Effect of Selection Operator on NSGA-III in Single, Multi, and Many-Objective Optimization

Haitham Seada and Kalyanmoy Deb
Computational Optimization and Innovation (COIN) Laboratory
Department of Computer Science and Engineering
Michigan State University, East Lansing, MI 48824, USA
Email: {seadahai,kdeb}@msu.edu

COIN Report Number 2015002

Abstract—**Decomposition-based elitist non-dominated sorting genetic algorithm (NSGA-III) is a recently proposed many-objective optimization algorithm that uses multiple pre-defined yet adaptable reference directions to maintain diversity among its solutions. Designing to solve specifically many-objective problems having four or more objectives, the authors of NSGA-III restricted the population size to be equal to the number of chosen reference directions. This restriction hinders the usage of NSGA-III to single-objective optimization problems, where, by definition, there is only one reference direction. For this reason, a unified algorithm – U-NSGA-III – has been recently proposed to handle this issue. U-NSGA-III is capable of adapting automatically to the dimensionality of the problem in hand through its niching based selection operator. However, the authors of U-NSGA-III abided by this single-fold restriction in all NSGA-III simulations of their study. In this paper we test the possibility of ignoring this restriction of NSGA-III and use multiple population folds to solve single, multi and many-objective problems. Simulations are performed on a variety of constrained and unconstrained single, multi and many-objective problems for this purpose. The strengths and weaknesses of multi-fold NSGA-III compared to those of U-NSGA-III are thoroughly investigated here. The robustness of NSGA-III in each type of problems is also discussed. This study provides a more comprehensive evaluation of the original NSGA-III procedure, which seems to have a wider scope than the original study had foreseen.**

I. Introduction

Interest in solving many-objective optimization algorithms to solve four or more objectives has been growing over the last ten years. Researcher made significant leaps from designing algorithms to deal with only bi-objective and tri-objective problems, like NSGA-II [8], SPEA2 [19] and PESA [4] since early nineties to algorithms capable of handling up to 10 or even 15 objectives, like MOEA/D [17], BORG [12] and most recently NSGA-III [9], [13] among others [6], [3]. In all these studies, most of the emphasis has been focused on experimenting with different dominance principle [15] and/or diversity management techniques [1] to maintain a balance between convergence and diversity of population members. Some researchers were also interested in designing operators or combining existing operators to enhance the performance of their algorithms [12]. However, *selection* operator remained mostly an invariant part in majority of these algorithms. The effect of selection as the dimensionality of the problem increases has not been thoroughly investigated. In this study, we use NSGA-III and U-NSGA-III [16] to investigate the effect of selection in single, multi, and many-objective scenarios.

NSGA-III was proposed in early 2014, to solve both unconstrained [9] and constrained [13] many-objective optimization problems. NSGA-III uses an external guidance mechanism to maintain diversity among its solutions [5]. Throughout the optimization process, each feasible solution is attempted to adhere to one reference direction out of a set of pre-defined reference directions (niching). These directions can be set either uniformly in the absence of any preference information among objectives, or preferentially to represent the preferences of the decision maker. An achievement scalarization function (ASF)-based normalization is adopted in NSGA-III to handle objectives with different ranges of magnitudes. In addition, NSGA-III - like its predecessor NSGA-II consistently emphasized non-dominated solutions. Figure 1 shows the working principle of NSGA-III.

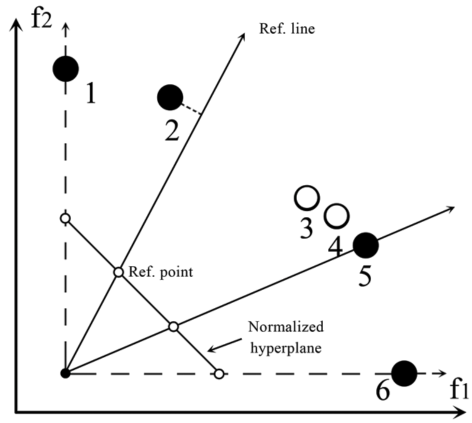


Fig. 1: NSGA-III working principle.

Although NSGA-III has been able to successfully deal with both constrained and unconstrained problems having up to 15 objectives in the original study, its performance with one and two objectives remained untested. The authors of NSGA-III did not include any single or bi-objective simulations in their study. Designing the algorithm for solving many-objective problems in mind, authors restricted the population size (N) to be roughly equal to the number of reference directions (H). The philosophy behind this restriction is that, by the end of each run, only one Pareto-optimal solution is meant to be discovered for each reference direction. Consequently, any selection between competing feasible population members was completely eliminated, because no two individuals arising to be ideal for different reference directions should be competing with each other. In fact, NSGA-III needs to keep surviving all population members, each in its designated location. This is also supposed to give the algorithm enough time to converge to the actual Pareto-optimal front without introducing too much pressure due to selection. Although this design was found to be extremely beneficial and frugal in the many-objective scenario, the algorithm is likely to lack adequate selection pressure for handling single and even bi-objective optimization problems. Let us discuss NSGA-III's apparent behavior in solving single and bi-objective problems in the following paragraphs.

