Dynamic Multi-Objective Optimization and Decision-Making Using Modified NSGA-II: A Case Study on Hydro-Thermal Power Scheduling

Kalyanmoy Deb, Udaya Bhaskara Rao N., and S. Karthik

Kanpur Genetic Algorithms Laboratory (KanGAL)
Indian Institute of Technology Kanpur, PIN 208016, India
deb@iitk.ac.in, uday.iitk@gmail.com, ksindhya@iitk.ac.in

KanGAL Report Number 2006008

Abstract. Most real-world optimization problems involve objectives, constraints, and parameters which constantly change with time. Treating such problems as a stationary optimization problem demand the knowledge of the pattern of change a priori and even then the procedure can be computationally expensive. Although dynamic consideration using evolutionary algorithms has been made for single-objective optimization problems, there has been a lukewarm interest in formulating and solving dynamic multi-objective optimization problems. In this paper, we modify the commonly-used NSGA-II procedure in tracking a new Pareto-optimal front, as soon as there is a change in the problem. Introduction of a few random solutions or a few mutated solutions are investigated in detail. The approaches are tested and compared on a test problem and a real-world optimization of a hydro-thermal power scheduling problem. This systematic study is able to find a minimum frequency of change allowed in the problem for two dynamic EMO procedures to adequately track the Pareto-optimal frontiers on-line. Based on these results, this paper also suggests an automatic decision-making procedure for arriving at a dynamic single optimal solution on-line.

1 Introduction

A dynamic optimization problem involves objective functions, constraint functions and problem parameters which can change with time. Such problems often arise in real-world problem solving, particularly in optimal control problems or problems requiring an on-line optimization. There are two computational procedures usually followed. In one approach, optimal control laws or rules are evolved by solving an off-line optimization problem formed by evaluating a solution on a number of real scenarios of the dynamic problem [10]. This approach is useful in problems which are computationally too expensive for any optimization algorithm to be applied on-line. The other approach is a direct optimization procedure on-line. In such a case, the problem is considered stationary for some time period and an optimization algorithm be allowed to find optimal or near-optimal solution(s) within the time span in which the problem remains stationary. Thereafter, a new problem is constructed based on the current problem scenario and a new optimization is performed for the new time period. Although this procedure is approximate due to the static consideration of the problem during the time for optimization, efforts are made to develop efficient optimization algorithms which can track the optimal solution(s) within a small number of iterations so that the required time period for fixing the problem is small and the approximation error is reduced. In this paper, we consider solving dynamic optimization problems having more than one objective functions using the direct on-line optimization procedure described above.

Although single-objective dynamic optimization have received some attention in the past [2], the dynamic multi-objective optimization is yet to receive a significant attention. When a multi-objective optimization problem changes with time in stepped manner, the task of an dynamic EMO procedure is to find or track the Pareto-optimal front as and when there is a change. After the idea is put forward earlier [5], there has been a lukewarm interest on this topic [7, 6]. In this paper, we suggest two variations of NSGA-II for tracking new Pareto-optimal frontiers. Whenever there is a change detected in a problem, addition of random solutions or mutated solutions to existing population members are tried. The effect of frequency of change in a problem and the proportion of added random or mutated solutions are parameters which are systematically studied to evaluate the developed procedures for their tracking efficiency. The proposed NSGA-II procedures are applied to a complex hydro-thermal power scheduling problem involving two conflicting objectives. The change in problem appears due to a change in demand in power with time. The efficacy of modified NSGA-II procedures is illustrated by finding the smallest frequency of change which can be allowed before the EMO procedure can track the optimal front in with a significant confidence. Finally, a decision-making aid is coupled with the dynamic NSGA-II procedures to help identify one solution from the obtained front automatically (on-line). Interesting conclusions about the particular problem and about dynamic multi-objective optimization problem, in general, are made from this study.

2 Dynamic Problems as On-line Optimization Problems

Many search and optimization problems in practice change with time and therefore must be treated as an on-line optimization problems. The change in the problem with time t can be either in its objective functions or in its constraint functions or in its variable boundaries or in any combination of above. Such an optimization problem ideally must be solved at every time instant t or whenever there is a change in any of the above functions with t. In such optimization problems, the time parameter can be mapped with the iteration counter τ of the optimization algorithm. One difficulty which may arise in solving the above on-line optimization task is that the underlying optimization algorithm may not get too many iterations to find the optimal solutions before there is a change in the problem. If the change is too frequent, the best hope of an optimization task is to *track* the optimal solutions as closely as possible within the time span allowed to iterate. However, for steady changes in a problem (which is usually the case in practice), there lies an interesting trade-off which we discuss next. Let us assume that the change in the optimization problem is gradual in t. Let us also assume that each optimization iteration requires a finite time G to execute and that τ_T iterations are needed (or allowed) to track the optimal frontier. Here, we assume that problem does not change (or assumed to be constant) within a time interval t_T, and $G\tau_T < t_T$. Here, initial $G\tau_T$ time is taken up by the optimization algorithm to track the new trade-off frontier and to make a decision for implementing a particular solution from the frontier. Here, we choose $\alpha = G\tau_T/t_T$ to be a small value (say 0.25), such that after the optimal frontier is tracked, $(1-\alpha)t_T$ time is spent on using the outcome for the time period. Figure 1 illustrates this dynamic procedure. Thus, if we allow a large value of t_T (allowing a large number of optimization iterations τ_T), a large change in the problem is expected, but the change occurs only after a large number of iterations of the optimization algorithm. Thus, despite the large change in the problem, the optimization algorithm may have enough iterations to track the trade-off optimal solutions. On the other hand, if we choose a small τ_T, the change in the problem is frequent (which approximates the real scenario more closely), but a lesser number of iterations are allowed to track new optimal solutions for a problem which has also undergone a small change. Obviously, there lies a lower limit to τ_T below which, albeit a small change in the problem, the number of iterations are not enough for an algorithm to track the new optimal solutions adequately. Such a limiting τ_T will depend on the nature of the dynamic problem and the chosen algorithm, but importantly allows the best scenario (and closest approximation to the original problem) which an algorithm can achieve. Here, we investigate this aspect in the context of dynamic multi-objective optimization problem and find such a limiting τ_T for two variants of NSGA-II algorithm. The procedure adopted in this study is generic and can be applied to other dynamic optimization problems as well.

