Controlled Elitist Non-dominated Sorting Genetic Algorithms for Better Convergence

Kalyanmoy Deb and Tushar Goel

Kanpur Genetic Algorithms Laboratory (KanGAL)
Indian Institute of Technology Kanpur
Kanpur, PIN 208 016, India
{deb,tusharg}@iitk.ac.in
http://www.iitk.ac.in/kangal

KanGAL Report No. 200004

Abstract. Preserving elitism is found to be an important matter in the study of evolutionary multi-objective optimization (EMO). Although there exists a number of new elitist algorithms, where elitism is introduced in different ways, the extent of elitism is the important matter. The desired extent of elitism is directly related to the so-called exploitation-exploration issue of an evolutionary algorithm (EA). For a particular recombination and mutation operators, there exists a selection operator with a particular extent of elitism that will cause a smooth working of an EA. In this paper, we suggest an approach where the extent of elitism can be controlled by fixing a user-defined parameter. By applying an elitist multi-objective EA (NSGA-II) to a number of difficult test problems, we show that the NSGA-II with controlled elitism has much better convergence property than the original NSGA-II. The suggested controlled elitism-preserving approach is generic and can also be implemented with other elitist multi-objective evolutionary algorithms.

1 Introduction

It is now well established through a number of studies [11, 1] that elitist multi-objective evolutionary algorithms (MOEAs) have better convergence characteristics than non-elitist MOEAs. Motivated by these studies, researchers and practitioners now concentrate on developing and using elitist MOEAs. This have resulted in a number of elitist MOEAs, such as strength Pareto EA or SPEA [13], Pareto-archived evolution strategy or PAES[8], and others. After the early suggestion of a non-elitist multi-objective genetic algorithm (NSGA) [9], the first author and his students have also suggested an elitist, fast, and parameter-less multi-objective GA (NSGA-II) [2, 4]. All these elitist algorithms are interesting and have shown to have tremendous potential in solving multi-objective optimization problems.

In the context of single-objective optimization, the elitism is introduced in a number of different ways. Preserving elitism means emphasizing the currently-best solutions in the subsequent generations. In some implementations, this is achieved by simply carrying over the best $\epsilon\%$ of the population in the next generation. In other implementations, the parent and child populations are combined together and the best 50% of the population is retained. This way previously-found good solutions are given a chance to carry over to subsequent generations.

In the context of multi-objective optimization, the meaning of elite solutions is different from that in single-objective optimization. Here, all solutions that belong to the currently-best non-dominated front are best solutions in the population and are all equally important. Thus, all these solutions are elite solutions. In many occasions, a population may be mostly comprised of currently-best non-dominated solutions. When this happens, the preservation of elitism means acceptance of all such solutions. In such a scenario, not many new solutions can be accepted in the population. As a result, the search process may stagnate or prematurely converge to a suboptimal solution set. Thus, there is a need of introducing elitism in a controlled manner, in the context of multi-objective optimization.

In this paper, we address the issue of controlling elitism from the point of maintaining a balance between underlying exploitation and exploration issues. Thereafter, we argue that our earlier elitist implementation NSGA-II can have uncontrolled elitism in tackling certain problems. We suggest a controlled elitist approach, where only a certain portion of the population is allowed to contain the currently-best non-dominated solutions. The dominated solutions are purposefully kept in the population to reduce the elitism effect. By introducing a parameter to control the extent of elitism, we study its effect on five complex test problems. In all simulation results, it is clear that the controlled NSGA-II has a better convergence property than the original NSGA-II with uncontrolled elitism.

2 Multi-Objective Evolutionary Algorithms

The details of NSGA-II algorithm appears elsewhere [4]. Essentially, NSGA-II differs from our original NSGA implementation [9] in a number of ways. Firstly, NSGA-II uses an elite-preserving mechanism, thereby assuring preservation of previously found good solutions. Secondly, NSGA-II uses fast non-dominated sorting procedure. Thirdly, NSGA-II does not require any tunable parameter, thereby making the algorithm independent of the user.

Initially, a random parent population P_0 is created. The population is sorted based on the non-domination. A special book-keeping procedure is used in order to reduce the computational complexity down to $O(N^2)$. Each solution is assigned a fitness equal to its non-domination level. Binary tournament selection, recombination, and mutation operators are used to create a child population Q_0 of size N. Thereafter, we use the algorithm shown in Figure 1 in every generation.