For single-objective minimization problems, NSGA-III's non-dominated sorting procedure degenerates to an ordinal fitness-based sorting. Theoretically, only one reference direction (along the increasing fitness value) exists in this case (that is, $H = 1$), which is the positive real line. Thus, normalization and niching operations become defunct in single-objective case. According to the restriction imposed in NSGA-III on population size ($N = H$), there should be only one individual in the population. Because of the tournament-recombination operator pair, a population size multiple of four was suggested so that four population member produced two offspring members in one stroke. Thus, for $H = 1$, a population of size $N = 4$ is destined to be used for all single-objective optimization problems. Any experienced evolutionary computation (EC) researcher and applicationist knows that this is too small a population to use even for simple optimization problems. A population-based EC algorithm requires a certain minimum population size for its recombination-based operator to produce useful offspring members for the search to continue towards interesting regions of the search space [7]. For bi-objective problems, NSGA-III is expected to somewhat be handicapped due to the lack of any selection operator.

To resolve these issues, the authors recently proposed U-NSGA-III, a unified variant of NSGA-III. U-NSGA-III is unique in its ability to degenerate from an efficient many-objective optimization algorithm, to an efficient multi-objective algorithm, to degenerate to a standard single-objective optimization algorithm. An earlier study also proposed such a unified approach but the scope was restricted to one and two-objective optimizations only [11]. To design U-NSGA-III, a new niching-based selection operator was added to NSGA-III. The new operator performs a selection operation, if the two competing individuals belong to the same niche (reference direction). If they do, the lower-ranked individual is picked. If they both have the same rank, the one closer to the reference direction is picked. Figure 2 shows how the new selection operator works in U-NSGA-III. Obviously, selection will be applied more frequently as the difference ($N - H$) becomes larger. Thus, U-NSGA-III is free from the $N = H$ restriction of NSGA-III. Population size (N) can be larger than the number of reference directions (H). The frequency and effect of selection will be dependent on their relative values. For example, in single-objective scenarios, where $H = 1$ and $N \gg H$, selection will be performed at all times, because all the individuals will belong to the same unique reference direction. On the other extreme, in many-objective scenarios, where $N = H$ was restricted to make the algorithm computationally efficient, each solution is expected to be attached to a different direction, meaning that selection is practically absent. In that study, we were able to show the efficiency of U-NSGA-III in single, multi and many-objective problems and for both constrained and unconstrained situations. We compared the proposed unrestricted U-NSGA-III with the already existing restricted NSGA-III (where $N = H$). None of the simulations included in that study tested the performance of NSGA-III if the population restriction is ignored.

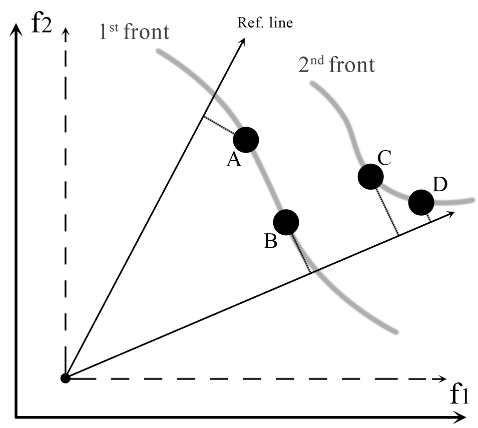


Fig. 2: Selection in U-NSGA-III.

In this study, we explore the possibility of ignoring the $N = H$ restriction when applying NSGA-III to different types of problems. Performance of the unrestricted (multi-fold) NSGA-III is compared to the performance of U-NSGA-III over a wide range of constrained and unconstrained problems. We propose two main hypotheses in this study and try logically and empirically validate their correctness.

In Section II we propose our two hypotheses and justify each through arguments. Results of our extensive simulations confirming our logical arguments are presented in Section III. Finally, Section IV concludes this study.

II. Hypotheses

From the previous section, we can see that the two main features characterizing NSGA-III are the absence of selection and the usage of reference directions as an external guidance mechanism to preserve diversity among solutions. In the following two subsections, our two main hypotheses are stated and discussed in detail in the highlight of the absence of selection in NSGA-III and its effect when N is significantly larger than H.

A. NSGA-III Convergence

As long as the population size is equal to the number of reference directions, the absence of selection is not expected to affect the results, because each reference direction is expected to attract only one point as they go further in generations. However, the situation becomes different if N is greater than H. In such a case, some population members will be guided by the reference directions while the others will keep floating randomly in the search space due to lack of any selection pressure for them to be focused anywhere in the search space (thereby providing excessive randomness to the additional population members). No selection to decide between individuals and not enough reference directions to guide the whole population, this effect should be more evident as the gap between N and H becomes bigger. This leads us to our first hypothesis.