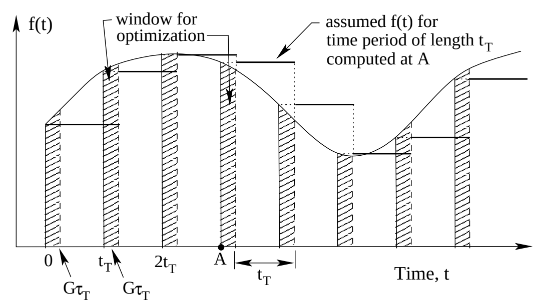

Fig. 1. The on-line optimization procedure adopted in this study. For simplicity, only one objective is shown.

The above procedure can be applied to its extreme as well. If we allow the problem to change as frequently as the time needed to complete one iteration of the optimization algorithm (that is, $t_T = \tau_T = 1$, yielding $G = 1$), we have a true on-line optimization procedure in which the problem changes continuously with generation counter.

3 Proposed Modifications to NSGA-II

We make some changes to the original NSGA-II procedure to handle dynamic optimization problems. First, we introduce a test to identify whether there is a change in the problem. For this purpose, we randomly pick a few solutions from the parent population (10% population members) and re-evaluate them. If there is a change in any of the objectives and constraint functions, we establish that there is a change in the problem. In the event of a change, all parent solutions are re-evaluated before merging parent and child population into a bigger pool. This process allows both offspring and parent solutions to be evaluated using the changed objectives and constraints.

In the first version (DNSGA-II-A) of the proposed dynamic NSGA-II, we introduce new random solutions whenever there is a change in the problem. A ζ% of the new population is replaced with randomly created solutions. This helps to introduce new (random) solutions whenever there is a change in the problem. This method may perform better in problems undergoing a large change in the objectives and constraints. In the second version (DNSGA-II-B), instead of introducing random solutions, ζ% of the population is replaced with mutated solutions of existing solutions (chosen randomly). This way, the new solutions introduced in the population are related to the existing population. This method may work well in problems undergoing a small change in the problem.

4 Simulation Results on a Test Problem

Farina, Deb and Amato [5] proposed five dynamic test problems. FDA2 is a Type-II unconstrained problem, in which the Pareto-optimal front changes from convex to non-convex shapes in the objective space with time and a part of the decision variables ($\mathbf{x}_{III}$) also changes with time. Here is a modified version of FDA2:

$$\begin{aligned}
&\text{Minimize } f_1(\mathbf{x}_I) = x_1,\\
&\text{Minimize } f_2(\mathbf{x}) = g \times h,\\
&\text{where} \quad g(\mathbf{x}_{II}) = 1 + \sum_{x_i \in X_{II}} x_i^2, \quad h(\mathbf{x}_{III}, f_1, g) = 1 - \left(\frac{f_1}{g}\right)^{2^{\left(H(t) + \sum_{x_i \in X_{III}} (x_i - H(t)/4)^2\right)}},\\
&\qquad H(t) = 2\sin(0.5\pi(t-1)), \quad t = 2\left\lfloor \frac{\tau}{\tau_T} \right\rfloor \frac{\tau_T}{\tau^{\max} - \tau_T},\\
&\qquad \mathbf{x}_I = x_1 \in [0,1], \quad \mathbf{x}_{II}, \mathbf{x}_{II} \in [-1,1].
\end{aligned} \tag{1}$$

There are five variables in $\mathbf{x}_{II}$ and seven variables in $\mathbf{x}_{III}$, thereby making a total of 13 variables. Here we use a maximum generation of $\tau^{\max} = 200$. We consider that the problem remains fixed for τ_T generations and thereafter the parameter t changes by an amount $2\tau_T/(\tau^{\max} - \tau_T)$ (thereby making $\alpha = 1$). Thus, the above problem simulates the following scenario. The time parameter t changes within

$[0, 2]$, independent to the value of τ_T. If τ_T is large, the problem changes less frequently but the amount of change is large. Since a large number of iterations are allowed, an optimization procedure may not have difficulties in tracking the new optimal front. On the other hand, if τ_T is small, the problem changes frequently, but the amount of change is small. It would then be interesting to find a critical τ_t below which an algorithm will not perform well due to the availability of too few generations in tracking the new frontier.

