2nd International Conference on Scientific and Academic Research
March 14-16, 2023 : Konya, Turkey
All Sciences Proceedings
http://as-proceeding.com/

ICSAR

Oscillation-based Linear Dynamic Sampling Allocation for Noisy Multiobjective Optimization Problems

Tolga Altinoz[1*]

[1]Department of Electrical and Electronics Engineering, Ankara University, Turkey

*(taltinoz@ankara.edu.tr) Email of the corresponding author

Abstract – The physical quantities in real-life can be measured by using devices called sensors. These devices are not perfect/ideal devices which have the properties called sensitivity and resolution. As the sensors became more sensible with a higher resolution the current/real physical data can be obtained. However even by using the best possible sensor, the sensor still under the measurement noise. Therefore, it is natural to have noise in engineering problems. To solve these engineering problems the noise should be considered in the calculations. In Multiobjective optimization this noise can be added to the objectives and called noise Multiobjective optimization problems. To solve these problems the most common method is called re-sampling which is the calculation the objectives many times and taking the average of their values. The dynamic re-sampling is a method for efficient with respect to the computational source. In this research a new dynamic re-sampling method is proposed and named as oscillation-based linear dynamic sampling method. This method is integrated into four different Multiobjective optimization algorithms and applied to eight benchmark problems. The results showed that the proposed method gives acceptable results with relatively small number of additional function evaluation.

Keywords – Optimization, Multiobjective, Computational Optimization, Noise, Decomposition

I. Introduction

The multiobjective optimization algorithms are composed of objective functions more than one, and for this reason unlike single objective optimization problems not a single solution but a set of solutions are obtained from the problems. In noise Multiobjective optimization the noise is added to all objectives as defined in Equation 1.

$$F_{noisy}(x) = \overline{F(x)} + \overline{r} \quad (1)$$

where noise vector (r) is added/summed with each objective value (F), where Normal distribution with zero mean and 0.15 standard deviation is added as the noise. The decision variables (x) are defined inside and range with upper and lower boundary. The best solution is known as Pareto set and the shape of the objective values of the Pareto set is called the Pareto front. From the optimization algorithms it is expected to generate solutions which are close to the Pareto front and distributed among it.

One of the possible solutions for the noisy optimization problems is called averaging or re-sampling method [7,8,9]. In this method the objective functions are evaluated more than one and the average of this calculation is taken as the objective value. Therefore, the effect of the noise decreases. There are two types of the re-sampling method exists which are static and dynamic re-sampling methods. In static method for each iteration same number of functions are evaluated. This method needs more computations resources and calculates unnecessary function evaluations. Because at the initial iterations it is expected from the optimization algorithm to explore the search space. Therefore, noisy objective may help the algorithm. To reduce and using the computational resources efficiently, dynamic re-sampling is a good method for noisy optimization problems.

Table 1. IGD values for the proposed method for a = 2

Problem	M	D	NMPSO	DMOEAeC	MOEAD	NSGAII
BT1ND	2	30	3.4835e+0 (1.40e-1) -	2.3256e+0 (6.40e-1) =	3.5405e+0 (1.46e-1) -	2.6123e+0 (3.80e-1)
BT2ND	2	30	1.2463e+0 (2.07e-1) -	2.6078e-1 (2.27e-1) +	8.0586e-1 (1.18e-1) -	3.4036e-1 (6.92e-2)
BT3ND	2	30	3.3255e+0 (5.84e-1) -	3.5528e-1 (2.34e-1) +	3.0282e+0 (4.72e-1) -	7.7404e-1 (3.65e-1)
BT4ND	2	30	2.9905e+0 (5.03e-1) -	3.0392e-1 (8.49e-2) +	2.7558e+0 (2.44e-1) -	5.3633e-1 (2.30e-1)
BT5ND	2	30	3.3144e+0 (2.54e-1) -	1.9186e+0 (2.59e-1) +	3.5002e+0 (1.77e-1) -	2.3792e+0 (3.03e-1)
BT6ND	2	30	1.3101e+0 (4.33e-1) -	4.5562e-1 (9.73e-2) =	3.2381e-1 (7.96e-2) +	5.2458e-1 (1.24e-1)
BT7ND	2	30	1.4979e+0 (6.38e-1) -	6.5944e-1 (1.37e-1) -	4.8487e-1 (1.33e-1) =	3.7472e-1 (9.32e-2)
BT8ND	2	30	3.6347e+0 (1.38e+0) -	5.0775e-1 (2.23e-1) =	2.1622e+0 (5.97e-1) -	4.2716e-1 (2.13e-1)
+/-/=			0/8/0	4/1/3	1/6/1	