Hypothesis 1 *In NSGA-III, a use of a population size larger than the number of reference directions will slow down the convergence of the algorithm towards the optimum.*

A direct consequence of the above hypothesis 1 is that NSGA-III should be slower than any equivalent evolutionary algorithm involving any form of selection that honors better solutions. This very same fact can be stated differently by saying that NSGA-III tends to be less greedy than other selection-based evolutionary algorithms.

B. NSGA-III and Local Optima

Although NSGA-III is expected to be slower than its selection-based counterparts, this is not always a disadvantage. Actually, this might be beneficial in some cases. Because of the inherent diversity of solutions maintained by additional population members in NSGA-III (discussed above), the algorithm is expected to have a higher ability to escape local optima. It is also expected to be less-dependent on mutation operators. Again, this behavior is justified, owing to the less-greedy nature of the algorithm. Hence, we can state our second hypothesis.

Hypothesis 2 *If $N > H$, NSGA-III will have a higher ability to escape local optima than its selection-based counterparts, and consequently it becomes less dependent on mutation operators.*

In Section III, our simulation results are presented and discussed in the highlight of these two hypothesis. Besides supporting our hypotheses, we also paint a clear picture of the performance of multi-fold NSGA-III to single, multi, and many-objective optimization problems.

III. Results

We conduct two experiments. In both experiments, we compare NSGA-III with U-NSGA-III. As mentioned earlier, U-NSGA-III is a variant of NSGA-III with the additional niching-based selection operator. The authors previously showed in [16] that U-NSGA-III in single-objective scenarios becomes equivalent to a standard $(\mu + \lambda)$ evolutionary strategy (ES) [2]. The parameters used in our single-objective simulations are shown in Table I. For each problem, $n = 20$ (number of decision variables) is used. The results shown in all the plots of this study are the medians of 11 simulations with the same set of parameters. Regarding higher dimensions, U-NSGA-III and NSGA-III have been compared elsewhere [16] abiding by the condition $N = H$ in both algorithms. Using such a setting, U-NSGA-III degenerates to NSGA-III after its additional selection operator loses its effect. Here in all our comparisons we use the same number of population folds for both algorithms. This will enable us to investigate the effect of selection in higher dimensions as well.

TABLE I: Parameter listing.

Parameter	Value
SBX distribution index (η_c)	0
Polynomial mutation distribution index(η_m)	20
Crossover Probability	0.75
Mutation Probability	0.02

A. Single-Objective Problems

The first experiment is to compare both NSGA-III and U-NSGA-III using multiple population folds. In single objective problems, only one reference direction exists, which is the extreme situation having $N >> H$. Our simulations is performed on a group of unconstrained as well as constrained problems spanning a wide range of difficulties. The unconstrained test problems are Ellipsoidal, Rosenbrock's, Schwefel's, Ackley's and Rastrigin's listed in Equations 1 through 5, while the constrained problems are G01, G02, G04, G06, G07, G08, G09, G10, G18 and G24 form the standard G-test suite [14]. Each figure compares both methods for three different population sizes: 100, 300 and 500. The only exception is the relatively easy Ellipsoidal problem, for which we used $N = 48$, 100 and 148. Log scale is used whenever necessary to show the differences in performance in late generations. It is clear from Figures 3 and 17 that NSGA-III converges *slower* than U-NSGA-III in both constrained and unconstrained problems across all three population sizes, thereby confirming the correctness of our Hypothesis 1. In most cases, NSGA-III is able to catch up with U-NSGA-III if enough time (or generations) is given. In some cases, especially in constrained problems, like G01, G06, G07 and G10, NSGA-III needs almost twice the number of generations needed by U-NSGA-III to achieve the same level of convergence.

We also included the median fitness value per generation of 11 NSGA-III runs where $N = 4$ (blue line), in order to evaluate the usefulness of multiple folds compared to a single fold NSGA-III. For the blue line, the x-axis does not represent the number of generations anymore. However, to maintain a fair comparison, at any given point on the x-axis, the blue line and the dotted black and red lines have the same number of function evaluations. The figures show that NSGA-III where $N = 4$ tends to converge quickly at the beginning; then it gets stuck after a while without reaching the global optimum. This is expected because of the too small population size ($N = 4$) which makes the algorithm prone to be trapped in local optima even if given the same number of function evaluations as the multiple folds version of the algorithm.

