Application NSGA-II and its differential evolution counterpart

1st Jose Daniel Lopez Hernandez
School of Science and Engineering
Tecnologico de Monterrey
Puebla, Mexico
josedaniel2499@gmail.com

***Abstract*—This paper will show an introduction to multi-objective optimization methods, starting by explaining the main differences in the approached for single and multi objective optimization. Furthermore, an implementation of NSGA-II will be compared to another multi-objective optimizator that works similar to NSGA-II but has differential evolution techniques replacing crossover and mutation, aptly named NSDE-II. The main comparison between these will show the optimal pareto front they should reach, and compare their R2 score as well as their Riesz s-energy.**

***Index Terms*—Multi-objective, Wilcoxon, NSGA, graphing.**

I. Introduction

Multi-objective optimization refers to the problems where one has to use the same set of variables to optimize the value of a set of functions. The optimization might be different for each problem, for example, some problems may ask you to maximize or minimize the functions, other problems can add weights to the the functions, making the optimization of one of them more important, meaning that you are inclined to search for non optimal solutions for the other functions [1]. All of these considerations heavily differ from single objective optimization, where no matter what you want to obtain as good of a solution as the constraints and the functions allow you.

II. Background

A. Genetic Algorithms

Genetic algorithms are algorithms that create individual solutions called population, or phenotype. These solutions then are evaluated as potential solutions to the problem, and depending on their correctness are assigned a fitness value. By finding the best individuals of a population, these can be reproduced, by creating new individuals that are similar to the best of the last generation, a solution can be approximated. However this could easily fall into local solutions, so by using a mutation parameter, exploration increases and global solutions are easier to find [2].

B. Multi Objective Optimization Algorithms

When tackling optimization for multi-objective problems it is important to note that the answer will most likely never reach an optimal value for individual functions, instead the algorithms seek to find an area, where all solutions have an optimal solution, this is found by searching which solutions dominate others, and all of the solutions that are non-dominated are included in the area, this area is called the pareto front, figure 1 shows this front, to show what dominance is, you can see it on the graph with point A, B and C. The result value of f1 and f2 are lower for A and B than they are for C, this means that for all functions, both A and B are overall better than C. This only happens when all the values of the functions are equal or lower in one point than in the other. For example, while C has a better f2 value than the point to the right of B, it has a worse f1 value, so neither point dominates each other, the reason C is not in the pareto front is due to it being dominated by both A and B [1].

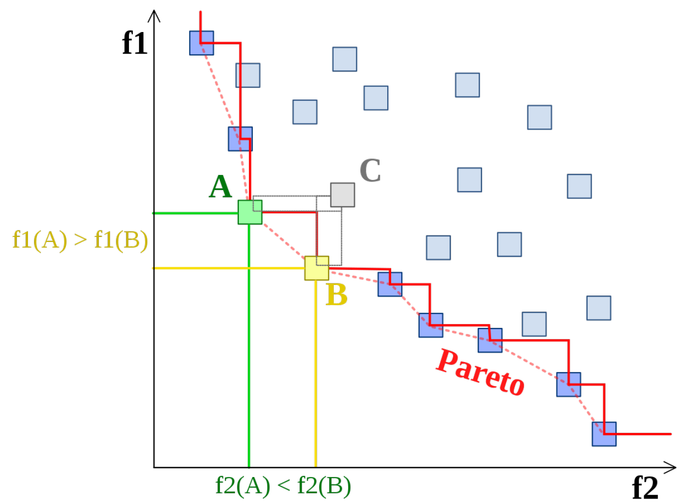

Fig. 1. Pareto Front

C. NSGA-II

NSGA-II stands for nondominated sorting genetic algorithm II, as mentioned by the name, this is indeed a genetic algorithm and this shows in the population, while the initial population is akin to any other algorithm, the way new generations are created is by crossover and mutation, showing basic genetic algorithm ideas. The main difference between NSGA-II and other genetic algorithms (besides it being used for multi-objective optimization) is the fact that children matrix includes

Identify applicable funding agency here. If none, delete this.

	dtlz1	dtlz2	dtlz5	dtlz7
NSGA R2 mean	-434.61	-58.7072	-12.578	-454.86
NSGA R2 std	573.314	13.2585	8.8933	260.359
NSGA RSE mean	11.911	.70106	.73404	2.4188
NSGA RSE std	31.687	.04489	.0566	10.96
NSDE R2 mean	-1459.69	-124.747	-19.78	-815.14
NSDE R2 std	1226.85	8.5653	8.2438	431.78
NSDE RSE mean	.10517	.7216	.7340	.3593
NSDE RSE std	.1677	.0468	.05664	.44185