In this study a new dynamic re-sampling method is proposed which is named as Oscillation-based Linear Dynamic Sampling Allocation method. In this method sample size is detected by using the current number of function evaluation and oscillated manner to get/select the sample size. By this way the exploration property of the optimization algorithm is supported by this proposed method therefore with a relatively small number of additional function calculations better results will be obtained.

This paper is organized beginning with the introduction section. After the introduction section at the material and method section the proposed dynamic re-sampling method with the brief explanations of the optimization algorithms and benchmark functions with performance measurement metrics are explained. After the implementation section the conclusion is given as the final section.

II. Materials and Method

In this section the proposed dynamic re-sampling method is explained. Next the optimization algorithms are explained briefly. The reader gets detailed information from the reference of algorithm. Finally, the benchmark problems and performance measurements used in this research is presented.

A. *Dynamic Re-sampling*

The static re-sampling methods needs more evaluations of the objective functions when compared with dynamic re-sampling methods. In this research a new re-sampling method is proposed and named as oscillation-based linear dynamic sampling method. The mathematical description of the method is given in Equation 2.

$$S = ceil\left[a + 2sin\left(\frac{2\pi FE}{FE_{max}}\right)\right] \quad (2)$$

where S is the dynamic sampling size, a is the parameter which is the mean of the sample size, FE is the current function evaluation and FE_{max} is the maximum number of function evaluation. The sample size is changes with respect to the number of function evaluations. The oscillation-based behaviour of the sample size is demonstrated in Figure 1 and mathematically in Equation 2.

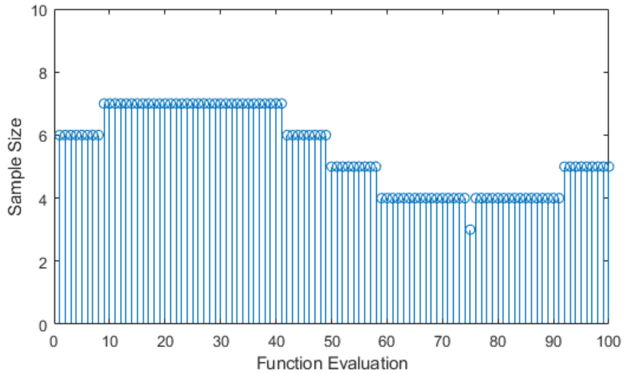

Figure 1. Distribution of the Sample Size for Dynamic Re-sampling Method for a=5

Table 2. IGD values for the proposed method for a = 3

Problem	M	D	NMPSO	DMOEAeC	MOEAD	NSGAII
BT1ND	2	30	3.4451e+0 (1.35e-1) -	2.0536e+0 (8.09e-1) =	3.4935e+0 (1.35e-1) -	2.2783e+0 (5.42e-1)
BT2ND	2	30	1.0420e+0 (1.52e-1) -	2.1593e-1 (4.63e-2) +	8.0180e-1 (9.52e-2) -	3.6940e-1 (6.07e-2)
BT3ND	2	30	2.9863e+0 (5.36e-1) -	3.1375e-1 (2.44e-1) =	3.0206e+0 (3.07e-1) -	4.5466e-1 (3.19e-1)
BT4ND	2	30	2.6691e+0 (4.04e-1) -	3.2220e-1 (2.24e-1) =	2.3993e+0 (3.83e-1) -	4.3195e-1 (2.40e-1)
BT5ND	2	30	3.2634e+0 (1.95e-1) -	1.5749e+0 (2.63e-1) +	3.4708e+0 (2.69e-1) -	2.1045e+0 (2.98e-1)
BT6ND	2	30	1.7098e+0 (6.22e-1) -	3.7629e-1 (1.25e-1) =	2.8445e-1 (6.53e-2) =	3.9662e-1 (1.56e-1)
BT7ND	2	30	7.4348e-1 (4.43e-1) -	4.7945e-1 (2.52e-1) =	4.0315e-1 (1.04e-1) =	3.9032e-1 (1.58e-1)
BT8ND	2	30	4.3367e+0 (8.93e-1) -	4.1427e-1 (3.21e-1) =	2.2907e+0 (6.60e-1) -	3.6586e-1 (9.03e-2)
+/-/=			0/8/0	2/0/6	0/6/2	