$$f_{\text{elp}}(\mathbf{x}) = \sum_{i=1}^{n} i x_i^2, \quad -10 \leq x_i \leq 10, i = 1, \ldots, n \tag{1}$$

$$f_{\text{ros}}(\mathbf{x}) = \sum_{i=1}^{n-1} [100({x_i}^2 - x_{i+1})^2 + (x_i - 1)^2], \quad -10 \leq x_i \leq 10, i = 1, \ldots, n \tag{2}$$

$$f_{\text{ack}}(\mathbf{x}) = -20 \exp\left[\frac{-1}{5}\sqrt{\frac{1}{n}\sum_{i=1}^{n} {x_i}^2}\right] - \exp\left[\frac{1}{n}\sum_{i=1}^{n} cos(2\pi x_i)\right] + 20 + e, \quad -32.768 \leq x_i \leq 32.768, i = 1, \ldots, n \tag{3}$$

$$f_{\text{ras}}(\mathbf{x}) = 10n + \sum_{i=1}^{n} [x_i^2 - 10\cos(2\pi x_i)], \quad -5.12 \leq x_i \leq 5.12, i = 1, \ldots, n \tag{4}$$

$$f_{\text{sch}}(\mathbf{x}) = 418.9829n - \sum_{i=1}^{n} (x_i \sin\sqrt{|x_i|}), \quad -500 \leq x_i \leq 500, i = 1, \ldots, n \tag{5}$$

B. Multi and Many-Objective Problems

In higher dimensions however, diversity becomes an as important goal as convergence. Since Hypothesis 1 relates only to convergence speed, we needed a way of testing only the convergence ability of each algorithm first. Having this in mind, we use the Generational Distance (GD) as our metric. Another important aspect is problem selection. Our test problems should primarily test the convergence ability of the algorithm, again to check the validity of Hypothesis 1. For this reason, we choose ZDT4 [18] as our bi-objective test bed. ZDT4 has $(21^9 - 1)$ local Pareto optimal fronts. Unless the convergence ability of the algorithm is good, it will get trapped in one of these local fronts. We also included

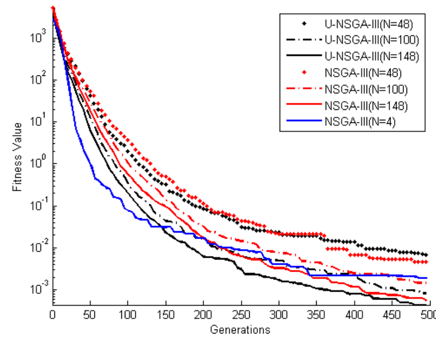


Fig. 3: Ellipsoidal problem.

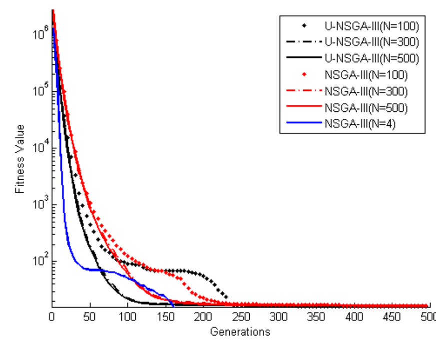


Fig. 4: Rosenbrock's problem.

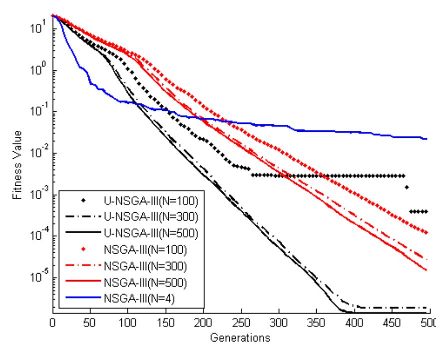


Fig. 5: Ackley's problem.

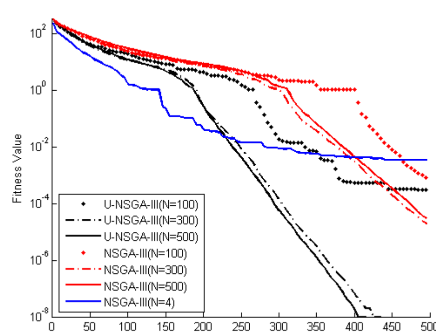


Fig. 6: Rastrigin's problem.

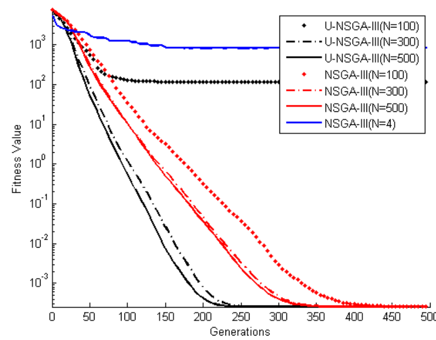


Fig. 7: Schwefel's problem.

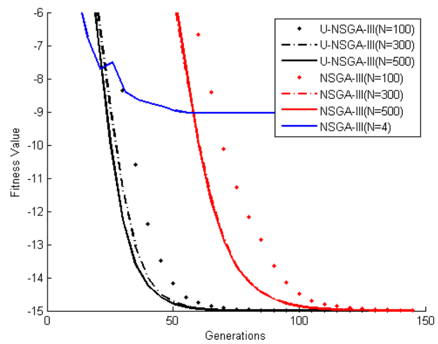


Fig. 8: G1 problem.

