approximation seems to require smaller objective values. However, the reason for relatively poor performance of the algorithm with further increase in variables (384 and 480) is due to the increase in complexity in finding feasible solutions with larger variables. Figure 14 shows the computational time taken by the NSGA-II procedure to complete 2,000 generations for different problem sizes. It is interesting to note that the time complexity increases polynomially ($O(n^{1.942})$, with an order slightly lower than quadratic) up to the 480-variable problem. Recall that our computation of estimate for the proposed procedure (in Sect. 3.2.1) was between linear and quadratic to the number of variables. Each case was run with 10 different initial populations and minimum and maximum computational times are also plotted. They are so close to each other that in most cases these two limits are not visible in the figure.

When the procedure is applied to a 576-variable problem (for $M = 96$), the algorithm fails to find a feasible solution till 500 generations (using as many as about 2.3 million solution evaluations). Figure 15 shows the proportion of feasible solutions with generation counter for various problem sizes. Up to 480 variables, the proposed procedure is able to find a feasible solution early and then find more feasible solutions exponentially quickly. However, for the 576-variable problem, the proposed procedure could not find a feasible solution in 500 generations. For clarity, the figure is shown only till 50 generations. Although feasible solutions exist for such a large-sized problem, the proportion of them in the entire search space gets smaller with an increase in variable size, and it becomes difficult for an optimization procedure to find even one feasible solution. To be able to solve such a highly-constrained problem, we employ two strategies: (i) increase the variable bounds so that proportion of feasible solutions is more and (ii) use some problem information in initialization and/or in genetic operators so that more feasible solutions can be obtained by the action of these operators. We discuss these techniques in the following subsections, one at a time.

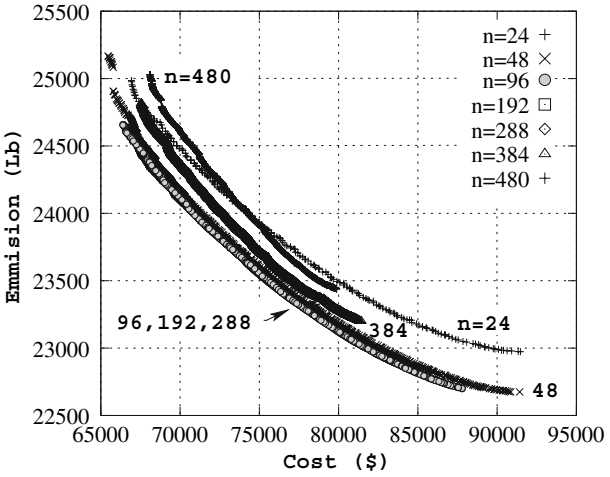

Fig. 13 Obtained non-dominated frontiers are shown for various problem sizes

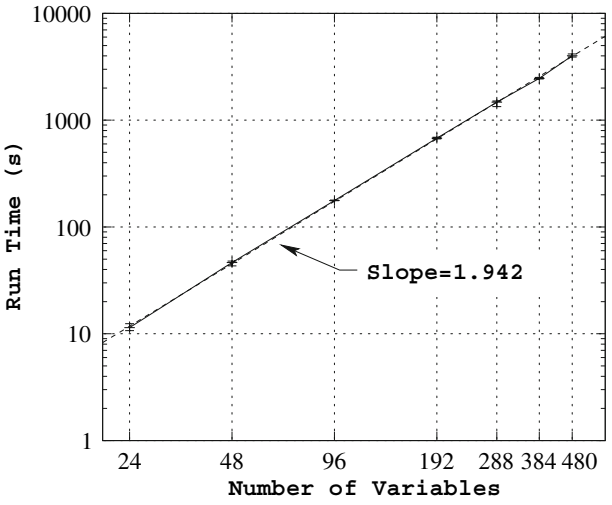

Fig. 14 Computational time with problem sizes (up to 480 variables) shows $O(n^{1.942})$ variation