B. *Optimization Algorithms*

In this research four optimization algorithms are selected to compare the performance of the proposed dynamic re-sampling method. Therefore, four optimization algorithms are selected for a fair comparison. In this subsection these algorithms explained briefly especially the distinguish properties of the algorithms.

Particle swarm optimization with a balanceable fitness estimation (NMPSO [1]):

NMPSO is a PSO-based multiobjective optimization algorithm. Initially the algorithm begins with the randomly assigned -position- members. Then their objective values are evaluated and based on the domination idea the global best member and the personal best member which is the same position at the first iteration. Then the position and velocities are updated, and personal best members updated. By using the reference vectors and the perpendicular distance to the reference vectors the population is updated to the next iteration.

A multiobjective evolutionary algorithm based on decomposition (MOEAD [2]):

In this research two decomposition-based algorithms are applied to the problems. The best known is the MOEAD algorithm which is the first algorithm to propose decomposition as the selection operator in the genetic operator. In the algorithm crossover and mutation is applied to get a set of new population from SBX crossover and polynomial mutation methods. In addition, the neighbourhood matrix with the weight vector for the reference points are recorded and updated at each iteration. Based on the neighbouring matrix and weight vectors, the Tcbycheff method is applied as decomposition -aggregation- method. If a solution is better than it is survived to the next generation.

Decomposition-based multiobjective evolutionary algorithm with the e-constraint framework (DMOEAec [3]):

DMOEAeC is a decomposition-based optimization algorithm like MOEAD. In MOEAD the problem is divided into many subproblems by using the scalarizing function. However, in DMOEAeC only one objective is selected and other objectives as the constraint. In DMOEAeC, an upper bound vectors are generated with dividing objective axis into many equal spaces. Then solution-to-subproblem matching is applied which is based on calculating the Euclidean distance and minimum distance to certain subproblem to solution is matched. Finally with the dynamic resource allocation so that computational effort allocation to different subproblems with different difficulties.

Table 3. IGD values for the proposed method for a = 4

Problem	M	D	NMPSO	DMOEAeC	MOEAD	NSGAII
BT1ND	2	30	3.1170e+0 (2.64e-1) -	1.7740e+0 (8.60e-1) +	3.5920e+0 (1.81e-1) -	2.0915e+0 (3.24e-1)
BT2ND	2	30	9.1336e-1 (1.26e-1) -	3.2365e-1 (1.39e-1) =	6.8339e-1 (3.87e-2) -	3.7746e-1 (7.92e-2)
BT3ND	2	30	2.2321e+0 (6.05e-1) -	5.1183e-1 (5.00e-1) =	2.2177e+0 (7.94e-1) -	3.3765e-1 (2.23e-1)
BT4ND	2	30	2.2881e+0 (5.49e-1) -	2.0526e-1 (8.38e-2) =	2.0057e+0 (3.89e-1) -	2.2339e-1 (1.23e-1)
BT5ND	2	30	3.0749e+0 (2.97e-1) -	1.5188e+0 (3.74e-1) +	3.4555e+0 (2.44e-1) -	1.9475e+0 (4.54e-1)
BT6ND	2	30	1.4672e+0 (7.64e-1) -	3.6664e-1 (9.99e-2) =	3.3627e-1 (6.53e-2) =	3.9591e-1 (1.28e-1)
BT7ND	2	30	4.9460e-1 (3.11e-1) =	5.7539e-1 (1.45e-1) -	4.9973e-1 (1.56e-1) -	2.9319e-1 (1.19e-1)
BT8ND	2	30	3.4343e+0 (1.24e+0) -	5.7170e-1 (3.55e-1) -	1.8182e+0 (6.20e-1) -	3.1723e-1 (1.73e-1)
+/-/=			0/7/1	2/2/4	0/7/1	

