The Effect of Initialization in MOEAs

Ruilong Liu*
rl611@exeter.ac.uk
Department of Computer Science, University of Exeter
Exeter, UK

ABSTRACT

The allocation of computational budget between initialization and search is an important consideration in multi-objective evolutionary algorithms (MOEAs). Initialization refers to the process of generating the initial set of solutions that will be used as the starting point for the optimization process, while search refers to the process of iteratively improving the solutions over time. The quality of the initial solutions can have a significant impact on the performance of the optimization algorithm, as it determines the direction and scope of the search. In this report, we investigate the effect of varying the proportion of the computational budget allocated to initialization versus search for MOEAs. We perform experimental studies using a range of Initialization methods and analyze the results to determine the impact of the allocation of budget on the performance of the optimization algorithm. My findings provide ideas into the trade-offs between initialization and search in MOEAs and can inform the design of optimization algorithms and the allocation of computational resources.

CCS CONCEPTS

• **Optimization**; • **Evolutionary algorithm**; • **Initialization**; • **Computational budget**;

KEYWORDS

multi-objective evolutionary algorithms (MOEAs), initialization, computational budget, search, optimization performance

1 INTRODUCTION

1.1 General Introduction

Initialization is a crucial step in the optimization process, as it determines the starting point for the search. In multi-objective evolutionary algorithms (MOEAs), the initial set of solutions, also known as the initial search population, plays a key role in the optimization process, as it determines the search space that will be explored. The quality of the initial solutions can have a significant impact on the performance of the optimization algorithm, as it determines the direction and scope of the search. So it is important to consider the trade-offs between the computational budget allocated to initialization versus search. In this project, we aim to explore the research question: how much of the computational budget should be spent on initialisation for MOEAs? We will perform experimental studies using a range of optimization problems and varying the proportion of the computational budget allocated to initialization. We will then analyze the results to determine the impact of the allocation of budget on the performance of the optimization algorithm.

1.2 Review of the Literature

There have been several studies that have investigated the impact of initialization on the performance of MOEAs. For example, Li et al. (2012) [2] explored the effect of varying the computational budget for initialization on the performance of the non-dominated sorting genetic algorithm II (NSGA-II). They found that increasing the computational budget for initialization led to better initial solutions and improved the quality of the final solutions. However, they also found that the improvement in solution quality diminished as the computational budget for initialization increased.

Other studies have also found that the computational budget for initialization can have a significant impact on the performance of MOEAs. For example, Kaur et al. (2018) [1] explored the effect of varying the computational budget for initialization on the performance of the strength Pareto evolutionary algorithm 2 (SPEA2). They found that increasing the computational budget for initialization led to better initial solutions and improved the quality of the final solutions.

There have also been studies that have found that the computational budget for initialization may not have a significant effect on the performance of MOEAs. For example, Saremi et al. (2018) [3] explored the effect of varying the computational budget for initialization on the performance of the multi-objective particle swarm optimization (MOPSO) algorithm. They found that increasing the computational budget for initialization did not have a significant effect on the quality of the final solutions.

Overall, the existing literature suggests that the impact of initialization on the performance of MOEAs can be significant, but it can also depend on the specific optimization problem and the efficiency of the optimization algorithm. Further research is needed to better understand the trade-offs between computational cost and solution quality in initialization for MOEAs.

2 RESEARCH AND EXPERIMENT

2.1 Experiment Environment

The experiment is executed in Windows 11, python3.9, Anaconda Navigator, Jupyter Notebook 6.4.5.

2.2 Experiment Design

We construct two functions f1, f2 to maximize, because it is easy to show the result on the figure and also we can generate larger size of population on PC. In this algorithm, we initially generate 100000 of the initial individuals.I write a evolutionary algorithm on my own to run the process, and another evolutionary algorithm is done by deap:(doc:https://deap.readthedocs.io/en/master/tutorials/basic/part2.html). For initialization, there are 3 methods to improve: enlarge the initial population, use another EA to generate the initial population, use sorted arrays to select the N best ones.

In the first part of the experiment, these individuals will experience a randomly generating methods to generate them in a number we set. Then they participate into the evolutionary algorithm with tournament, cross over and mutation in a fixed number of round. We set the addition of number from generating the initial set and the round of evolution multiply 10 as a fixed value. We will adjust the number to see if there is any difference under different parameters.