First, a combined population $R_t = P_t \cup Q_t$ is formed. This allows parent solutions to be compared with the child population, thereby ensuring elitism. The population R_t is of size $2N$. Then, the population R_t is sorted according to

$R_t = P_t \cup Q_t$
$\mathcal{F} =$ `fast-non-dominated-sort(`R_t`)`
$P_{t+1} = \emptyset$ and $i = 1$
until $|P_{t+1}| + |\mathcal{F}_i| \leq N$
    $P_{t+1} = P_{t+1} \cup \mathcal{F}_i$
    `crowding-distance-assignment(`$\mathcal{F}_i$`)`
    $i = i + 1$
Sort$(\mathcal{F}_i, \prec_n)$
$P_{t+1} = P_{t+1} \cup \mathcal{F}_i[1 : (N - |P_{t+1}|)]$
$Q_{t+1} =$ `make-new-pop(`P_{t+1}`)`
$t = t + 1$

Fig. 1. NSGA-II algorithm is shown.

non-domination [10] and different non-dominated fronts F_1, F_2, and so on are found. The algorithm is illustrated in the following:

The new parent population P_{t+1} is formed by adding solutions from the first front F_1 and continuing to other fronts successively till the size exceeds N. Individuals of each front are used to calculate the crowding distance—the distance between the neighboring solutions [4]. Thereafter, the solutions of the last accepted front are sorted according to a *crowded comparison criterion* and a total of N points are picked. Since the diversity among the solutions is important, the crowded comparison criterion uses a partial order relation $\prec_n$ as follows:

Definition 1 *Solution i is better than solution j in relation $\prec_n$ if $(i_{rank} < j_{rank})$ or $((i_{rank} = j_{rank})$ and $(i_{distance} > j_{distance}))$.*

That is, between two solutions with differing non-domination ranks we prefer the point with the lower rank. Otherwise, if both the points belong to the same front then we prefer the point which is located in a region with lesser number of points (or with larger crowded distance). This way solutions from less dense regions in the search space are given importance in deciding which solutions to choose from R_t. This constructs the population P_{t+1}. This population of size N is now used for selection, crossover and mutation to create a new population Q_{t+1} of size N. We use a binary tournament selection operator but the selection criterion is now based on the crowded comparison operator $\prec_n$. The above procedure is continued for a specified number of generations.

It is clear from the above description that NSGA-II uses (i) a faster non-dominated sorting approach, (ii) an elitist strategy, and (ii) no niching parameter. It has been shown elsewhere [2] that the above procedure has $O(MN^2)$ computational complexity.

2.1 Exploitation Versus Exploration Issue

The above description of NSGA-II raises an important issue relating to EA research: the issue of exploitation and exploration [7]. Let us imagine that at a generation, we have a population R_t where most of the members lie on the non-dominated front of rank one and this front is away from the true Pareto-optimal front. This will happen in the case of multi-modal problems, where a population can get attracted to a local Pareto-optimal front. Since most members belong to the current best non-dominated front, the elitism operation will result in deleting most members belonging to other fronts. In the above NSGA-II algorithm, elitist solutions are emphasized in two occasions. Once in the `make-new-pop`(P_{t+1}) operation and again during the elitism operation in the `until` loop. The former operation involves the crowded tournament selection operator, which emphasizes the elitist solutions (the currently-best non-dominated solutions). In the latter case, solutions are selected starting from the currently-best non-dominated solutions till all population slots are filled. This way, the elitist solutions also get emphasized. This dual emphasis of elitist solutions will cause a rapid deletion of solutions belonging to non-elitist fronts. Although the crowding tournament operator will ensure diversity along the current non-dominated front, lateral diversity will be lost. In many problems, when this happens the search slows down, simply because there may be a lack of diversity in certain decision variables left to push the search towards better regions of optimality. Thus, in order to ensure better convergence, a search algorithm may need diversity in both directions—along the Pareto-optimal front and lateral to the Pareto-optimal front, as shown in Figure 2. In the test problems suggested elsewhere [5], the lateral diversity is

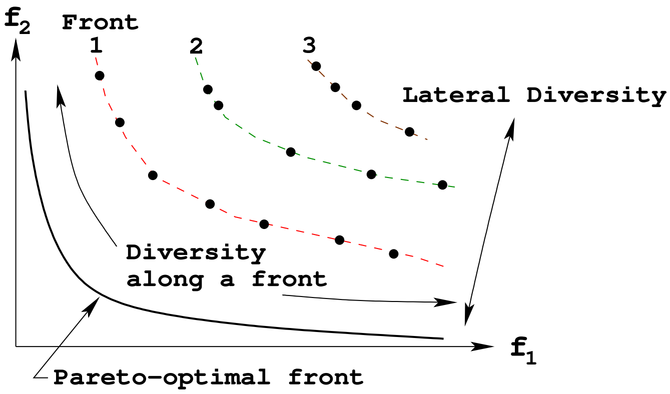

Fig. 2. Controlled elitism procedure is illustrated.

ensured by the functional $g()$ with $(n-1)$ decision variables. Solutions converging to any local Pareto-optimal front will make all these decision variables to take an identical value. The dual elitism will reduce the variability in these solutions and eventually prevent the algorithm to move towards the true Pareto-optimal front.