4.5.1 Extending the variable bounds

We increase the bounds for all thermal power units to [20, 600] MW and keep the bounds for hydroelectric power units the same as before. All other NSGA-II parameters are the same as before. Figure 16 shows the scale-up results with this technique. Now, we are able to solve up to 1,008-variable problem and the time complexity of the procedure is $O(n^{1.917})$, which is also less than quadratic. With a further increase in number of variables, it becomes difficult for the algorithm to find a single feasible solution. To solve such very large-sized problems, a further increase in bounds of thermal power units must have to be considered. Nevertheless, the solution of a non-linear, non-differentiable, and highly-constrained problem having more than 1,000 variables using a population-based evolutionary optimization procedure remains

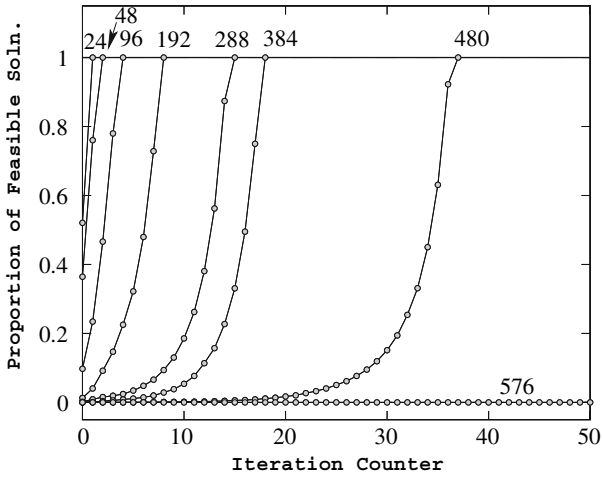

Fig. 15 Proportion of feasible solutions with the original procedure

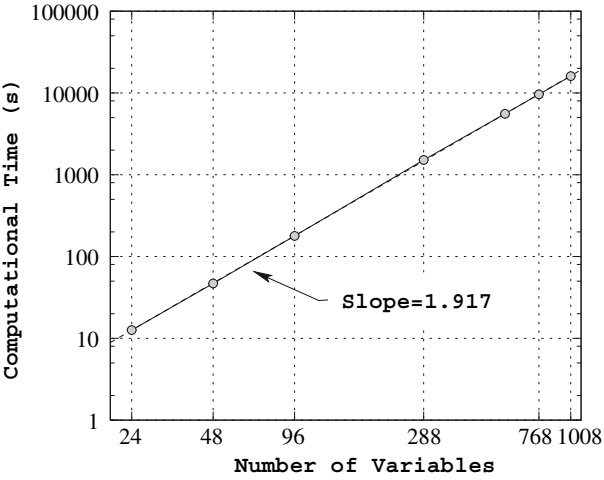

Fig. 16 Computational time complexity (up to 1,008-variable problem) with the extended bounds

as a significant aspect of this study. Next, we discuss a biased initialization procedure which is potential to solve large-sized problems and without making any adjustment in terms of extending the variable bounds.

4.5.2 Using problem information

There is no unique way to introduce problem information in an optimization algorithm. In the past, we were able to solve a million-variable integer linear programming problem involving a real-world casting sequencing task in less than quadratic time complexity with a customized genetic algorithm [11]. If problem information is available, it is always wise to use such information in creating a customized initial population and, if possible, in devising

customized selection, recombination and mutation operators. Here, we use a simple fact to make a biased initialization.

We notice that one way to create one feasible solution to this hydro-thermal power scheduling problem is to first create a solution for which each hydroelectric unit produce identical power in all time periods sharing the water head equally and all N_s thermal units produce identical power during each time period using the power balance Eq. 3. Then if the resulting solution lies within the corresponding variable bounds, this solution becomes a feasible solution. Here, we use the limited variable bounds on thermal units, as used in Sect. 4.1. Thus, assuming P^{h}_{hm} values are identical to P^{h}_{h} for all time periods ($m = 1, 2, \ldots, M$), we obtain the following quadratic equation from water balance constraint Eq. 5:

$$a_{2h}(P^{\mathrm{h}}_h)^2 + a_{1h}P^{\mathrm{h}}_h - (W_h/48 - a_{0h}) = 0, \quad h = 1, 2. \tag{26}$$