First, we study the effect of frequency of change ($\tau_t = 50$, 25, 20, 10, and 5) on problem FDA2. We fix $\zeta = 0.2$. At a particular τ_t value, the performance will degrade so much that the optimization procedure will not be able to track the Pareto-optimal frontier. NSGA-II parameters used in this study are as follows: Population size is 100, SBX crossover probability is 0.9, polynomial mutation probability is $1/n$ (where n is the number of variables), and distribution indices for crossover and mutation are 10 and 20, respectively. To illustrate the deterioration of fronts for two cases of $\tau_T = 20$ and 10, we have plotted all 200/20 or 10 and 200/10 or 20 fronts obtained using DNSGA-II-A (shown with circles) against the true Pareto-optimal fronts (shown in solid lines) in Figures 2 and 3, respectively. It is somewhat clear from these two figures that the fronts close to the middle of the time period (when there is a comparatively larger shift in the front), a change in every 10 generation is not adequate.

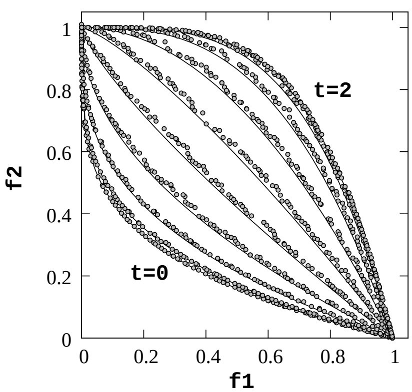

Fig. 2. Obtained fronts against theoretical fronts with $\tau_T = 20$ in FDA2.

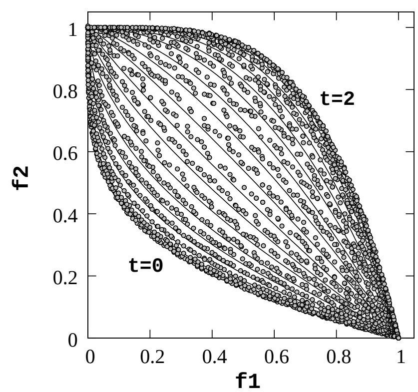

Fig. 3. Obtained fronts against theoretical fronts with $\tau_T = 10$ in FDA2.

To perform this study, we consider the performance index to be the ratio of hypervolumes of achieved and true trade-off fronts (obtained mathematically) with respect to fixed reference points. Figure 4 shows the average ratio of hypervolumes of different τ_T values with time (generation). It is observed that with a more frequent change in the problem, the performance deteriorates. If a hypervolume ratio smaller than 94% (say) is considered to be a threshold for indicating a poor performance, then a change more frequent than $\tau_t = 20$ is considered to produce poor performance by the DNSGA-II-A procedure with a 20% change in population by random solutions.

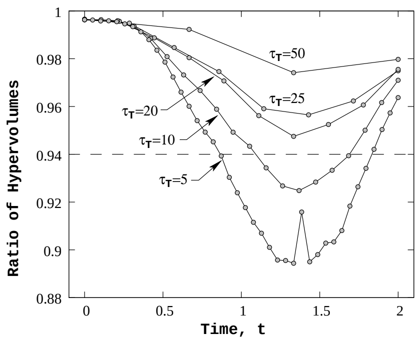

Fig. 4. DNSGA-II-A results on FDA2 ($\zeta = 0.2$).

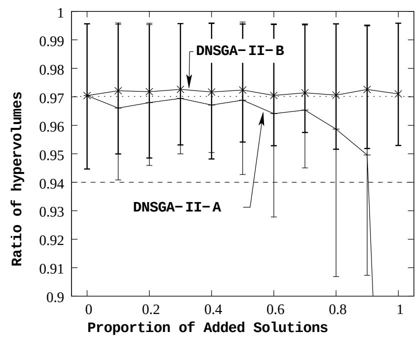

Fig. 5. Effect of varying ζ in DNSGA-II-A and DNSGA-II-B in FDA2 ($\tau_T = 20$).

Next, we perform a parametric study of varying ζ on the FDA2 problem with a variation of the problem after every $\tau_T = 20$ generations. Figure 4 shows the variation of the ratio of obtained hypervolume to the exact hypervolume with ζ using both DNSGA-II-A and DNSGA-II-B. The figure shows that with an introduction of more random solutions, the performance of DNSGA-II-A (random solution addition) deteriorates. With 20 generations to track the new optimal frontier, the task becomes difficult with the introduction of more random solutions in the existing population. Next, we study the effect of adding mutated solutions by using DNSGA-II-B on FDA2. The mutation probability is doubled and the distribution index is reduced to $\eta_m = 4$ to make a significant change in some variables in an existing solution. Interestingly, Figure 4 shows that the performance deteriorates slightly with an increase in addition of mutated solutions, but DNSGA-II-B performs much better than DNSGA-II-A. The addition of a limited proportion of new mutated or random solutions seems to perform better than not adding any new solution at all. With DNSGA-II-B procedure, almost any proportion of addition of mutated solution produce better performance of the algorithm, whereas with DNSGA-II-A, about 20-40% addition of random solutions is better on multiple runs. With this background, we are now ready to apply dynamic NSGA-II procedures to hydro-thermal power scheduling problems.