The difficulty arises because the way elitism is implemented in NSGA-II. There is no control on the exploitation of the current best non-dominated solutions. In the above discussion, it is clear that in certain complex functions, NSGA-II, in the absence of a lateral diversity-preserving operator such as mutation, causes too much exploitation of currently-best non-dominated solutions. In order to counteract this excessive selection pressure, an adequate exploration by means of the search operators must be used. A proper balance of these two issues is not possible to achieve with the uncontrolled elitism mechanism in NSGA-II. In an earlier study [2], we have shown that in the test problem ZDT4 with Rastrigin's multi-modal function as the g functional, NSGA-II could not converge to the global Pareto-optimal front. However, when a mutation operator with a larger mutation strength is used, NSGA-II succeeds in converging to the global Pareto-optimal front. Increasing variability through mutation enhances the exploration power of NSGA-II and a balance between enhanced exploitation of NSGA-II and the modified exploration can be maintained.

Although many researchers has adopted an increased exploration requirement by using a large mutation strength in the context of single-objective EAs, the extent of needed exploration is always problem dependent. In the following section, instead of concentrating on changing the search operators, we suggest a controlled elitism mechanism which will control the extent of exploitation rather than controlling the extent of exploration.

3 Controlled Approach

In the proposed controlled NSGA-II, we restrict the number of individuals in the current best non-dominated front adaptively. We attempt to maintain a predefined distribution of number of individuals in each front. Specifically, we use a geometric distribution for this purpose:

$$n_i = r n_{i-1}, \tag{1}$$

where n_i is the maximum number of allowed individuals in the i-th front and r (< 1) is the reduction rate. Although the parameter r is user-defined, the procedure is adaptive as follows.

First, the population $R_t = P_t \cup Q_t$ is sorted for non-domination. Let us say that the number of non-dominated fronts in the population (of size $2N$) is K. Thus, according to the geometric distribution, the maximum number of individual allowed in the i-th front $(i = 1, 2, \ldots, K)$ in the new population of size N is

$$n_i = N \frac{1-r}{1-r^K} r^{i-1}. \tag{2}$$

Since $r < 1$, the maximum allowable number of individuals in the first front is highest. Thereafter, each front is allowed to have an exponentially reducing number of solutions. This exponential distribution considered above is an assumption and must be tried with other distributions such as an arithmetic distribution or a harmonic distribution. But the principle in all these approaches is the same—forcibly allow solutions from all non-dominated fronts to co-exist in the population.

Although equation 2 denotes the maximum allowable number of individuals n_i in each front i, in a population, there may not exist exactly n_i individuals in front i. We resolve this problem by starting a procedure from the first front. The number of individuals in the first front is counted. Let us say that there are n_1^t individuals. If $n_1^t > n_1$ (that is, there are more solutions than allowed), we only choose n_1 solutions using the crowded tournament selection. This way, exactly n_1 solutions that are residing in less crowded region are selected. On the other hand, if $n_1^t < n_1$ (that is, there are less solutions in the population than allowed), we choose all n_1^t solutions and count the number of remaining slots $\rho_1 = n_1 - n_1^t$. The maximum allowed number of individuals in the second front is now increased to $n_2 \rightarrow n_2 + \rho_1$. Thereafter, the actual number of solutions n_2^t present in the second front is counted and is compared with n_2 as above. This procedure is continued till all N individuals are selected. Figure 3 shows that a population of size $2N$ (having four non-dominated fronts with topmost subpopulation representing front one and so on) is reduced to a new population P_{t+1} of size N using the above procedure. In the transition shown in the right, all four fronts have representative solutions. Besides this elitism procedure the rest of the procedure is kept the same as that in NSGA-II. The figure also shows

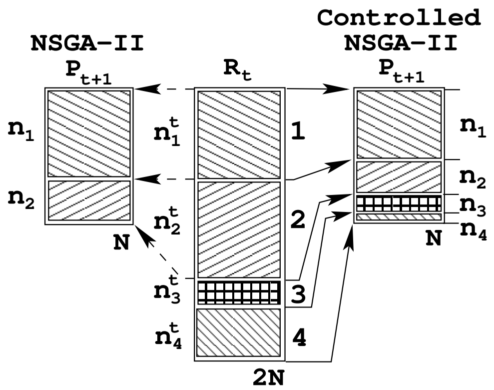

Fig. 3. Controlled elitism procedure is illustrated.

the new population P_{t+1} which would have obtained using the usual NSGA-II procedure. It is clear that the new population under the controlled NSGA-II is more diverse than that under NSGA-II.