Solving, we obtain $P^{\mathrm{h}}_1 = 219.76\,\mathrm{MW}$ and $P^{\mathrm{h}}_2 = 398.11\,\mathrm{MW}$, which are well within the chosen lower and upper bounds for these two variables. The other root in each case is negative and hence is neglected. Using these values in the power balance constraint Eq. 3 and keeping all P^{s}_{sm} values identical to P^{s}_s for all thermal units within a particular time period, we obtain the following quadratic equation:

$$\left(\sum_{i=1}^{4}\sum_{i=1}^{4} B_{ij}\right)(P^{\mathrm{s}}_s)^2 + \left(\sum_{i=1}^{4}\sum_{j=1}^{2}(B_{ij} + B_{ji})P^{\mathrm{h}}_j - 6\right)P^{\mathrm{s}}_s + \left(\sum_{i=1}^{2}\sum_{j=1}^{2} B_{ij}P^{\mathrm{h}}_i P^{\mathrm{h}}_j - P^{\mathrm{h}}_1 - P^{\mathrm{h}}_2 + P_{Dm}\right) = 0, \quad m = 1, 2, \ldots, t_m. \tag{27}$$

The solution to the above equation will depend on the power demand P_{Dm} in that time period m. We choose the smallest positive root of the above equation in each case. The minimum and maximum demand values used in this study are 900 and 1,300 MW, respectively. For these two values of P_{Dm}, the corresponding P^{s}_s values (minimum value of the two roots) are 50.13 and 118.82 MW, respectively. It is interesting to note that these values are within the chosen lower and upper bounds for thermal power units, except that for the fourth thermal power unit the chosen lower bound of 50 MW (refer to the appendix) is precariously close to the above-found $P^{\mathrm{s}}_s = 50.13\,\mathrm{MW}$. Although it makes the above solution feasible for any number of time periods (thereby ensuring feasibility for any number of variables for this problem), arriving at this solution for larger problem sizes using any optimization algorithm becomes an increasingly more difficult task.

In the biased initialization, we simply enforce one copy of the above solution (P^{s}_s value for each time period is obtained using the exact P_{Dm} value for this time period) in the initial population and form other population members at random, as before. The population size and other NSGA-II parameters are kept the same as before. Having such a good solution in the initial population causes other feasible solution in its vicinity to be created quickly by EA operations. Figure 17 shows the increase in number of feasible solutions in the NSGA-II run for up to 1,008 variables. The complete population becomes feasible in less than about 50 generations in all cases, despite the use of limited range of allowable variable bounds. Recall that the original unbiased initialization procedure could not find a feasible solution for more than 480-variable problem.

Figure 18 shows the scaling of the algorithm with a number of decision variables. In a wide range of 24–1,008 variables, the proposed procedure is able to find a widely distributed Pareto-optimal front in less than quadratic time complexity ($O(n^{1.911})$). This biased initialization

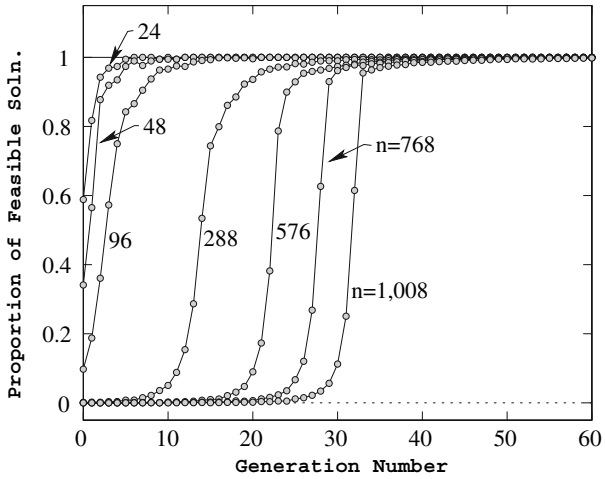

Fig. 17 Proportion of feasible solutions with the biased initialization procedure