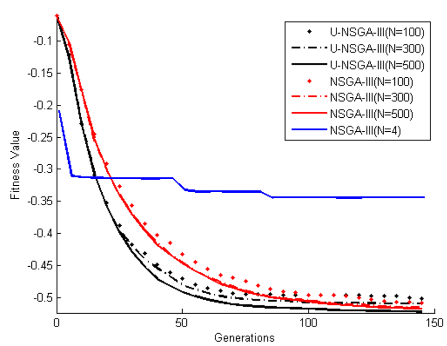


Fig. 9: G2 problem.

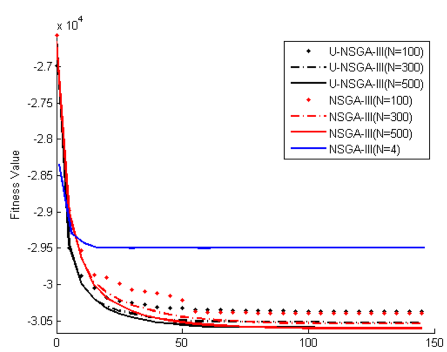


Fig. 10: G4 problem.

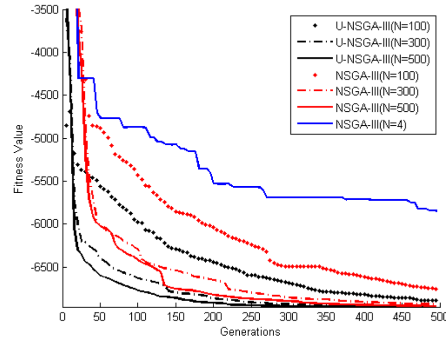


Fig. 11: G6 problem.

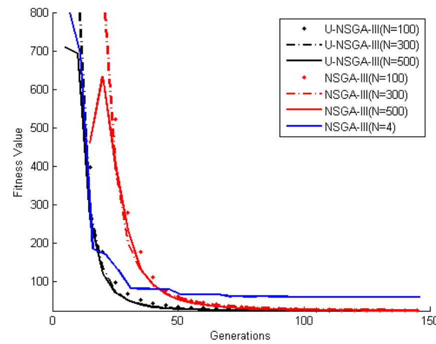


Fig. 12: G7 problem.

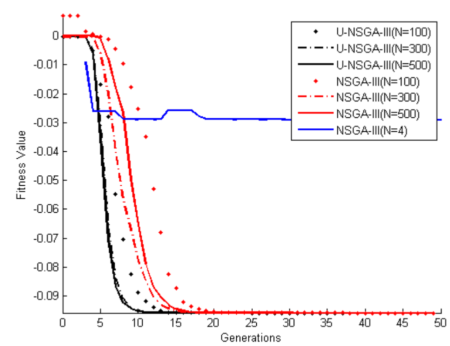


Fig. 13: G8 problem.

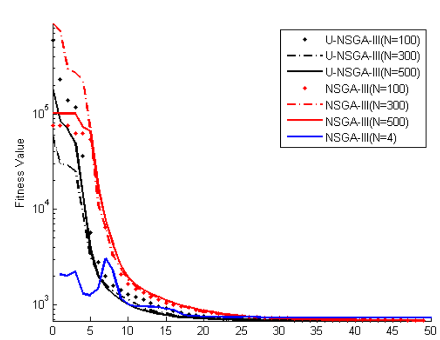


Fig. 14: G9 problem.

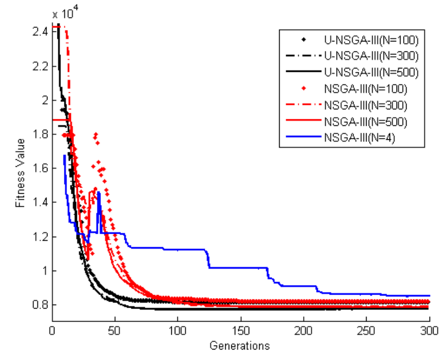


Fig. 15: G10 problem.

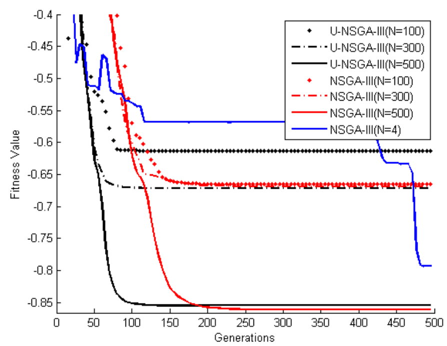


Fig. 16: G18 problem.

two easy problems, ZDT1 and ZDT2 to show the effect of selection in simple and easy problems.