Since the population is halved, it is likely that in each front there would be more solutions than allowed. However, there could be some situations where after all $2N$ solutions are processed as above, there are still some slots left in the new population to be filled. This may happen particularly when r is large. In such cases, we make another pass with the left-out individuals from the first front, continuing to other fronts, and start including them till we fill up the remaining slots.

3.1 Discussions

As mentioned earlier, keeping individuals from many non-dominated fronts in the population help the recombination operator to create diverse solutions. NSGA-II and many other successful MOEAs thrive at maintaining diversity among solutions of independent non-dominated fronts. The controlled elitism procedure suggested above will help maintain diversity in solutions across the non-dominated fronts. In solving difficult multi-objective optimization problems, this additional feature may be helpful in progressing towards the true Pareto-optimal front.

It is intuitive that the parameter r is important in maintaining the correct balance between the exploitation and exploration issue discussed earlier. This parameter sets up the extent of exploration allowed in an MOEA. If r is small, the extent of exploration is large and vice versa. In general, the optimal value of r will depend on the problem and will be difficult to find theoretically. In the following section, we present simulation results on a number of difficult problems to investigate if there exists any range of r, where NSGA-II performs well.

4 Simulation Results

In order to demonstrate the working of the proposed methodology, we use a set of difficult test problems. The first four problems have multiple Pareto-optimal fronts, of which one is the globally Pareto-optimal set. Next, we consider one constrained test problem, which has infeasible regions restricting the search towards the true Pareto-optimal front.

In order to investigate the effect of the controlled elitism alone, we do not use the mutation operator. For controlled NSGA-II runs, we do not use the selection operator in `make_new_pop()` to create the child population. For all problems, we use a population size of 100, a crossover probability of 0.95, and spread factor for the SBX operator [6] of 20. All simulation runs are continued till 200 generations are over.

4.1 Multi-objective Rastrigin's Problem

This problem is identical to the ZDT4 function, introduced elsewhere [12]:

$$\mathsf{ZDT4}: \begin{cases} \text{Minimize } f_1(\mathbf{x}) = x_1 \\ \text{Minimize } f_2(\mathbf{x}) = g(\mathbf{x})\left[1 - \sqrt{x_1/g(\mathbf{x})}\right] \\ \qquad\qquad g(\mathbf{x}) = 91 + \sum_{i=2}^{10}\left[x_i^2 - 10\cos(4\pi x_i)\right] \\ \qquad\qquad x_1 \in [0,1], \quad x_i \in [-5,5],\ i = 2,\ldots,10. \end{cases} \tag{3}$$

It was also observed in another study [2] that NSGA-II has difficulty in converging to the global Pareto-optimal front. However, when the mutation strength is increased, NSGA-II could converge to the correct front.

Figure 4 shows that a simulation run with NSGA-II could not converge to the true Pareto-optimal front (the plot shown with diamonds). Since mutation is not

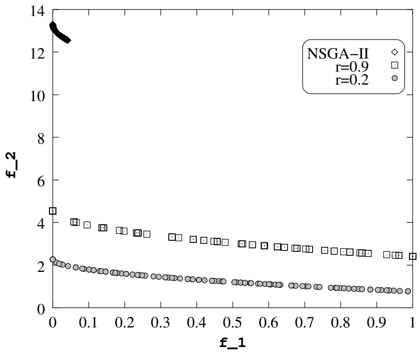

Fig. 4. Distribution of obtained set of solutions for ZDT4.

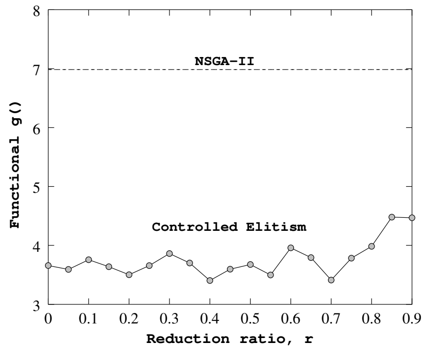

Fig. 5. Average $g()$ for controlled and uncontrolled NSGA-IIs for ZDT4.

used, NSGA-II also faced difficulty in maintaining a diverse set of solutions in the obtained front. The figure also shows two other simulations with controlled elitism ($r = 0.2$ and $r = 0.9$), starting from the same initial population. The figure clearly shows that when controlled elitism is used, existence of multiple fronts ensures better convergence and better diversity among obtained solutions. To make a thorough investigation, we have used 19 controlled elitist NSGA-IIs in the range $r \in [0.05, 0.9]$ with an interval of 0.05 and compared the results with the original NSGA-II. For each NSGA-II, we have performed 25 simulation studies, each starting with a different initial population. But for all NSGA-IIs, 25 identical populations are used. The average value of the best functional $g()$ in 200 generations is calculated and plotted in Figure 5. Equation 3 reveals that all solutions having $g = 1$ lie in the global Pareto-optimal front. Thus, an MOEA with a smaller value of functional $g()$ can be considered to have done well in solving the above problem. The figure clearly shows that all controlled elitism runs have done much better than the original NSGA-II. The average $g()$ value is much closer to one than NSGA-II. Although there is some fluctuations in the obtained $g()$ values for different setting of the reduction rate, all controlled NSGA-IIs are better.