Nondominated Sorting Genetic Algorithm (NSGA-II [4]):

Nondominated sorting is the well-known method to categorize the solution candidates with respect to the dominance idea so that the population is divided into different fronts. In NSGA-II algorithm the best members beginning with the first front survives to the next generation. For the remaining members the crowding distance is calculated, and more sparse members are selected to be in the next generation. Since this algorithm presents good results especially for two and three objective problems, it is selected to compared with other algorithms.

C. Benchmark Problems and Performance Measurements

The benchmark problems are the function who are defined to compare and test the optimization algorithms since their solution is known it is easy to compare the algorithms. However, the simple problems are not help to distinguish the performance of the optimization algorithms since all the algorithms may solve the problems. For this reason, a new set of benchmark problems proposes in the literature. Among them in [5] a set of problems called BT are proposed the difference of this algorithm is to use biases on the objective functions make them harder to solve. Unlike the single objective optimization algorithms, the solution of the Multiobjective optimizations is not a single value but a set of value that succeeded each objective. Since it is not easy to compare these sets, the functions are defined for performance measurements that called metrics. In this study to get the numerical value for how the solutions are close to the Pareto front is calculated from inverted generalized distance (IGD) [6]. In this metric the average of the Euclidean distance between obtained solution set and the Pareto front is calculated. The smaller value is expected for a good solution set.

III. Implementation and Results

The proposed method as dynamic re-evaluation method is integrated into four optimization algorithms are these algorithms are applied to the BT benchmark problems. Each problem has two objectives with 30-dimensional decision space. The implementations run for 100 population size and 2x105 maximum number of function evaluation. In addition, the implementations are repeated 10 times and statistical properties of the IGD metric, mean and standard deviation is reported into tables. Also, statistically rank sum test results reported on the tables. As given in Equation 2 as the parameter a gets different value the total number of additional function evaluation increases. The parameters are selected as [2,3,4,5] and corresponding additional function calculations are [4998,6998,9894,10998], respectively. Therefore maximum 5% and minimum 2.5% additional function are evaluated.

Table 4. IGD values for the proposed method for a = 5

Problem	M	D	NMPSO	DMOEAeC	MOEAD	NSGAII
BT1ND	2	30	3.0751e+0 (2.49e-1) -	1.6989e+0 (1.06e+0) =	3.6125e+0 (1.24e-1) -	1.9107e+0 (2.43e-1)
BT2ND	2	30	8.6731e-1 (8.69e-2) -	2.5084e-1 (1.26e-1) +	7.0676e-1 (5.72e-2) -	3.5848e-1 (6.16e-2)
BT3ND	2	30	1.9144e+0 (5.66e-1) -	5.4437e-1 (4.59e-1) =	2.3839e+0 (6.72e-1) -	4.1266e-1 (3.32e-1)
BT4ND	2	30	1.9250e+0 (3.46e-1) -	1.6711e-1 (8.48e-2) =	1.8389e+0 (3.96e-1) -	1.9968e-1 (8.30e-2)
BT5ND	2	30	3.0413e+0 (2.43e-1) -	1.3306e+0 (2.60e-1) +	3.3503e+0 (2.33e-1) -	1.9905e+0 (1.99e-1)
BT6ND	2	30	1.1381e+0 (6.31e-1) -	3.3843e-1 (4.72e-2) =	3.4019e-1 (4.68e-2) =	3.2993e-1 (1.16e-1)
BT7ND	2	30	5.9555e-1 (2.77e-1) -	5.6290e-1 (1.77e-1) -	4.7423e-1 (1.09e-1) -	2.4403e-1 (7.52e-2)
BT8ND	2	30	3.5638e+0 (9.10e-1) -	4.3507e-1 (1.61e-1) =	1.7019e+0 (5.20e-1) -	3.3425e-1 (7.31e-2)
+/-/=			0/8/0	2/1/5	0/7/1	

Table 1 gives the results for a=2. Four optimization algorithms are compared, and the results showed that DMOEAeC algorithm gives be4st result among all the algorithms even NSGA-II which presents good results especially for two objective problems. However statistical rank sum test indicates that the performance of DMOEAeC and NSGA-II algorithms are similar for BT1, BT6 and BT8 problems. For other problems DMOEAeC clearly gives better results.