For three and five objectives, we choose multi-modal DTLZ1 problem [10], which has $(11^5 - 1)$ local Pareto-optimal fronts, requiring good convergence ability as well.

In simple and easy problems containing no local Pareto-optimal fronts, both the algorithms exhibit an almost identical convergence behavior, as shown in Figures 18 and 19 (except the blue lines). This is expected, because in order for the difference to show up between these algorithms, the problems should exhibit adequate difficulties preventing them from reaching the global Pareto-optimal front. This difficulty is introduced in ZDT4. Figure 20 shows the median GD metric of 11 runs for 500 generations using $N = 1.5H, 2H$ and $2.5H$. It is now clear from the figure that U-NSGA-III converges faster than NSGA-III. As the population size goes up, NSGA-III's convergence ability approaches that of U-NSGA-III. The difference between the two algorithms becomes more obvious from Figures 21 and 22. Final fronts of the best, median and worst runs for the two simulations $(N = 1.5H)$ and $(N = 2H)$ are shown in these two figures, respectively. The difference is negligible in the best runs. It increases gradually becoming visible at median runs, and relatively large at worst runs. Using a smaller population size increases the difference in convergence between the two algorithms.

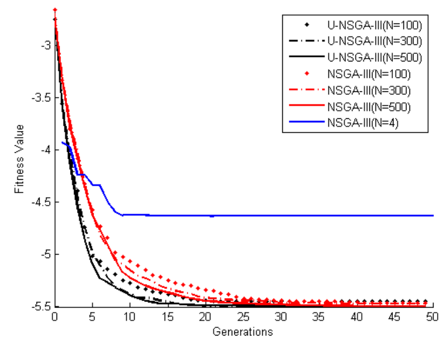


Fig. 17: G24 problem.

Figures 23 and 24 show the behavior of the GD metric for DTLZ1 in three and and five-objective problems, respectively. Although, U-NSGA-III still maintains a slight edge over NSGA-III in terms of convergence, the difference is small. This can be attributed to the fact that convergence becomes less important compared to diversity maintenance as the number of objectives increases. All these results clearly support the correctness of our Hypothesis 1 for bi-objective and multi-objective problems.

In order to justify using multiple folds in multi-objective scenarios, the median GD plots of 11 runs of ZDT1, ZDT2 and ZDT4 (single fold) are added to Figures 18, 19 and 20, respectively (blue lines). Obviously, using only a single fold NSGA-III is unable to converge to the global Pareto-optimal front.

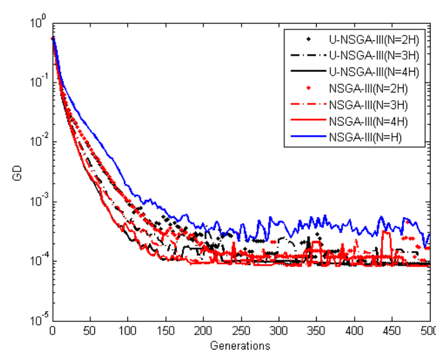


Fig. 18: ZDT-1 Median GD (H=25, N=2H, 3H, 4H). Blue line is for $N = H$.

C. *Multi-modal Problems*

Our second experiment tests the ability of NSGA-III to escape local optima and its dependence on the mutation operator compared to U-NSGA-III. Again, U-NSGA-III is used here as a candidate single-objective algorithm involving selection. For our purpose, we choose three highly multi-modal problems, namely, Ackley's, Rastrigin's and Schwefel's. A relatively small population size $(N = 48)$, making both the algorithms more prone to be trapped in local optima. We set the mutation probability to zero, thus completely removing mutation

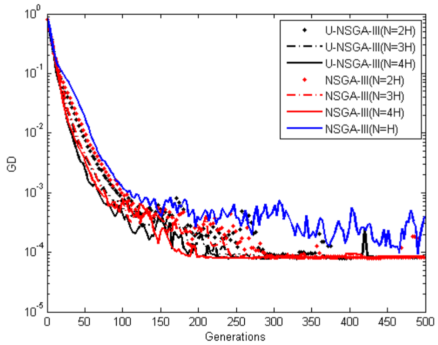


Fig. 19: ZDT-2 Median GD (H=25, N=2H, 3H, 4H). Blue line is for $N = H$.

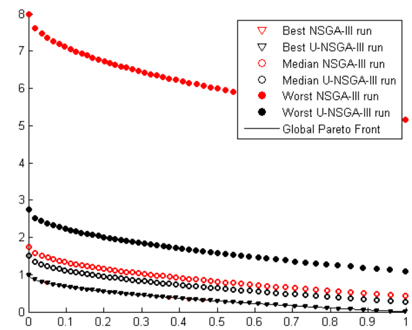


Fig. 21: ZDT-4 Final Generation (H=50, N=1.5H).