In order to investigate the composition of population with controlled elitism, we count the number of solutions in the first and the second fronts in 25 different NSGA-II and controlled NSGA-II with $r = 0.5$ runs. The average of these numbers is plotted in Figures 6 and 7. Figure 6 shows that in the case of original

NSGA-II, the number of solutions is in the first front grows very rapidly after a certain number of generations. Within about 90 generations, all population members belong to the best non-dominated front. Thereafter, NSGA-II works by keeping all population members in the first front. The only way a new solution will be accepted, if it lies in the current best non-dominated front and if it resides in a less crowded area. On the other hand, with controlled NSGA-II (with $r = 0.5$) the population is never filled with currently-best non-dominated solutions alone. With $r = 0.5$, only about 50% population members belong to the best non-dominated front. Moreover, the rate of increase of solutions in the best non-dominated front is also small and importantly, the growth of solutions in this front begins much later than that with the original NSGA-II. Figure 7

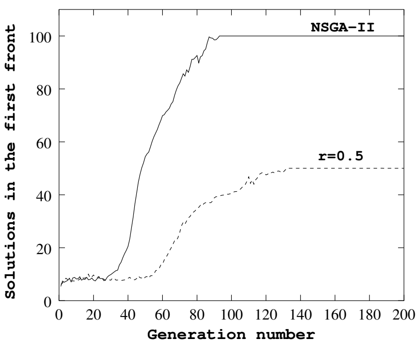

Fig. 6. Average number of solutions in the first (best) non-dominated front.

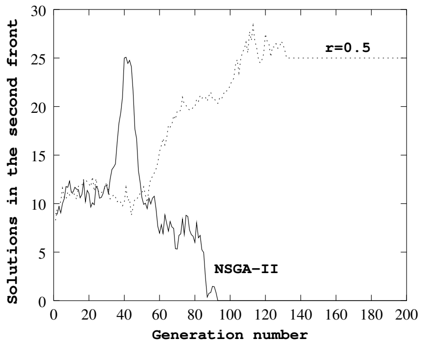

Fig. 7. Average number of solutions in the second non-dominated front.

shows the number of solutions in the second front. The figure clearly shows that in the original NSGA-II, there exists no member in the second front after about 90 generations, meaning that all solutions belong to the first non-dominated front. But, in the case of controlled NSGA-II, about 25% population belongs to the second front. The presence of so many solutions in the second front and subsequent fronts provides a lateral diversity needed for a better convergence to the global Pareto-optimal front.

We would like to distinguish the original NSGA-II algorithm with controlled NSGA-II having $r = 0$. With $r = 0$, the allowed number of solutions in the best non-dominated front is N (the population size). Thus, there is no pressure to maintain other fronts in the population. This algorithm may apparently look the same as the original NSGA-II, but there is a difference. In the $r = 0$ controlled NSGA-II, the tournament selection in the `make_new_pop()` operator is eliminated. This way, the selection pressure for currently-best elitist solutions is smaller than that in the original NSGA-II. Although this simple change in NSGA-II exhibits a better performance, the controlled NSGA-IIs with $r > 0$ are generic. Moreover, NSGA-IIs with some values of $r > 0$ show better performance than NSGA-II with $r = 0$. It is a matter of future research to investigate if a dynamically changing r with generation makes the convergence better than a fixed r.

4.2 Multi-objective Griewangk Problem

Next, we construct a problem using equation 3, except that the functional $g()$ is now different ($g()$ is a Greigwank's function):

$$g(\mathbf{x}) = 2 + \sum_{i=2}^{10} x_i^2/4000 - \Pi_{i=2}^{10} \cos(x_i/\sqrt{i}). \tag{4}$$

Here, the variables x_2 till x_{10} are all allowed to vary in $[-512, 511]$. This function has 163^9 local Pareto-optimal fronts, of which only one is globally Pareto-optimal front having $g(\mathbf{x}) = 1$.

Figure 8 shows a typical performance plot of NSGA-II and controlled elitist NSGA-IIs on this function without mutation. It is clear that controlled elitism

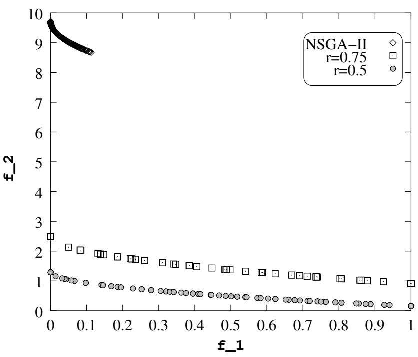

Fig. 8. Distribution of obtained set of solutions for ZDT4 with Griewangk's function.