Table 2 gives the results for a=3. Still the best results get from the DMOEAeC algorithm. However, for this time statistically it is much like NSGA-II performance which are approximately same in BT1, BT3, BT4, BT6, BT7 and BT8 problems, only different in BT2 and BT5 problems. When the results are compared with the first table, as the number of additional calculations of function increases the performance of the result also increases.

In Table 3, the results for a=4 are presented. For this case NSGA-II gets better but still DMOEAeC is the best algorithm for this case although statistically they are much closer to each other. When these results are compared with the first and second tables it is possible to comment that more than half of the problems in Table 3 the better results can be found than other previous tables.

Table 4 gives the higher number of additional calculations for a=5. The results support the results obtained in Table 3 so that DMOEAeC gives better or similar results with the NSGA-II algorithm. The other algorithms fall behind these two algorithms. The results in Table 4 can be compared with other tables and it can be concluded that the more than half of the benchmark problems the proposed dynamic method gives better results with relatively lower number of additional function evaluations.

IV. CONCLUSION

The aim of this research is to present a new method based on an oscillation/sinusoidal signal. The idea is to change sample size with respect to the iteration based on the maximum number of iterations. The proposed dynamic re-sampling idea is discussed on four different optimization algorithms and eight benchmark problems. Also, four different parameters the implementations are repeated. The results indicate that among all optimization algorithms DMOEAeC gives the best result with te NSGA-II algorithm. The proposed method calculates additional functions evaluations however only 2.5-5% of the total function evaluations which is relatively small. In addition, as the parameter increases the impact to the overall performance of the solution also increases.

REFERENCES

[1] Q. Lin, S. Liu, Q. Zhu, C. Tang, R. Song, J. Chen, C. A. Coello Coello, K. Wong, and J. Zhang, “Particle swarm

optimization with a balanceable fitness estimation for many-objective optimization problems," IEEE Transactions on Evolutionary Computation, vol. 22, no. 1, pp. 32-46, 2018.

[2] Q. Zhang and H. Li, "MOEA/D: A multiobjective evolutionary algorithm based on decomposition," IEEE Transactions on Evolutionary Computation, vol. 11, no. 6, pp. 712-731, 2007.

[3] J. Chen, J. Li, and B. Xin, "DMOEA-eC: Decomposition-based multiobjective evolutionary algorithm with the e-constraint framework," IEEE Transactions on Evolutionary Computation, vol. 21, no. 5, pp. 714-730, 2017.

[4] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," IEEE Transactions on Evolutionary Computation, vol. 6, no. 2, pp. 182-197, 2002.

[5] H. Li, Q. Zhang, J. Deng, "Biased Multiobjective Optimization and Decomposition Algorithm,"IEEE Transactions on Cybernetics, vol. 47, no. 1, pp. 52-66, 2017.

[6] H. Ishibuchi, H. Masuda, Y. Tanigaki, Y. Nojima, "Modified Distance Calculation in Generational Distance and Inverted Generational Distance." In António Gaspar-Cunha, Carlos Henggeler Antunes, Carlos A. Coello Coello (eds.), Evolutionary Multi-criterion Optimization, EMO 2015 Part I, volume 9018 of Lecture Notes in Computer Science, pp. 110-125. Springer, Heidelberg, Germany, 2015.

[7] L. Painton, Laura, U. Diwekar, "Stochastic annealing for synthesis under uncertainty," Eur. J. Oper. Res., vol. 83, no. 3, pp. 489–502, 1995.

[8] J. Branke, S. Meisel, C. Schmidt, "Simulated annealing in the presence of noise," J. Heuristics, vol. 14, no. 6, pp. 627–654, 2008.

[9] S.B. Gelfand, S.K. Mitter, "Simulated annealing withnoisy or imprecise energy measurements," J. Optim.Theory Appl., vol. 62, no. 1, pp. 49–62, 1989.