TABLE I
COMPARISON OF ALGORITHMS

	idtlz1	idtlz2	idtlz5	idtlz7
NSGA R2 mean	-18404.91	-109.32	54.8679	1864.69
NSGA R2 std	2188.779	34.358	9.764	788.54
NSGA RSE mean	.0100	.75844	.7595	33333.52
NSGA RSE std	.002	.1045	.1204	179505
NSDE R2 mean	19660	179.53	54.867	2562.39
NSDE R2 std	850.95	8.592	9.7642	398.345
NSDE RSE mean	.0110	.7583	.7595	.2562
NSDE RSE std	.0056	.1394	.1204	.3048

TABLE II
COMPARISON OF ALGORITHMS ON INVERSE FUNCTIONS

parents which have the max crowdin distance of the best pareto front.

III. DESCRIPTION OF THE PROPOSALS

A. NSGA-II

The NSGA-II to be applied for this comparison has the following characteristics:

- **Selection**: Tournament Selection, the best individuals of each generation are selected for breeding
- **Crossover**: Using Simulated Binary Crossover (SBX)
- **Mutation**: Using Polynomial mutation

B. NSDE-II

Standing for nondominated sorting differential evolution II, is a proposed alternative to NSGA-II, it uses the same algorithm as it. The main difference are the crossover and mutation which are changed to accommodate the differential evolution concepts in single objective optimization. In this case the crossover will be the Exponential crossover and differential mutation.

C. Indicator

Due to how complicated it is to accurately measure the results of an multi-objective optimization (due the expected results being in theory mostly optimal), indicator are used to see how varied, and correct these answers are. In this case the R2 [3] and the Riesz s-energy [4] will be shown, furthermore the Wilcoxon rank sum test will be done as well.

IV. RESULTS

A. Comparison of the GA and DE algorithms

This comparison will be made on 8 of the problems, 4 showed in [5], namely dtlz1, dtlz2, dtlz5, dtlz7 and the other 4 titles idtlz1, idtlz2, idtlz5, idtlz7 which stand for the inverse of the last problems. 30 runs of each algorithm will be made, with 20 population each and 20 generations. The best fitness individual of each generation will be put into an array, from which a final pareto front will be found to find how many generation individuals are placed. The indicator will be taken from each run and the mean and standard deviation of every run of the fist four algorithms will be shown in table I. The comparison of the inverse results will be shown in table II

To further compare the algorithms a Wilcoxon rank-sum [6] comparison will be made of their indicators, the positive ranges will mean greater values in NSGA while negative values will mean greater values in NSDE.

	dtlz1	dtlz2	dtlz5	dtlz7
R2 statistic	4.86	-6.652	3.1934	3.1934
RSE statistic	3.977	-2.321	1.005	-0.7835

TABLE III
STATISTICAL VALUE OF WILCOXON RANK-SUM

	idtlz1	idtlz2	idtlz5	idtlz7
R2 statistic	-2.29	-6.608	.0591	-3.829
RSE statistic	.3548	-5026	2.72	-.0739

TABLE IV
STATISTICAL VALUE OF WILCOXON RANK-SUM FOR INVERSE FUNCTIONS

Finally the graphs showing the comparison of the best pareto frontier of the algorithms are shown in the following graphs. In these graphs, the optimal solutions are also shown for the non-inverse functions. The average sum of the functions is found in the figure caption.