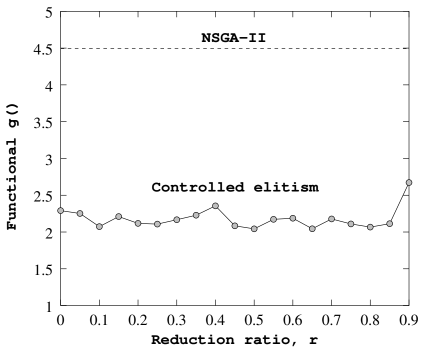

Fig. 9. Average $g()$ for controlled and uncontrolled NSGA-IIs for ZDT4 with Griewangk's function.

runs are better converged near the true Pareto-optimal front. Because of lateral diversity present in the controlled NSGA-II runs, diversity in the obtained set of solutions also comes as a bonus. Figure 9 shows a detailed comparative study. Once again, the figure confirms that the controlled NSGA-II results are better.

4.3 Multi-objective Schwefel's Problem

We replace the Griewangk's function with Schwefel's function:

$$g(\mathbf{x}) = 3771.847 - \sum_{i=2}^{10} x_i \sin(\sqrt{|x_i|}). \tag{5}$$

The variable boundaries are the same as that in the previous problem. Here, the functional $g()$ has 8^9 local Pareto-optimal solutions, of which only one (with $g(\mathbf{x}) = 1$) is globally optimal.

Like before, Figure 10 shows one simulation runs on a f_1-f_2 plot and Figure 11 shows the average $g()$ for 25 independent runs. It is clear from the figures that the

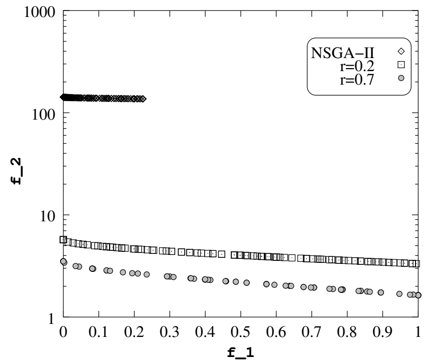

Fig. 10. Distribution of obtained set of solutions for ZDT4 with Schwefel's function.

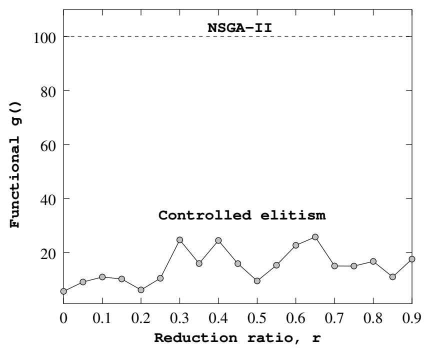

Fig. 11. Average $g()$ for controlled and uncontrolled NSGA-IIs for ZDT4 with Schwefel's function.

original NSGA-II (without mutation) gets attracted to a locally Pareto-optimal front far away from the globally optimal front. However, controlled NSGA-IIs can bring the population much closer to the globally Pareto-optimal front having $g(\mathbf{x}) = 1$).

4.4 Nonuniform distribution

We now use the problem ZDT6, which was used in another study [12].

$$\text{ZDT6}: \begin{cases} \text{Minimize } f_1(\mathbf{x}) = 1 - \exp(-4x_1)\sin^6(6\pi x_1) \\ \text{Minimize } f_2(\mathbf{x}) = g(\mathbf{x})\left[1 - (x_1/g(\mathbf{x}))^2\right] \\ \qquad g(\mathbf{x}) = 1 + 9\left[(\sum_{i=2}^{10} x_i)/9\right]^{0.25} \\ \qquad x_1 \in [0,1], \quad x_i \in [-5,5], \ i = 2, \ldots, 10. \end{cases} \tag{6}$$

The above problem should not cause much difficulty in converging to the correct front ($g(\mathbf{x}) = 1$), but causes difficulty in maintaining a uniform spread of solutions in the front.

Even in this problem, we observe from Figures 12 and 13 that the original NSGA-II without the help of mutation cannot come closer to the Pareto-optimal front. Moreover, the distribution of solutions in the obtained set is also non-uniform. Controlled NSGA-IIs performs better than the original NSGA-II in both converging near the true Pareto-optimal front and maintaining a uniform-like distribution. Figure 12 shows that NSGA-II's crowding mechanism is good enough to maintain a good distribution of solutions in this problem. But, NSGA-II's large selection pressure through uncontrolled elitism is detrimental. Once again, the performance of controlled NSGA-IIs is more-or-less insensitive to the chosen r value.