5 A Case Study: Hydro-Thermal Power Scheduling

In a hydro-thermal power generation systems, both the hydroelectric and thermal generating units are utilized to meet the total power demand. The optimum power scheduling problem involves the allocation of power to all concerned units, so that the total fuel cost of thermal generation and emission properties are minimized, while satisfying all constraints in the hydraulic and power system networks [14]. To solve the hydro-thermal scheduling problem, many different conventional such as Newton's method [16], Lagrange multiplier method [11], dynamic programming [15] and soft computing methodologies such as genetic algorithms [9], evolutionary programming [12], simulated annealing [13] etc. have been tried to solve the single-objective optimization problem. The problem is dynamic due to the changing nature of power demand with time. Thus, ideally the optimal power scheduling problem is truly a on-line dynamic optimization problem in which solutions must be found as and when there is a change in the power demand. In such situations, what can be expected of an optimization algorithm is that it tracks the new optimal solutions as quickly as possible, whenever there is a change.

To understand the insights about the complexity of the problem, at first, we formulate and solve the *stationary* problem using NSGA-II by converting it as an off-line optimization problem. This also facilitates us to compare NSGA-II with a simulated annealing based procedure exist in the literature on the same stationary problem [1]. Gaining the confidence on NSGA-II's ability to solve the constrained problem, we then consider a dynamic version of the problem and solve using the proposed dynamic NSGA-II procedures.

5.1 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_T so that there are a total of $M = T/t_T$ 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. In Section 6, we shall consider the problem as a dynamic optimization problem.

Let us also assume that the system consists of N_h number of hydroelectric (P_{ht}) and N_s number of thermal (P_{st}) generating units sharing the total power demand, such that $\mathbf{x} = (P_{ht}, P_{st})$. The bi-objective optimization problem is given as follows:

$$\begin{aligned}
&\text{Minimize } f_1(\mathbf{x}) = \textstyle\sum_{t=1}^{M}\sum_{s=1}^{N_s} t_T[a_s + b_s P_{st} + c_s P_{st}^2 + |d_s \sin\{e_s(P_s^{min} - P_{st})\}|],\\
&\text{Minimize } f_2(\mathbf{x}) = \sum_{t=1}^{M}\sum_{s=1}^{N_s} t_T[\alpha_s + \beta_s P_{st} + \gamma_s P_{st}^2 + \eta_s \exp(\delta_s P_{st})],\\
&\text{subject to } \textstyle\sum_{s=1}^{N_s} P_{st} + \sum_{h=1}^{N_h} P_{ht} - P_{Dt} - P_{Lt} = 0, \quad t = 1, 2 \ldots, M,\\
&\qquad t_T(a_{0h} + a_{1h}P_{ht} + a_{2h}P_{ht}^2) - W_h = 0, \quad h = 1, 2, \ldots, N_h,\\
&\qquad P_s^{min} \le P_{st} \le P_s^{max}, \quad s = 1, 2, \ldots, N_s, t = 1, 2, \ldots, M,\\
&\qquad P_h^{min} \le P_{ht} \le P_h^{max}, \quad h = 1, 2, \ldots, N_h, t = 1, 2, \ldots, M.
\end{aligned} \tag{2}$$

The transmission loss P_{Lt} term at the t-th interval is given as follows:

$$P_{Lt} = \sum_{i=1}^{N_h+N_s} \sum_{j=1}^{N_h+N_s} P_{it} B_{ij} P_{jt}. \quad (3)$$

This constraint involves both thermal and hydroelectric power generation units. Four power demand values of 900, 1,100, 1,000 and 1,300 MW are considered for the four time periods, respectively. All parameters mentioned in the above formulation are presented in the appendix. The water availability constraint (second set of constraints) requires hydroelectric unit values from different time intervals and makes a dynamic optimization task difficult. We shall discuss about this difficulty more in Section 6. In the present context of solving the problem as an off-line optimization problem, such a dependency is not a matter.

Thus, the bi-objective 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 specific stationary case considered here involves only four ($M = 4$) changes in demand over $T = 48$ hours having a time window of statis of $t_T = 12$ hours. The corresponding problem has six (two hydroelectric ($N_h = 2$) and four thermal ($N_s = 4$)) power units. For the above data, the optimization problem has 24 variables, two objectives, six equality constraints, and 48 variable bounds.

Handling quadratic equality constraints: First, we consider the water availability constraints. Each equality constraint (for a hydroelectric unit h) can be used to replace one of the M power generation values ($P_{h\mu}$) by finding the roots of the quadratic equation and by fixing other P_{ht} as they are in the GA solution:

$$P_{h\mu}^2 + \frac{a_{1h}}{a_{2h}} P_{h\mu} + \frac{1}{t_\mu a_{2h}} \left(-W_h + a_{0h}T + \sum_{\substack{t=1 \\ m\neq\mu}}^{M} t_T a_{1h} P_{ht} + \sum_{\substack{t=1 \\ m\neq\mu}}^{M} t_T a_{2h} P_{ht}^2 \right) = 0. \quad (4)$$

Since $\frac{a_{1h}}{a_{2h}}$ is always positive, only one root can be positive and we accept this root as $P_{h\mu}$. To maintain the structure of the solution, we maintain the ratio of M different P_{ht} values, as they are in a NSGA-II solution. That is, if the original value of μ-th hydroelectric unit was $P_{h\bar{\mu}}$, other units are replaced as follows: $P_{ht} \leftarrow (P_{h\mu}/P_{h\bar{\mu}})P_{ht}$ for $t = 1, 2, \ldots, M$ and $t \neq \mu$. If the above repair mechanism for all N_h hydroelectric units is not successful, we declare the GA solution as infeasible and no further consideration of power balance constraints nor the computation of objective functions are performed. Recall that NSGA-II employs a constraint handling which does not require objective values for infeasible solutions, thereby suiting the above procedure.