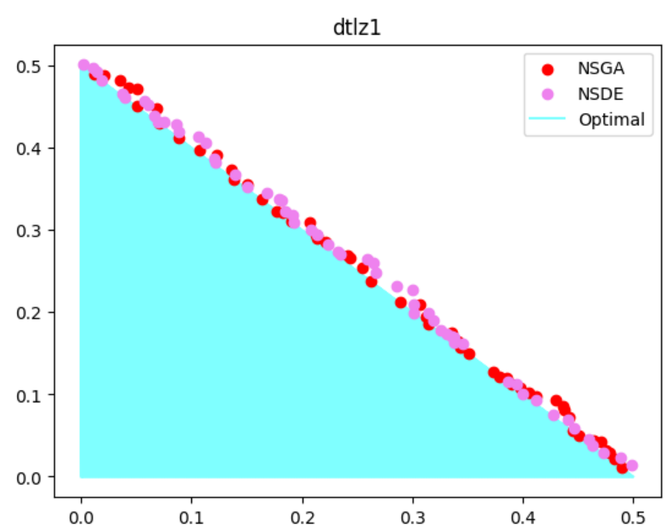

Fig. 2. dtlz1 fronts$\sum_x^n f_x = .521$

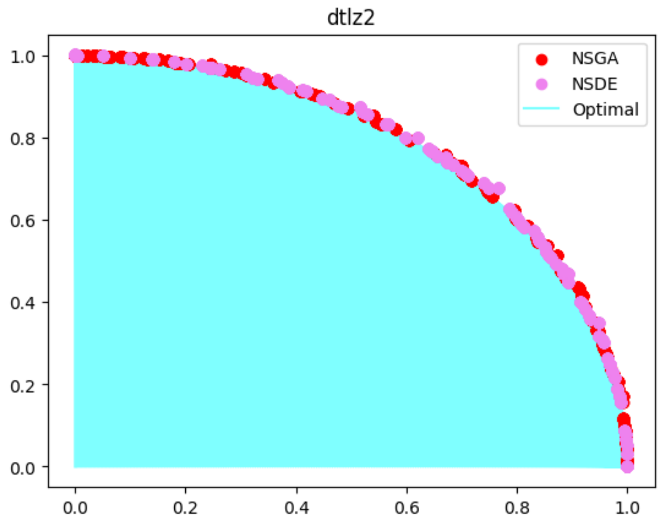

Fig. 3. dtlz2 fronts $\sum_x^n f_x^2 = 1.07$

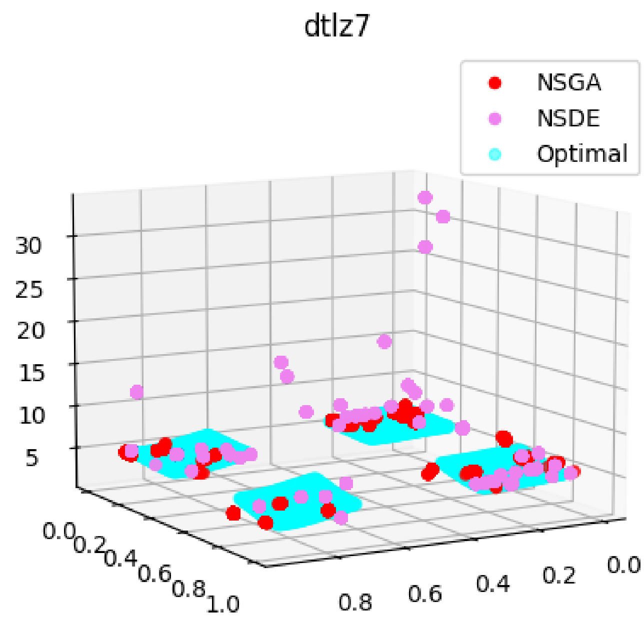

Fig. 5. dtlz7 fronts $\sum_x^n f_x =$

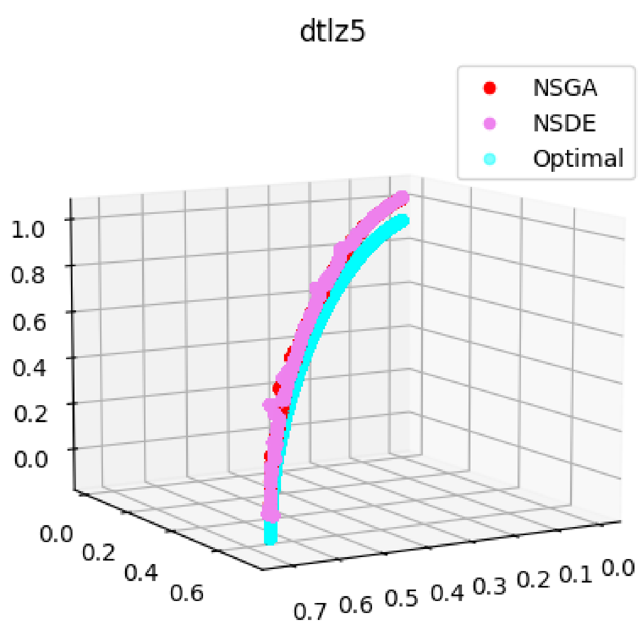

Fig. 4. dtlz5 fronts $\sum_x^n f_x = 1.12$

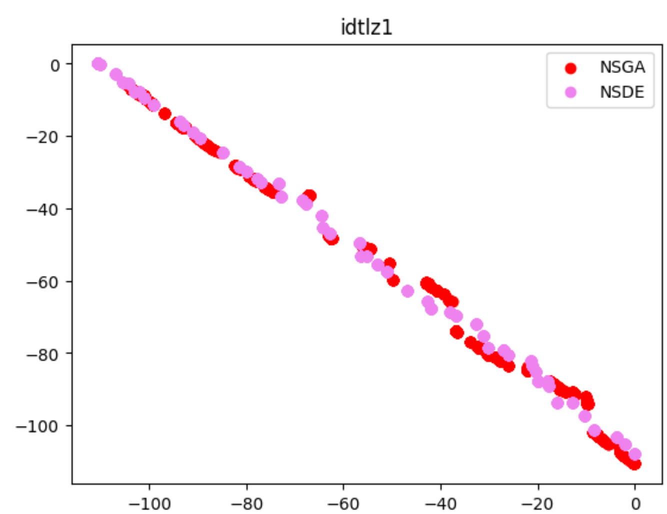

Fig. 6. idtlz1 fronts $\sum_x^n f_x = -110$

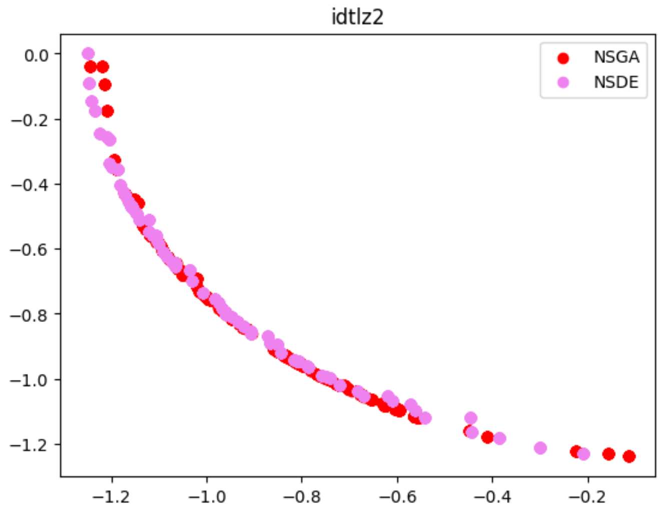

Fig. 7. idtlz2 fronts $\sum_x^n f_x^2 = -3$

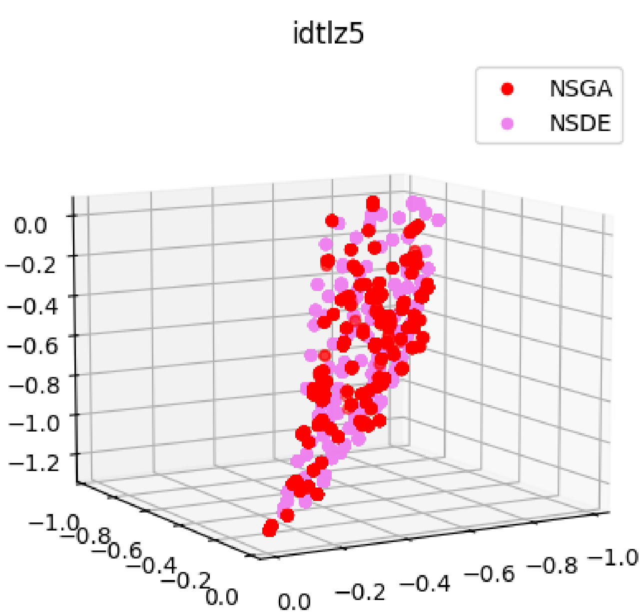

Fig. 8. idtlz5 fronts $\sum_x^n f_x = -2.162$

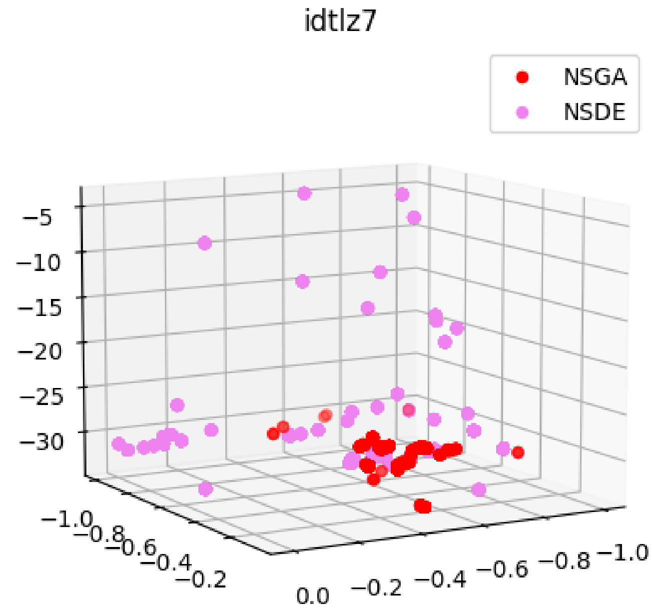

Fig. 9. idtlz7 fronts $\sum_x^n f_x = -33.8$

V. Discussion

Starting with the indicators in table I, it is easy to note that the R2 indicators are not very good. excluding dltz2 and dltz5, the results vary too much, which heavily affects the R2 results in dtlz1 this is exaplained by it having a multifrontal solution, meaning that solutions might get stuck on non-optimal pareto fronts, for dltz7 this is explained by it having separated optimal solutions fronts, meaning that variations is expected. This is further supported by the RSE parameters, dltz1 and dltz7 have a higher distribution of variables, while the others have a more defined answer, having less variation. Following this trend, table II shows very bad values of R2 giving very big distribution of values mainly because the idtlz series of problems have very big optimal pareto fronts due to them being maximization problems. RSE values are very small, which