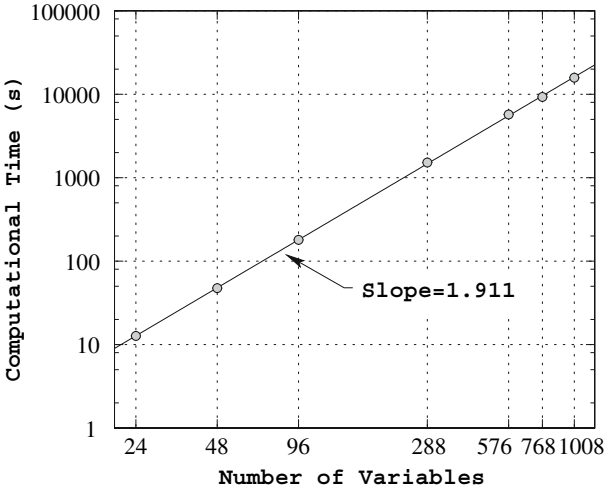

Fig. 18 Computational time complexity (up to 1,008-variable problem) with the biased initialization procedure

procedure is also able to solve larger than 1,008 variable problems, for brevity, we do not discuss those results here.

4.6 Trade-off solutions

In this subsection, we discuss the properties of different trade-off solutions obtained for two conflicting objectives considered in this study. For this purpose, we consider the $M = 48$ (288-variable) problem. In this problem, we approximate the 48-h time span into 1-h time period of statis. The corresponding Pareto-optimal front showing the entire trade-off relationship between cost and emission is shown in Fig. 19. We pick three solutions (A,

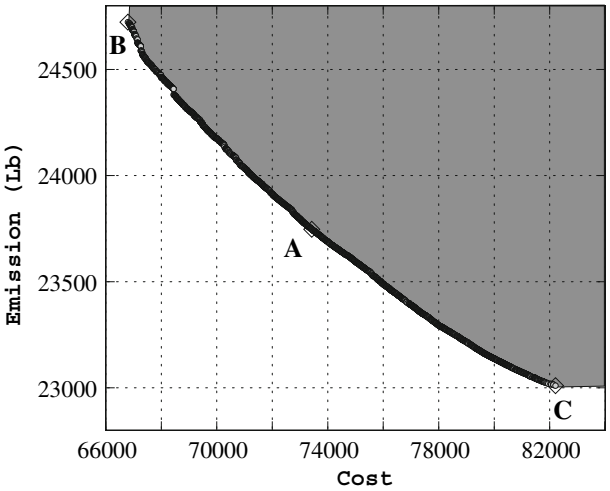

Fig. 19 Obtained non-dominated front for 48 time period problem (288 variables). Three solutions are chosen from the frontier for cost-emission analysis

B, and C) from this front, corresponding to almost 50–50%, 100–0%, and 0–100% trade-offs, respectively, between the objectives. It is interesting to note that these solutions are not the optimal solutions corresponding to the weighted objective with above weight values. The above weights are pseudo-weights [6], representing relative weights with respect to the entire range of Pareto-optimal solutions. For example, the 50–50% solution represents a solution closest to the middle point of the Pareto-optimal front in each objective axis. It is also important to note that such a solution is only possible to find if the entire Pareto-optimal front is found first. If both ideal and nadir points can be obtained by separate computational procedures, such an intermediate solution can also be found by performing a single-objective study.

The corresponding solutions (24 variables) are shown in Figs. 20–22, respectively. It is interesting to note that in all three cases the sum of all six power generations meet the changing power demand. The slight difference between the demand and power generation is due to meeting the power loss term in constraint Eq. 3. The overall cost and emission values for each of three solutions are presented in the following table. As expected, when

Case (%)	Cost ($)	Emission (Lb)
50–50	73, 424.66	23, 746.51
100–0	66, 795.07	24, 722.47
0–100	82, 209.90	23, 010.16

100% importance is given to cost objective (second row), the overall cost is minimum and when 100% importance to emission objective (third row) is given, minimum emission value is obtained. The 50–50% case (first row) makes a compromise between emission values.

Another interesting aspect to note that the minimum-cost (100–0% case) solution (Fig. 21) to require abrupt changes in power generation values with time compared to that the minimum-emission solution, a matter which agrees well with our previous finding about