We follow a similar procedure as above for handling the power balance constraint and repair a particular thermal unit $P_{\psi t}$ of four thermal units for each time slot. The quadratic equation for this variable can be written as follows:

$$B_{\psi\psi} P_{\psi m}^2 + (2\sum_{j=1}^{n-1} B_{\psi j} P_{jt} - 1) P_{\psi m} + (P_{Dt} + \sum_{i=1}^{n-1}\sum_{j=1}^{n-1} P_{it} B_{ij} P_{jt} - \sum_{i=1}^{n-1} B_{\psi i} P_{it}) = 0, \quad (5)$$

where $n = N_h + N_s$. Since the hydroelectric power units (P_{ht}) are available, the above equation can be solved for $P_{\psi t}$. If this particular value comes within the variable bounds, then the variable is accepted and we go for next constraint involving P_{st} of the next time period. Otherwise, another root-finding equation is tried for the next thermal unit. If for a time period, none of the N_s thermal units resulted in a successful replacement, a penalty is computed and the solution is declared infeasible.

5.2 Simulation Results on the Stationary Problem

NSGA-II is combined with the above-discussed constraint handling method for solving the hydro-thermal scheduling problem. Here, we only consider four changes in the problem in the entire period of 48 hours. Thus, the off-line optimization problem has two objectives, $4(2+4)$ or 24 variables, and six constraints. NSGA-II parameters used in this study are as follows: Population size = 240, Number of generations = 2,000, Crossover probability = 0.9, Mutation probability = 0.04, Distribution indices for crossover and mutation = 10 and 20, respectively. To validate the obtained NSGA-II front, we employ a single-

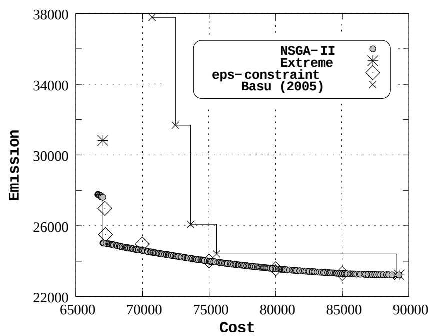

Fig. 6. Pareto-optimal front obtained by NSGA-II, verified by single-objective methods, and by a previous study.

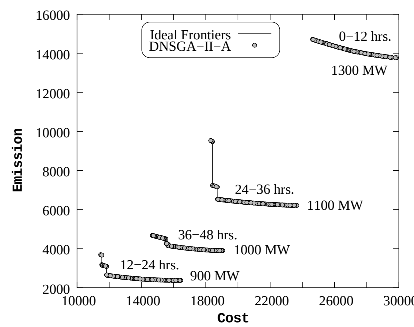

Fig. 7. Four fronts, each change in demand, obtained using DNSGA-II-A with $\zeta = 0.2$.

objective GA and solve several ϵ-constraint problems [8] by fixing f_1 value at different levels. These points are shown in Figure 6 and it is observed that all these points more or less match with those obtained by NSGA-II. Each objective is also optimized independently by a GA and two solutions obtained are plotted in the same figure. One of the extreme points (minimum f_1) is dominated by a NSGA-II solution and the minimum emission solution is matched by a NSGA-II solution. These multiple optimization procedures give us confidence about the optimality of the obtained NSGA-II frontier.

Basu [1] used a simulated annealing (SA) procedure to solve the same problem. That study used a naive penalty function approach in which if any SA solution if found infeasible, it is simply penalized. For different weight vectors scalarizing both objectives, the study presented a set of optimized solutions. A comparison of these results with our NSGA-II approach (in Figure 6) reveals that the front obtained by NSGA-II *dominate* that obtained the previous study. This is mainly due to the use of a better constraint handling strategy in our approach. These results give us confidence in our approach of handling constraints and using NSGA-II for the bi-objective hydro-thermal power dispatch problem. Now, we apply the two proposed dynamic NSGA-II methodologies to the dynamic version of the problem.

6 Dynamic Hydro-Thermal Power Scheduling Problem

The dynamic version of the problem involves more frequent changes in the demand P_{Dt}. To make the demand varying in a continuous manner, we make a piece-wise linear interpolation of power demand values with the following (t, P_{dm}) values: (0, 1,300), (12, 900), (24, 1,100), (36, 1,000) and (48, 1,300) in (Hrs, MW). We keep the overall time window of $T = 48$ hours, but increase the frequency of changes (that is, increase M from four to 192, so that the time window t_T for each demand level varies from 12 hours to 48/192 hours or 15 minutes. It will then be an interesting task to find the smallest time window of statis which a specific multi-objective optimization algorithm can solve successfully. We run the dynamic NSGA-II procedures for $960/M$ (M is the number of changes in the problem) generations for each change in the problem.