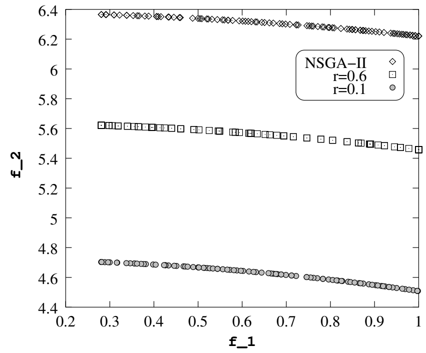

Fig. 12. Distribution of obtained set of solutions for ZDT6.

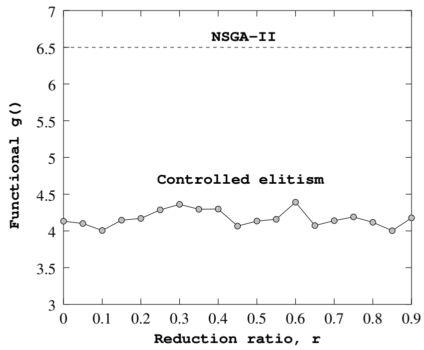

Fig. 13. Average $g()$ for controlled and uncontrolled NSGA-IIs for ZDT6.

4.5 Constrained Test Problem (CTP7)

Finally, we attempt to solve a constrained test problem shown below [3]:

$$\text{CTP7}: \begin{cases} \text{Minimize } f_1(\mathbf{x}) = x_1 \\ \text{Minimize } f_2(\mathbf{x}) = g(\mathbf{x})\left(1 - \frac{f_1(\mathbf{x})}{g(\mathbf{x})}\right) \\ \text{Subject to } c(\mathbf{x}) \equiv \cos(\theta)(f_2(\mathbf{x}) - e) - \sin(\theta) f_1(\mathbf{x}) \geq \\ \qquad a\left|\sin\left(b\pi\left(\sin(\theta)(f_2(\mathbf{x}) - e) + \cos(\theta) f_1(\mathbf{x})\right)^c\right)\right|^d . \end{cases} \tag{7}$$

The decision variable x_1 is restricted in $[0, 1]$. The functional $g()$ is the Rastrigin's function used in equation 3. But, here we use five variables, instead of 10 used in ZDT4. Other parameter values are as follows:

$$\theta = -0.05\pi, \quad a = 40, \quad b = 5, \quad c = 1, \quad d = 6, \quad e = 0.$$

The feasible search space and the corresponding disconnected Pareto-optimal regions are shown in Figure 14.

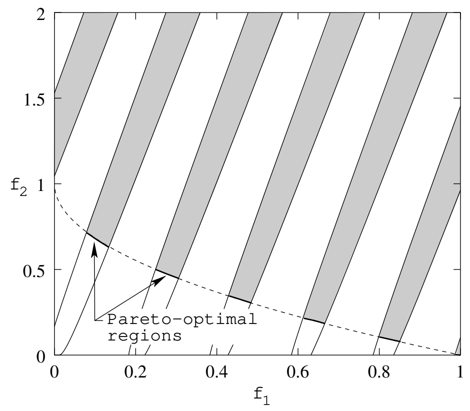

Fig. 14. Constrained test problem CTP7.

In the absence of mutation operator, we observe that NSGA-II's strong elitism prevents the algorithm to come closer to the true Pareto-optimal front (Figure 15). The outcome of 25 runs suggests that the controlled elitism can overcome this problem somewhat and help NSGA-II to converge closer to the true Pareto-optimal front (Figure 16). The absence of mutation in all NSGA-II runs prohibits them to maintain a diverse set of population members in the obtained set.

5 Conclusion

In this paper, we have discussed the need to make a balance between exploitation offered by a selection operator along with an elite-preserving mechanism with exploration offered by a recombination operator. Although NSGA-II has been largely successful in many problems, in some difficult problems, it was observed to have convergence problems. Here, we have clearly shown that the uncontrolled elitism present in NSGA-II produces a large selection pressure for currently-best non-dominated solutions. In difficult multi-modal problems and other problems, this forces NSGA-II without any mutation operator to have a premature convergence to a suboptimal front.

By forcing a number of non-dominated fronts to co-exist in an NSGA-II, we have shown that the selection pressure offered by NSGA-II's elitism operator can be controlled. On five difficult test problems, including a difficult constraint-handling problem, we have shown that the controlled NSGA-II has a better

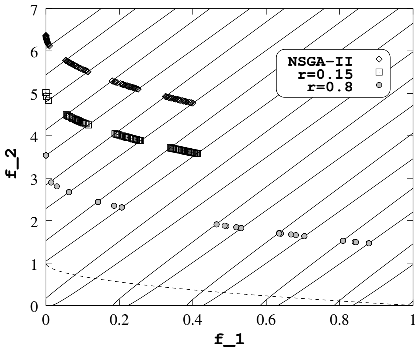

Fig. 15. Distribution of obtained set of solutions for CTP7.