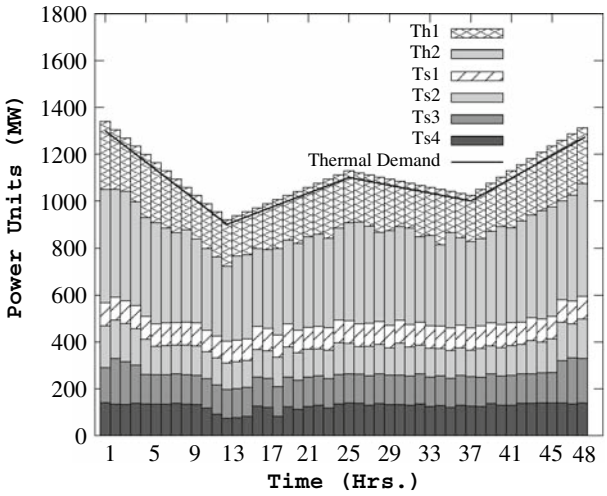

Fig. 20 All six power generations for 48, 1-h time periods are shown for the 50–50% trade-off solution (marked as 'A')

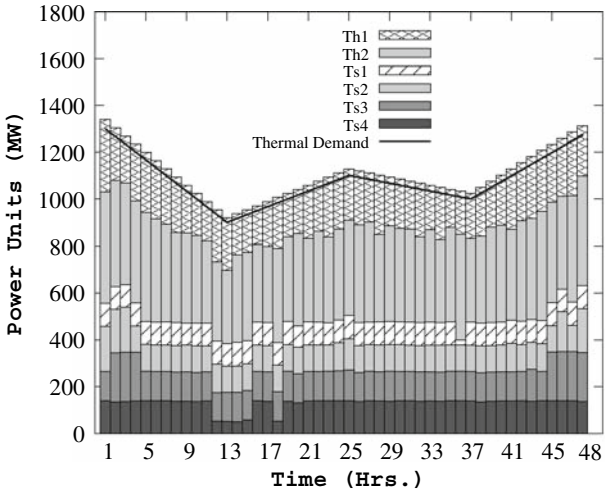

Fig. 21 All six power generations for 48, 1-h time periods are shown for the 100–0% trade-off solution (marked as 'B')

the sensitivity of minimum-cost region of the obtained Pareto-optimal frontier to parameter fluctuations. It is such abrupt change in such isolated solutions which causes constraints and variable bounds to get barely satisfied to make the solution feasible and achieve smallest value of the cost term. Such sensitive solutions are often of not much importance in practice. The robust multi-objective procedure described in Sect. 4.3 demonstrates a way to arrive at a trade-off frontier which is different from the theoretical Pareto-optimal frontier and which contains robust solutions only.

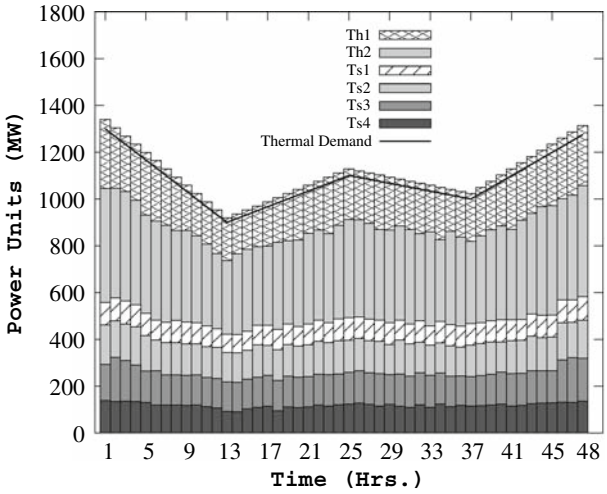

Fig. 22 All six power generations for 48, 1-h time periods are shown for the 0–100% trade-off solution (marked as 'C')

5 Conclusions

In this paper, we have demonstrated the power of population-based evolutionary optimization procedures for handling complex and difficult aspects associated with a hydro-thermal power dispatch optimization problem. The problem is challenging to any optimization algorithm due to the presence of non-linear and non-differentiable objective and constraint functions, uncertainties in decision variables, multiple objectives, and a large number of variables. Following important conclusions can be drawn from this study:

1. The study has shown how multiple trade-off multi-objective optimal solutions can be found and a systematic verification by a number of other single-objective optimization tasks and by theoretical means can be achieved to build confidence in obtained solutions.
2. The study has shown how multiple trade-off solutions can be used to decipher important insights about the problem which has provided a clear idea of the properties needed for a solution to become an optimal solution. This is possible only by finding and analyzing a set of trade-off optimal solutions, instead of a single optimal solution.
3. The study has also shown how *robust* solutions (which are relatively insensitive to parameter uncertainties) can be found and good regions of robust solutions in a multi-objective scenario can be identified.
4. The study has also demonstrated the importance of using problem information in large-scale problem-solving tasks.