Equation 2 requires hydroelectric power generation units from different time intervals to be used together to satisfy the equation. In an dynamic optimization problem, this is a difficulty, as this means that an information about all hydroelectric units are needed right in the first generation. This constraint equates the total required water head to be identical to the available value for each hydroelectric system. In this study, we use a simple principle of allocating an identical water head W_h/M for each time interval.

6.1 Simulation Results

We apply the two dynamic NSGA-II procedures (DNSGA-II-A and DNSGA-II-B) discussed above to solve the dynamic optimization problem. The parameters used are the same as in the off-line optimization case presented before. To compare the dynamic NSGA-II procedures, we first treat each problem as a static optimization problem and apply the original NSGA-II procedure [3] for a large number (500) of generations so that no further improvement is likely. We call these fronts as ideal fronts and compute the hypervolume measure using a reference point which is the nadir point of the ideal front. Thereafter, we apply each dynamic NSGA-II and find an optimized non-dominated front. Then for each front, we compute the hypervolume using the same reference point and then compute the ratio of this hypervolume value with that of the ideal front. This way, the maximum value of the ratio of hypervolume for an algorithm is one and as the ratio becomes smaller than one, the performance of the algorithm gets poorer. First, we consider the problem in which we consider a change after every 12 hours ($M = 4$). Figure 7 shows the four Pareto-optimal fronts obtained using DNSGA-II-A with 20% addition of random solutions every time there is a change in the problem. The DNSGA-II-A procedure is able to find a set of solutions very close to the ideal frontiers in all four time periods. The figure makes one aspect clear. As the demand is more, the power production demands larger cost and emission values.

Increasing number of changes in the problem: Figures 8 to 11 show the hypervolume ratio for different number of changes ($\tau_T = 4$ to 192) in the problem with different proportion of addition of random solutions, ζ, using DNSGA-II-A. The figures also mark the 50th, 90th, 95th and 99th percentile of hypervolume

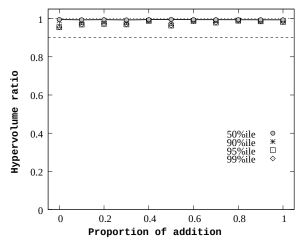

Fig. 8. 3-hourly ($M = 16$) change with DNSGA-II-A.

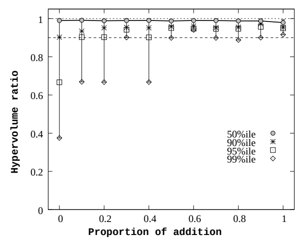

Fig. 9. 1-hourly ($M = 48$) change with DNSGA-II-A.

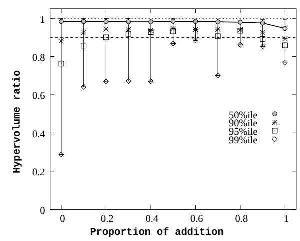

Fig. 10. 30-min. ($M = 96$) change with DNSGA-II-A.

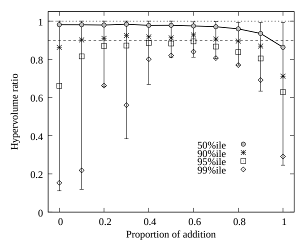

Fig. 11. 15-min.($M = 192$) change with DNSGA-II-A.

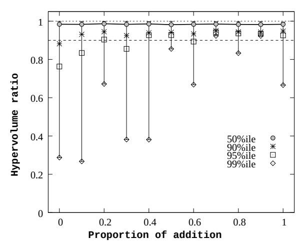

Fig. 12. 30-min. ($M = 96$) change with DNSGA-II-B.

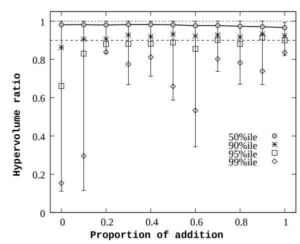

Fig. 13. 15-min. ($M = 192$) change with DNSGA-II-B.

ratio, meaning the cut-off hypervolume ratio which is obtained by the best 50, 90, 95, and 99 percent of M frontiers in a problem with M changes. Figures reveal that as M increases, the performance of the algorithm gets poorer due to the fact that a smaller number of generations ($960/M$) was allowed to meet the time constraint. If a 90% hypervolume ratio is assumed to be the minimum required hypervolume ratio for a reasonable performance of an algorithm and if we consider 95 percentile performance is adequate, the figures show that we can allow a maximum of 96 changes (with a 30-min. change) in the problem. For this case, about 20 to 70% random solutions can be added whenever there is a change in the problem to start the next optimization. Too low addition does not introduce much diversity to start the new problem and too large addition of random solutions destroys the population structure which would have helped for the new problem. The wide range of addition for a successful run suggests the robustness of the DNSGA-II procedure for this problem. Next, we consider DNSGA-II-B procedure in which mutated solutions are added instead of random solutions. Mutations are performed with double the mutation probability and with a $\eta_m = 2$. Figures 12 to 13 show the performance plots for two M values. Here, the effect is somewhat different. In general, with an increase in addition of mutated solutions, the performance is better, as mutations perturb existing solutions locally, thereby helping to introduce adequate diversity needed for the next problem. Once again, 96 changes in the problem in 48 hours seem to be the largest number of changes allowed for the algorithm to perform reasonably well. However, addition of mutated solutions over $\zeta = 40\%$ of the population seems to perform well. Once again, DNSGA-II-B procedure is also found to work well with a wide variety of ζ values.