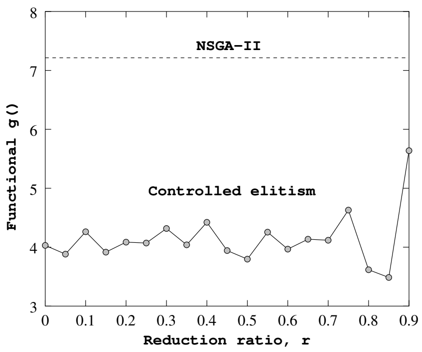

Fig. 16. Average $g()$ for controlled and uncontrolled NSGA-IIs for CTP7.

convergence property than the original NSGA-II. Since diverse non-dominated fronts are also maintained in the population, controlled NSGA-II also produces a better distribution of solutions than the original NSGA-II.

Although controlled NSGA-IIs with any r value has performed well on most problems, there seems to be not much effect of the parameter r. Ideally, we would expect that the performance would improve with increasing r. This happens because of the increased importance to maintain lateral diversity. After a critical r, the performance is expected to go down. This happens because of the lack of effect of elitism. One of the reasons why we have not observed such a behavior in the simulations of this study is that we are dealing with problems having a continuous search space. We observed that the solutions of the first and second front are so close to each other that their fronts do not seem to be separated from each other visually. Thus, the crossover operator does not exploit the advantage of maintaining more than one front in the population. However, in discrete search space problems, the expected difference between two fronts may be significant and crossover can take advantage of this diversity to improve convergence to the true front. For continuous search space problems, the true lateral diversity can be achieved by forcing solutions of distant fronts to coexist in the population. For this purpose, fronts with a critical distance among them can be emphasized by using the geometric distribution.

The controlled elitism mechanism implemented here with NSGA-II is generic and can also be implemented with other elitist MOEAs. It would be a matter of future research to investigate whether the convergence property of other MOEAs improves with such a controlled elitism operator.

Acknowledgements

Authors acknowledge the support provided by Department of Science and Technology (DST) for this study.

References

1. Corne, D. W., Knowles, J. D., and Oates, M. J. (2000). The Pareto envelope-based selection algorithm for multiobjective optimization. *Proceedings of the Parallel Problem Solving from Nature VI Conference*, pp. 839–848.
2. Deb, K., Pratap, A., Agrawal, S. and Meyarivan, T. (2000). A fast and elitist multi-objective genetic algorithm: NSGA-II. Technical Report No. 2000001. Kanpur: Indian Institute of Technology Kanpur, India.
3. Deb, K., Mathur, A. P., Meyarivan, T. (2000). Constrained test problems for multi-objective evolutionary optimization. Technical Report No. 200002. Kanpur: Kanpur Genetic Algorithms Laboratory, IIT Kanpur, India.
4. Deb, K., Agrawal, S., Pratap, A., Meyarivan, T. (2000). A Fast Elitist Non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. *Proceedings of the Parallel Problem Solving from Nature VI Conference*, pp. 849–858.
5. Deb, K. (1999) Multi-objective genetic algorithms: Problem difficulties and construction of test Functions. *Evolutionary Computation, 7*(3), 205–230.
6. Deb, K. and Agrawal, R. B. (1995) Simulated binary crossover for continuous search space. *Complex Systems, 9* 115–148.
7. Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning*. Reading: Addison-Wesley.
8. Knowles, J. and Corne, D. (1999) The Pareto archived evolution strategy: A new baseline algorithm for multiobjective optimisation. *Proceedings of the 1999 Congress on Evolutionary Computation*, Piscataway: New Jersey: IEEE Service Center, 98–105.
9. Srinivas, N. and Deb, K. (1995). Multi-Objective function optimization using non-dominated sorting genetic algorithms. *Evolutionary Computation*(2), 221–248.
10. Steuer, R. E. (1986). *Multiple criteria optimization: Theory, computation, and application*. New York: Wiley.
11. Zitzler, E. (1999). Evolutionary algorithms for multiobjective optimization: Methods and applications. Doctoral thesis ETH NO. 13398, Zurich: Swiss Federal Institute of Technology (ETH), Aachen, Germany: Shaker Verlag.
12. Zitzler, E., Deb, K., and Thiele, L. (2000). Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary Computation Journal*, 8(2), 173–196.
13. Zitzler, E. and Thiele, L. (1998). An evolutionary algorithm for multiobjective optimization: The strength Pareto approach. *Technical Report No. 43 (May 1998).* Zürich: Computer Engineering and Networks Laboratory, Switzerland.