All these optimization tasks have tremendous practical importance and can be followed in other such complex problems. Practical problems often possess different complexities (non-linearities and non-differentiabilities in objectives and constraints, dimensionality in objectives, constraints and decision variables, strong correlations among variables, etc.), which EAs (or any other algorithms) do not necessarily guarantee handling efficiently in all problems. However, this paper has shown how systematic and efficient procedures can be devised to handle some such vagaries in practical problems in terms of performing a parametric study involving an algorithm's parameters, a sensitivity analysis of solutions

around optima, a multi-objective optimization involving conflicting objectives, procedures of introducing problem information to expedite execution of optimization algorithms, and an *innovization* task of unveiling a number of important properties of trade-off optimal solutions. Additionally, this study has emphasized the importance of verifying EA-optimized solutions for real-parameter optimization using theoretical Kuhn–Tucker conditions involving gradients for differentiable objectives and constraints, and using subdifferentials for non-differentiable objective functions. Due to systematic consideration of both theoretical and a wide variety of practical aspects, this study should remain as a clear demonstration of power and efficiency of evolutionary population-based search procedures in complex and large-scale optimization problem-solving activities. Importantly, a detail analysis using various optimization tasks in a problem reveal salient insights about relative importance of constraints, variable bounds and objectives and the parameters involved therein, thereby providing a plethora of knowledge about the problem to a user, which may not be possible to obtain in any other manner. As a by-product of using practical considerations and theoretical aspects in this paper, the study should also cause both classical and evolutionary optimization researchers and practitioners to engage in more useful collaborative research efforts in the coming years.

Acknowledgments Discussions with Dr. Joydeep Dutta on theoretical analysis of obtained solutions, and Barnali Kar and Uday Bhaskar Rao for hydro-thermal power dispatch problem are appreciated. The support from the Academy of Finland and Foundation of Helsinki School of Economics (grant #118319) is appreciated.

Appendix

Parameters for hydro-thermal power dispatch problem

Hydroelectric system data

Unit	a_{0h} (acre-ft/h)	a_{1h} (acre-ft/MWh)	a_{2h} (acre-ft/$(MW)^2$h)	W_h (acre-ft)	$P^{\mathrm{h}}_{h,\min}$ (MW)	$P^{\mathrm{h}}_{h,\max}$ (MW)
1	260	8.5	0.00986	125000	0	250
2	250	9.8	0.01140	286000	0	500

Cost related thermal system data

Unit	a_s ($/h)	b_s ($/MWh)	c_s ($/$(MW)^2$h)	d_s ($/h)	e_s (1/MW)	$P^{\mathrm{s}}_{s,\min}$ (MW)	$P^{\mathrm{s}}_{s,\max}$ (MW)
3	60.0	1.8	0.0030	140	0.040	20	125
4	100.0	2.1	0.0012	160	0.038	30	175
5	120.0	2.1	0.0010	180	0.037	40	250
6	40.0	1.8	0.0015	200	0.035	50	300

Emission related thermal system data

Unit	$\alpha_s(Lb/h)$	$\beta_s(Lb/MWh)$	$\gamma_s(Lb/(MW)^2h)$	$\eta_s(Lb/h)$	$\delta_s(1/MW)$
3	50	−0.555	0.0150	0.5773	0.02446
4	60	−1.355	0.0105	0.4968	0.02270
5	45	−0.600	0.0080	0.4860	0.01948
6	30	−0.555	0.0120	0.5035	0.02075

$$B = \begin{bmatrix} 0.000049 & 0.000014 & 0.000015 & 0.000015 & 0.000020 & 0.000017 \\ 0.000014 & 0.000045 & 0.000016 & 0.000020 & 0.000018 & 0.000015 \\ 0.000015 & 0.000016 & 0.000039 & 0.000010 & 0.000012 & 0.000012 \\ 0.000015 & 0.000020 & 0.000010 & 0.000040 & 0.000014 & 0.000010 \\ 0.000020 & 0.000018 & 0.000012 & 0.000014 & 0.000035 & 0.000011 \\ 0.000017 & 0.000015 & 0.000012 & 0.000010 & 0.000011 & 0.000036 \end{bmatrix} perMW.$$