7 Decision Making in Dynamic EMO

One of the issues which is not discussed enough in the EMO literature is the decision-making aspect after a set of trade-off solutions are found. Some studies in this direction for stationary problems have been just begun [4] and more such studies are called for. In dynamic multi-objective optimization problem, there is an additional problem with the decision-making task. A solution is to be chosen and implemented as quickly as the trade-off frontier is found, and in most situations before the next change in the problem has taken place. This definitely calls for an automatic procedure for decision-making with some pre-specified utility function or some other procedure. In this paper, we choose a utility measure which is related to the relative importance given to both cost and emission objectives. First, we consider a case in which equal importance to both cost and emission are given. As soon as a frontier is found for the forthcoming time period, we compute the pseudo-weight w_1 (for cost objective) for every solution $\mathbf{x}$ using the following term:

$$w_1(\mathbf{x}) = \frac{(f_1^{\max} - f_1(\mathbf{x}))/(f_1^{\max} - f_1^{\min})}{(f_1^{\max} - f_1(\mathbf{x}))/(f_1^{\max} - f_1^{\min}) + (f_2^{\max} - f_2(\mathbf{x}))/(f_2^{\max} - f_2^{\min})}. \quad (6)$$

Thereafter, we choose the solution with $w_1(\mathbf{x})$ closest to 0.5.

To demonstrate the utility of this dynamic decision-making procedure, we consider the hydro-thermal problem with 48 time periods (meaning an hourly change in the problem). Figure 14 shows the obtained frontiers in solid lines and the corresponding preferred (operating) solution with a circle. It can be observed that due to the preferred importance of 50-50% to cost and emission, the solution comes nearly in the middle of each frontier. To meet the water availability constraint, the hydroelectric units of $T_{h1} = 219.76$ MW and $T_{h2} =$

398.11 MW are computed and kept constant over time. However, four thermal power units must produce power to meet the remaining demand and these values for all 48 time periods are shown in Figure 15. The changing pattern in overall

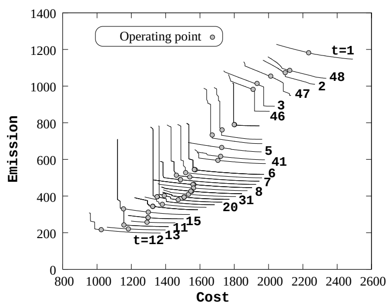

Fig. 14. Operating solution for 50-50% cost-emission case.

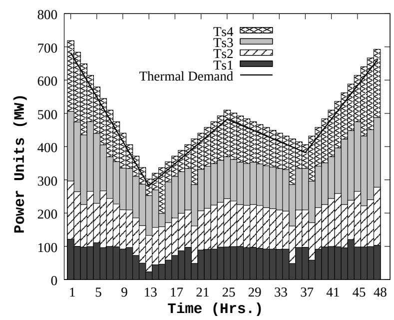

Fig. 15. Variation of thermal power production for 50-50% cost-emission case.

computation of thermal power varies similar to that in the remaining demand in power. The figure also shows a slight over-generation of power to meet the loss term P_{Lt} given in equation 3.

Next, we compare the above operating schedule of power generation with two other extreme cases: (i) 100-0% importance to cost and emission and (ii) 0-100% importance to cost and emission. Figure 16 shows the variation of cost for all the three cases. First, the optimal cost values fluctuate the way the power demand varies. Second, the case with 100% importance to cost requires minimum cost, but causes large emission values and the case with 100% importance to emission causes minimum emission values, but with large costs. A comparison of overall cost and emission values for the entire 48-hour operation for these three cases is summarized in the following table which demonstrates this fact.

Case	Cost	Emission
50-50%	74239.07	25314.44
100-0%	69354.73	27689.08
0-100%	87196.50	23916.09

8 Conclusions

In this paper, we have suggested and demonstrated the solution of a dynamic multi-objective optimization task in a systematic manner. Although the procedure can be used on-line, the current

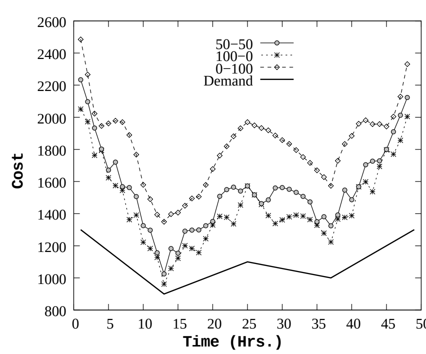

Fig. 16. Variation of operating cost with time for $M = 48$ (1-hourly change).