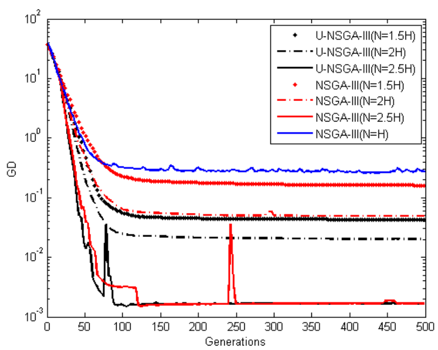


Fig. 20: ZDT-4 Median GD (H=50, N=1.5H, 2H, 2.5H). Blue line is for $N = H$.

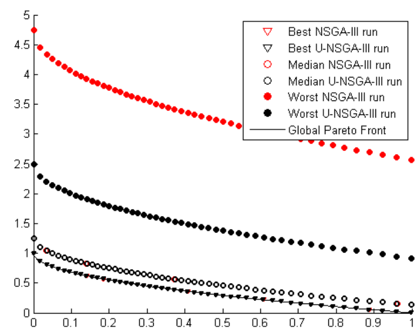


Fig. 22: ZDT-4 Final Generation (H=50, N=2H).

from both. Now the ability of the algorithm itself to escape local optima – without any benefit from the mutation operator – can be clearly observed. As shown in Figure 25, the ability of NSGA-III to escape local optima is evident compared to U-NSGA-III, which got easily trapped in local optima in the absence of mutation. This goes hand in hand with the less-greedy nature of NSGA-III discussed earlier, and confirms the correctness of Hypothesis 2.

IV. Conclusions

In this study, we have explored the performance of a multi-fold version of a recently proposed evolutionary many-objective optimization algorithm (NSGA-III), in which the population size is significantly larger than the number of pre-specified reference directions. This setting has never been tested before with NSGA-III. Through a careful set of computational optimizations, we have revealed a new and extended scope of the multi-fold NSGA-III. We have also proposed two hypotheses and discussed the correctness of each of them, both logically and empirically. Hypothesis 1 states that NSGA-III is slower than its selection-based counterparts when it comes to convergence. Hypothesis 2 states that NSGA-III has a higher ability to escape local optima and is less-dependent on mutation operators compared to its selection-based counterparts. A logically driven discussion of both hypothesis, mainly attributes their correctness to the less greedy nature of NSGA-III due to the absence of selection. Finally, two carefully designed experiments have been conducted and the results obtained from both supported the correctness of our hypotheses.

The excellent performance of NSGA-III in many-objective problems demonstrated in the original two-part papers [9], [13] has now been aided in this paper by showing the performance of a multi-fold NSGA-III approach. Although the multi-fold NSGA-III retains original NSGA-III's performance in many-objective problems, it is now demonstrated to work quite well in single and multi-objective problems as well. Thus, along with the original study, this study portrays a much wider scope of NSGA-III for solving different types of problems than what the authors of original NSGA-III had perceived.

References

[1] S. F. Adra and P. J. Fleming. Diversity management in evolutionary many-objective optimization. *IEEE Transactions on Evolutionary Computation*, 15(2):183–195, 2011.

[2] H.-G. Beyer and H.-P. Schwefel. Evolution strategies: A comprehensive introduction. *Natural computing*, 1:3–52, 2003.

[3] C. A. C. Coello, D. A. VanVeldhuizen, and G. Lamont. *Evolutionary Algorithms for Solving Multi-Objective Problems*. Boston, MA: Kluwer, 2002.

[4] D. W. Corne, J. D. Knowles, and M. Oates. The Pareto envelope-based selection algorithm for multiobjective optimization. In *Proceedings of the Sixth International Conference on Parallel Problem Solving from Nature VI (PPSN-VI)*, pages 839–848, 2000.

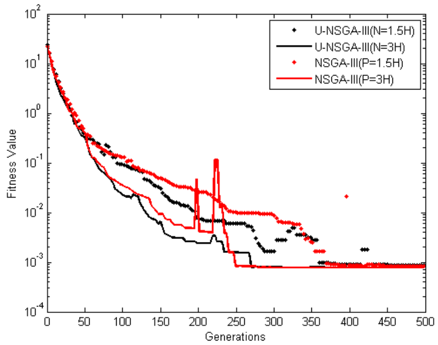


Fig. 23: DTLZ-1 (3 Obj.) Median GD (H=55, N=1.5H, 3H).