References

1. Basu, M.: A simulated annealing-based goal-attainment method for economic emission load disptch of fixed head hydrothermal power systems. Electr. Pow. Energy Syst. **27**(2), 147–153 (2005)
2. Bazaraa, M.S., Sherali, H.D., Shetty, C.M.: Nonlinear Programming: Theory and Algorithms. Wiley, Singapore (2004)
3. Bector, C.R., Chandra, S., Dutta, J.: Principles of Optimization Theory. Narosa, New Delhi (2005)
4. Clarke, F.H.: Optimization and Nonsmooth Analysis. Wiley-Interscience (1983)
5. Deb, K.: An efficient constraint handling method for genetic algorithms. Comput. Meth. Appl. Mech. Eng. **186**(2–4), 311–338 (2000)
6. Deb, K.: Multi-objective Optimization using Evolutionary Algorithms. Wiley, Chichester (2001)
7. Deb, K., Agrawal, R.B.: Simulated binary crossover for continuous search space. Complex Syst. **9**(2), 115–148 (1995)
8. Deb, K., Agrawal, S., Pratap, A., Meyarivan, T.: A fast and elitist multi-objective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**(2), 182–197 (2002)
9. Deb, K., Goyal, M.: A combined genetic adaptive search (GeneAS) for engineering design. Comput. Sci. Infor. **26**(4), 30–45 (1996)
10. Deb K., Gupta H.: Searching for robust Pareto-optimal solutions in multi-objective optimization. In: Proceedings of the Third Evolutionary Multi-Criteria Optimization (EMO-05) Conference (Also Lecture Notes on Computer Science 3410), pp. 150–164 (2005)
11. Deb, K., Reddy, A.R., Singh, G.: Optimal scheduling of casting sequence using genetic algorithms. J. Mater. Manuf. Process. **18**(3), 409–432 (2003)
12. Deb, K., Srinivasan, A.: innovization: Innovating design principles through optimization. In: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2006), pp. 1629–1636. The Association of Computing Machinery (ACM), New York (2006)
13. Fletcher, R., Reeves, C.M.: Function minimization by conjugate gradients. Comput. J. **7**, 149–154 (1964)
14. Fonseca, C.M., Flemming, P.J.: Genetic algorithms for multi-objective optimization: formulation, discussion, and generalization. In: Proceedings of the Fifth International Conference on Genetic Algorithms, pp. 416–423 (1993)
15. Fox, R.L.: Optimization Methods for Engineering Design. Addison-Wesley, Reading (1971)
16. Goldberg, D.E.: Computer-Aided Gas Pipeline Operation Using Genetic Algorithms and Rule Learning. PhD thesis, University of Michigan, Ann Arbor. Dissertation Abstracts International 44(10), 3174B (University Microfilms No. 8402282) (1983)
17. Goldberg, D.E.: Genetic Algorithms for Search, Optimization, and Machine Learning. Addison-Wesley, Reading (1989)
18. Goldberg, D.E.: The Design of Innovation: Lessons from and for competent genetic algorithms. Kluwer Academic Publishers (2002)
19. Goldberg, D.E., Deb, K., Clark, J.H.: Genetic algorithms, noise, and the sizing of populations. Complex Syst. **6**(4), 333–362 (1992)
20. Goldberg, D.E., Samtani, M.P.: Engineering optimization via genetic algorithms. In: Proceedings of the Ninth Conference on Electronic Computations, ASCE, pp. 471–482 (1986)
21. Grefenstette J.J. (ed.): Proceedings of the 1st International Conference on Genetic Algorithms. Lawrence Erlbaum Associates, Inc., Mahwah (1985)
22. Grefenstette, J.J. (ed.): Proceedings of the 2nd International Conference on Genetic Algorithms, Cambridge, MA, USA, July 1987. Lawrence Erlbaum Associates (1987)
23. Harik, G., Cantú-Paz, E., Goldberg, D.E., Miller, B.L.: The gambler's ruin problem, genetic algorithms, and the sizing of populations.. Evol. Comput. J. **7**(3), 231–254 (1999)
24. Holland, J.H.: Outline for a logical theory of adaptive systems. J. ACM **9**(3), 297–314 (1962)
25. Holland, J.H.: Adaptation in Natural and Artificial Systems. MIT Press, Ann Arbor (1975)