In the second part, we set the round of evolutionary algorithm as a fixed number and adjust the generating parameters in the initial set in order to change the total budget to see the difference.

In the third part, we will see the influence on different methods of initialization.

The first experiment focus on the allocation influence and next two foucus on the over sampling possibilities on initialization of the MOEAs.

All the experiment code and data can be accessed on github: https://github.com/DuguMingyue/COMM510_CA.

2.3 Result Evaluation

In the display of result, it is shown as a scatter figure. The figure contains the Pareto front generated from the solution and the objective functions. We use 7 approaches to evaluate the final result:

- Pareto Optimality: One way to evaluate the performance of a multi-objective optimization algorithm is to determine whether it produces solutions that are Pareto optimal. A solution is Pareto optimal if there is no other solution that can improve one objective without degrading one or more of the other objectives.
- Hypervolume: The hypervolume of a set of points in a multi-objective optimization problem is the volume of the region dominated by these points. It can be used as a measure of the quality of the solutions produced by a multi-objective optimization algorithm.
- Inverted Generational Distance (IGD): The inverted generational distance (IGD) is a measure of the distance between a set of points and the Pareto optimal front. It can be used to evaluate the performance of a multi-objective optimization algorithm by comparing the IGD of the solutions it produces to the IGD of the Pareto optimal front.
- Spacing: The spacing of a set of points in a multi-objective optimization problem is a measure of the diversity of the solutions. It can be used as a measure of the quality of the solutions produced by a multi-objective optimization algorithm.
- Spread: The spread of a set of points in a multi-objective optimization problem is a measure of the distance between the points and the centroid of the set. It can be used as a measure of the quality of the solutions produced by a multi-objective optimization algorithm.
- Mean Objective Vector Difference (MOVD): The mean objective vector difference (MOVD) is a measure of the difference between a set of points and the Pareto optimal front. It can be used to evaluate the performance of a multi-objective optimization algorithm by comparing the MOVD of the solutions it produces to the MOVD of the Pareto optimal front.
- Inverted Generational Distance Plus (IGD+): The inverted generational distance plus (IGD+) is a measure of the distance between a set of points and the Pareto optimal front. It can be used to evaluate the performance of a multi-objective optimization algorithm by comparing the IGD+ of the solutions it produces to the IGD+ of the Pareto optimal front.

2.4 Expected Result

The computational budget for initialisation should be balanced with the budget for search in order to achieve good results. It is generally a good idea to allocate a larger budget for initialisation when the search space is large and complex, as this can help the algorithm explore a wider range of solutions and avoid getting stuck in local optima. On the other hand, if the search space is small or well-behaved, a smaller budget for initialisation may be sufficient.

The effect of "over-sampling" the initial population, or sacrificing some of the search budget for initialisation, can vary depending on the specific algorithm and problem being solved. Some algorithms may be more sensitive to the quality of the initial population, while others may be more robust. It may be worthwhile to experiment with different ratios of initialisation to search budget to see which performs best for a particular problem.

As we always evaluate MOEAs in diversity and convergence, then for part 1, I think that as the budget goes balanced. the diversity and convergence goes well than other situations. And for part 2 and 3, if we keep on adding the budget on initializing, then the diversity may act worse, and the convergence will act better. But as the diversity might be too low, then the solution set is not a good set.

3 RESULTS AND DISCUSS

3.1 Task 1

In task 1 we will compare the influence of budget allocation. Note: NG: number of generating initial population; NEA: rounds of evolution; MOVD: mean objective vector difference; SPA: spacing; HV: hypervolume; PO: pareto optimality; SPR: spread IGD: inverted generational distance; IGD+: inverted generational distance plus; TT: total budget.

We can see the parameters Hypervolume and Inverted Generational Distance in table 1. These value indicate both diversity and convergence. As the value HV larger and IGD smaller, the property is better. Obviously the best choice of this optimize problem is at when the number of initial population is 10000 to 30000 while the time of evolution is around 7000 - 9000. But consider the mean objective vector difference, spreading and spacing, the diversity exist more in initial population at 90000-70000.

When we calculate the patero optimize point(which indicate the proportion of total non-dominated solution exist in the whole set), we can see this value initially increasing as the number of evolution goes up, but it goes down after the budget is allocated too much on evolution So consider all the element I think the best solution of task 1 is the when NG = 10000 because at this point both the diversity and convergence is good.