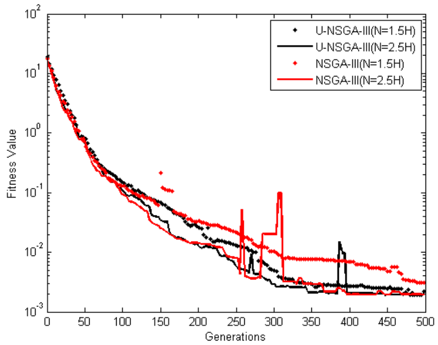
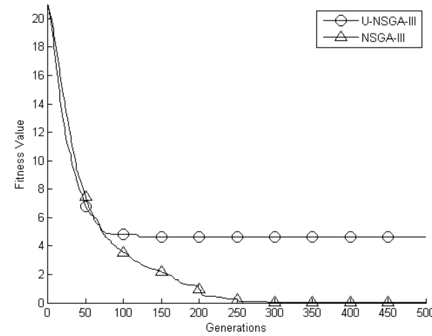
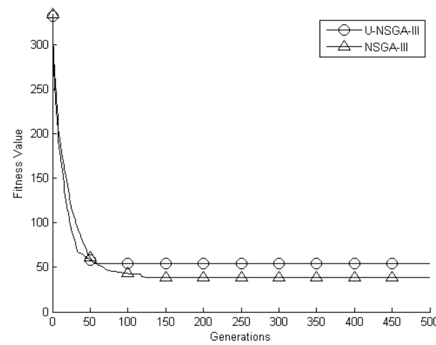


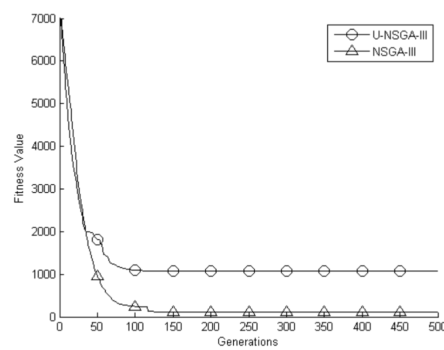
Fig. 24: DTLZ-1 (5 Obj.) Median GD (H=73, N=1.5H, 2.5H).



(a) NSGA-II



(b) NSGA-III



(c) NSGA-II

Fig. 25: Small population size in multi-modal problems.

[5] I. Das and J.E. Dennis. Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems. *SIAM Journal of Optimization*, 8(3):631–657, 1998.

[6] K. Deb. *Multi-objective optimization using evolutionary algorithms*. Wiley, Chichester, UK, 2001.

[7] K. Deb and S. Agrawal. Understanding interactions among genetic algorithm parameters. In *Foundations of Genetic Algorithms 5 (FOGA-5)*, pages 265–286, 1999.

[8] 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.

[9] K. Deb and H. Jain. An evolutionary many-objective optimization algorithm using reference-point based non-dominated sorting approach, Part I: Solving problems with box constraints. *IEEE Transactions on Evolutionary Computation*, 18(4):577–601, 2014.

[10] K. Deb, L. Thiele, M. Laumanns, and E. Zitzler. Scalable test problems for evolutionary multi-objective optimization. In A. Abraham, L. Jain, and R. Goldberg, editors, *Evolutionary Multiobjective Optimization*, pages 105–145. London: Springer-Verlag, 2005.

[11] K. Deb and S. Tiwari. Omni-optimizer: A generic evolutionary algorithm for global optimization. *European Journal of Operations Research (EJOR)*, 185(3):1062–1087, 2008.

[12] D. Hadka and P. Reed. BORG: An auto-adaptive many-objective evolutionary computing framework. *Evolutionary Computation*, 21(2):231–259, 2013.

[13] H. Jain and K. Deb. An evolutionary many-objective optimization algorithm using reference-point based non-dominated sorting approach, Part II: Handling constraints and extending to an adaptive approach. *IEEE Transactions on Evolutionary Computation*, 18(4):602–622, 2014.

[14] J. J. Liang, T. P. Runarsson, E. Mezura-Montes, M. Clerc, P. N. Suganthan, C. A. C. Coello, and K. Deb. Special session on constrained real-parameter optimization (http://www.ntu.edu.sg/home/epnsugan/), 2006.

[15] H. Sato, H. E. Aguirre, and K. Tanaka. Pareto partial dominance moea in many-objective optimization. In *Proceedings of Congress on Computational Intelligence (CEC-2010)*, pages 1–8, 2010.

[16] H. Seada and K. Deb. U-NSGA-III: A unified evolutionary algorithm for single, multiple, and many-objective optimization. Technical Report COIN Report Number 2014022, Computational Optimization and Innovation Laboratory (COIN), Electrical and Computer Engineering, Michigan State University, East Lansing, USA, 2014.

[17] Q. Zhang and H. Li. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *Evolutionary Computation, IEEE Transactions on*, 11(6):712–731, 2007.

[18] E. Zitzler, K. Deb, and L. Thiele. Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary Compu-*

tation Journal, 8(2):125–148, 2000.

[19] E. Zitzler, M. Laumanns, and L. Thiele. SPEA2: Improving the strength Pareto evolutionary algorithm for multiobjective optimization. In K. C. Giannakoglou et al., editor, *Evolutionary Methods for Design Optimization and Control with Applications to Industrial Problems*, pages 95–100. International Center for Numerical Methods in Engineering (CIMNE), 2001.