26. Horn, J., Nafploitis, N., Goldberg, D.E.: A niched Pareto genetic algorithm for multi-objective optimization. In: Proceedings of the First IEEE Conference on Evolutionary Computation. pp. 82–87 (1994)
27. Michalewicz, Z.: Genetic Algorithms + Data Structures = Evolution Programs. Springer-Verlag, Berlin (1992)
28. Michalewicz, Z.: A survey of constraint handling techniques in evolutionary computation methods. In: Proceedings of the 4th Annual Conference on Evolutionary Programming, pp. 135–155. MIT Press, Cambridge (1995)
29. Miettinen, K.: Nonlinear Multiobjective Optimization. Kluwer, Boston (1999)
30. Mühlenbein, H., Schlierkamp-Voosen, D.: The science of breeding and its application to the breeder genetic algorithm (bga). Evol. Comput. **1**(4), 335–360 (1993)
31. Orero, S.O., Irving, M.R.: A genetic algorithm modeling framework and solution technique for short term optimal hydrothermal scheduling. IEEE Trans. Power Syst. **13** (2), 501–516 (1998)
32. Rao, S.S.: Genetic algorithmic approach for multiobjective optimization of structures. In: Proceedings of the ASME Annual Winter Meeting on Structures and Controls Optimization, vol. 38, pp. 29–38 (1993)
33. Rashid, A.H.A., Nor, K.M.: An efficient method for optimal scheduling of fixed head hydro and thermal plants. IEEE Trans. PWRS **6** (2), 632–636 (1991)
34. Reklaitis, G.V., Ravindran, A., Ragsdell, K.M.: Engineering Optimization Methods and Applications. Wiley, New York (1983)
35. Rockafellar, R.T.: Convex Analysis. Princeton University Press, Princeton (1970)
36. Rudolph, G.: Convergence analysis of canonical genetic algorithms. IEEE Trans. Neural Netw. **5**(1), 96–101 (1994)
37. Schaffer, J.D.: Some Experiments in Machine Learning Using Vector Evaluated Genetic Algorithms. PhD thesis, Vanderbilt University, Nashville (1984)
38. Shukla, P., Deb, K.: Comparing classical generating methods with an evolutionary multi-objective optimization method. In: Proceedings of the Third International Conference on Evolutionary Multi-Criterion Optimization (EMO-2005), pp. 311–325. Lecture Notes on Computer Science 3410 (2005)
39. Sinha, N., Chakraborty, R., Chattopadhyay, P.K.: Fast evolutionary programming techniques for short-term hydrothermal scheduling. Electric Pow. Syst. Res. **66**, 97–103 (2003)
40. Srinivas, N., Deb, K.: Multi-objective function optimization using non-dominated sorting genetic algorithms. Evol. Comput. J. **2**(3), 221–248 (1994)
41. Strang, G.: Linear Algebra and its Applications. Academic Press, Orlando (1980)
42. Wolpert, D.H., Macready, W.G.: No free lunch theorems for optimization. IEEE Trans. Evol. Comput. **1**(1), 67–82 (1977)
43. Wong, K.P., Wong, Y.W.: Short-term hydrothermal scheduling part i: simulated annealing approach. IEE Proc. Gener. Trans. Distrib **141**(5), 497–501 (1994)
44. Wood, A.J., Woolenberg, B.F.: Power Generation, Operation and Control. John-Wiley & Sons (1986)
45. Yang, J.S., Chen, N.: Short-term hydrothermal co-ordination using multipass dynamic programming. IEEE Trans. PWRS **4**(3), 1050–1056 (1989)
46. Zaghlool, M.F., Trutt, F.C.: Efficient methods for optimal scheduling of fixed head hydrothermal power systems. IEEE Trans. PWRS **3**(1) (1988)
47. Zoutendijk, G.: Methods of Feasible Directions. Elsevier, Amsterdam (1960)