implementation assumes that the problem remains unchanged for a time period (statis) and the optimization algorithm is run for an initial fraction of the statis and the outcome is used for the remaining period. To restart the EMO procedure (NSGA-II has been used here) for the changed problem, two different strategies are suggested: Introduction of random solutions (DNSGA-II-A) and introduction of mutated solutions (DNSGA-II-B). The number of added solutions relative to the population size (ζ) are kept as a parameter for the study. The procedure is tested on a two-objective test problem and to a hydro-thermal power dispatch problem involving both hydro-electric and thermal power generation units with coupled and non-linear equality constraints. The problem is dynamic due to the change of power demand with time. First, the problem has been solved considering it as an off-line optimization problem (with known power demand) and a better Pareto-optimal front than that reported in an earlier study has been found here. Thereafter, the dynamic problem is solved and the effect of discretization (length of statis) on the performance of both dynamic NSGA-II procedures has been elaborated. NSGA-II with addition of random solutions works the best with about 20-70% addition of new solutions, whereas NSGA-II with addition of mutated solutions works the best for 40-100% addition of new solutions. For the 48-hour overall time range of operation, this systematic study has found that allowing at least an every 30-minute change in the problem is better solved by both proposed dynamic NSGA-II procedures. We are currently investigating a true on-line optimization procedure in which the problem is assumed to remain unchanged only during one generation of the dynamic NSGA-II procedure. A mixed addition of random and mutated solutions can also be tried. Nevertheless, this study proposes and demonstrates the working of two viable dynamic EMO procedures for on-line optimization problems and further studies are imminent to test and fine-tune the procedures for their practical use.

Acknowledgements

Authors acknowledge the financial support from STMicroelectronics Pvt. Ltd., Italy, Singapore and India during the course of this study. Authors would also like to thank Ms. Barnali Kar for introducing them to the hydro-thermal power dispatch optimization problem.

References

1. M. Basu. A simulated annealing-based goal-attainment method for economic emission load disptch of fixed head hydrothermal power systems. *Electric Power and Energy Systems*, 27(2):147–153, 2005.

2. Jurgen Branke. *Evolutionary Optimization in Dynamic Environments*. Kluwer Academic Publishers, Dordrecht, 2002.
3. K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan. A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.
4. K. Deb, S. Chaudhuri, and K. Meitinnen. Towards estimating nadir objective vector using evolutionary approaches. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2006)*, pages 643–650. New York: The Association of Computing Machinery (ACM), 2006.
5. M. Farina, K. Deb, and P. Amato. Dynamic multiobjective optimization problems: Test cases, approximations, and applications. *IEEE Transactions on Evolutionary Computation*, 8(5):425–442, 2000.
6. I. Hatzakis and D. Wallace. Dynamic multi-objective optimization with evolutionary algorithms: a forward-looking approach. In *Proceedings of the 8th annual conference on Genetic and evolutionary computation*, pages 1201–1208, 2006.
7. Y. Jin and B. Sendoff. Constructing dynamic optimization test problems using the multiobjective optimization concept. In *Lecture Notes in Computer Science, 3005*, pages 525–536, 2004.
8. K. Miettinen. *Nonlinear Multiobjective Optimization*. Kluwer, Boston, 1999.
9. S. O. Orero and M.R. Irving. A genetic algorithm modeling framework and solution technique for short term optimal hydrothermal scheduling. *IEEE Transactions on Power Systems*, 13(2), 1998.
10. D. Pratihar, K. Deb, and A. Ghosh. Fuzzy-genetic algorithms and time-optimal obstacle-free path generation for mobile robots. *Engineering Optimization*, 32:117–142, 1999.
11. A. H. A. Rashid and K. M. Nor. An efficient method for optimal scheduling of fixed head hydro and thermal plants. *IEEE Transactions PWRS*, 6(2), 1991.
12. N. Sinha, R. Chakraborty, and P.K. Chattopadhyay. Fast evolutionary programming techniques for short-term hydrothermal scheduling. *Electric Power Systems Research*, 66:97–103, 2003.
13. K. P. Wong and Y. W. Wong. Short-term hydrothermal scheduling part i: simulated annealing approach. *IEE Proc-C Gener., Trans., Distrib*, 141(5), 1994.
14. A. J. Wood and B. F. Woolenberg. *Power Generation, Operation and Control*. John-Wiley & Sons, 1986.
15. J. S. Yang and N. Chen. Short-term hydrothermal co-ordination using multipass dynamic programming. *IEEE Transactions PWRS*, 4(3), 1989.
16. M. F. Zaghlool and F. C. Trutt. Efficient methods for optimal scheduling of fixed head hydrothermal power systems. *IEEE Transactions PWRS*, 3(1), 1988.

A Parameters for Hydro-Thermal Problem

The following parameters are taken from a previous study [1].

Hydroelectric system data						
Unit	a_{0h}	a_{1h}	a_{2h}	W_h	P_h^{min}	P_h^{max}
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	b_s	c_s	d_s	e_s	P_s^{min}	P_s^{max}
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	α_s	β_s	γ_s	η_s	δ_s
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

$$, B = \begin{bmatrix} 49 & 14 & 15 & 15 & 20 & 17 \\ 14 & 45 & 16 & 20 & 18 & 15 \\ 15 & 16 & 39 & 10 & 12 & 12 \\ 15 & 20 & 10 & 40 & 14 & 10 \\ 20 & 18 & 12 & 14 & 35 & 11 \\ 17 & 15 & 12 & 10 & 11 & 36 \end{bmatrix} \times 10^{-6}.$$