means that there is not a lot of exploration done by the algorithms, this means that all of them are around the same are in the end, meaning that the optimal solution is found really early. In the case of idtlz7, the exploration is good for the NSGA while having better values than NSDE, this is in part explained with their non-optimal values, while NSDE is all over the place, NSGA is concentrated so all of the non-pareto optimal values are seen as discrepancies.

Continuing with the wilcoxon ratings, in table III it shows that the genetic algorithm has a bigger R2 value, this is due to it being a smaller negative number in comparison, meaning that it has a better mean and less variation. However RSE values signify that solutions are more distribuited for NSGA in general, NSDE only has more exploration in dltz2 and dltz7 and not by much. On the other hand, table IV interprets as the NSDE having very bad R2 results in comparison to NSGA.

However they all have very similar RSE values, meaning that solutions are almost as close together as they can be.

All of the normal dltz problems reach an optimal pareto front with a lot of values, in dltz7 NSDE has some problem with extreme cases, but most values are around the optimal planes. In order to get the optimal front dltz1 took the most amount of attempts, however all of the function sums signify optimal values except for dltz7 which does not have a particular optimal sum of values meaning that visualization is the only guide.

VI. Conclusions

All in all, the algorithms proved to work by obtaining optimal solutions in the normal algorithms, the lack of optimal pareto for the inverse solutions means that they are up to interpretation if they are correctly working or not, since they are maximizing problems, finding the biggest value is their objective, and it seems they achieved that within the variable constraints placed (which for all problems is between 0 and 1 for every decision variable). The high R2 values are exaplained by all the generations at the start which have horrible solutions, they heavily affect the R2 value, and in most cases the R2 value could be interpreted as a convergence time, higher values mean slower convergence, on the other hand most RSE values were small which mean that most solutions were localized solely on the paretro front and variation was not high.

References

[1] E. M., "A tutorial on multiobjective optimization: fundamentals and evolutionary methods," *Nat Comput*, vol. 17, pp. 585–609, 2018.

[2] K. Sastry, D. Goldberg, and G. Kendall, *Genetic Algorithms*, pp. 97–125. Boston, MA: Springer US, 2005.

[3] B. Dimo *et al.*, "On the properties of the r2 indicator," *2012 Genetic and Evolutionary Computation Conference*, pp. 465–472, 7 2012.

[4] J. Falcón-Cardona, E. Covantes Osuna, and C. Coello, "An overview of pair-potential functions for multi-objective optimization," pp. 401–412, 11 2021.

[5] D. Kalyanmoy *et al.*, *Scalable test problems for evolutionary multobjective optimization*, vol. 6. Springer, 4 2005.

[6] C. Ford. https://library.virginia.edu/data/articles/the-wilcoxon-rank-sum-test: :text=The