Table 1: Different budget on Initialization and Evolution in Task 1

NG	NEA	MOVD	SPA	HV
95000	500	0.13023	0.44959	3.1554E+191
90000	1000	0.25825	0.50797	6.49575E+137
70000	3000	0.11198	0.43007	6.99259E+233
50000	5000	0.12191	0.61857	4.32314E+67
30000	7000	0.15654	0.75482	1.72978E+102
10000	9000	0.09396	0.33467	5.15722E+269
500	9500	0.09621	0.33579	1.18839E+238

PO	SPR	IGD	IGD+	TT
0.66038	16.21216	0.13023	2.35334	100000
0.62810	17.83810	0.25825	3.21255	100000
0.78652	20.67978	0.11198	2.77151	100000
0.72727	7.72953	0.12191	1.92210	100000
0.73684	25.63467	0.15654	1.51437	100000
0.71495	18.91173	0.09396	3.41624	100000
0.68900	21.80283	0.09622	1.32550	100000

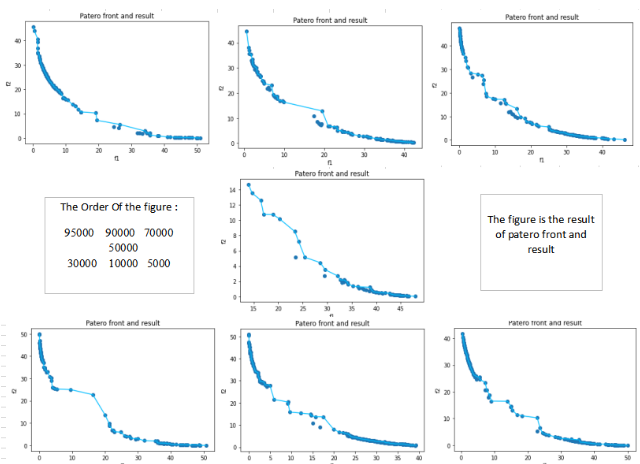

Figure 1: Result set and corresponding Pareto front in Task1

3.2 Task 2

In task two we will discuss the influence of increasing initialization budget, which will also increase the total budget.

It is obvious that initially when the budget of initialization increase, the diversity and convergence increase. But after the number of initial population is over 50000, the non-dominated solution in the solution set decrease because the evolution process is only 1000 rounds, so it is hard for the optimizer to access all the individuals. And after that as the budget increase, the diversity increase while both convergence and the quality of solution decrease. Note that when the population is over 200000, the solution goes better again because there are too many non-dominated solutions in the initial population, but it is meaningless because it has 20 times budget to the solution of 31000 total budget, but they have similar result.

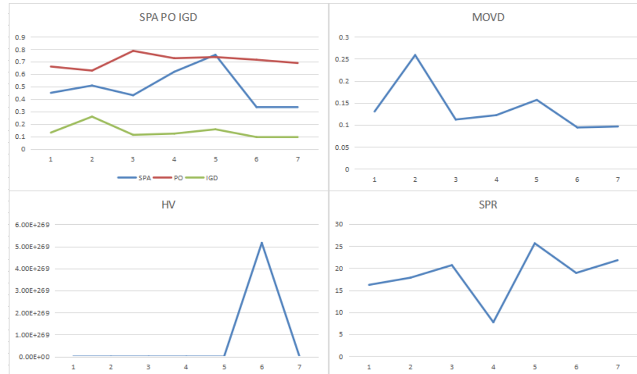

Figure 2: Evaluations of the solution set in Task1

Table 2: Different Total Budget in Task 2

NG	NEA	MOVD	SPA	HV
10000	1000	0.19344	0.44561	1.29664E+144
30000	1000	0.08434	0.52211	8.66332E+234
50000	1000	0.23923	0.59904	2.74435E+115
100000	1000	0.21075	0.58743	3.35832E+176
150000	1000	0.09150	0.42579	1.67882E+235
200000	1000	0.15006	0.39685	5.17887E+243
500000	1000	0.15555	0.39210	3.59071E+179

PO	SPR	IGD	IGD+	TT
0.60870	19.36418	0.19344	1.63313	20000
0.73077	8.59712	0.08434	1.51279	40000
0.64063	23.47306	0.23923	1.94536	60000
0.68056	18.52478	0.21075	3.06379	110000
0.72671	12.48252	0.09150	1.01570	160000
0.71508	9.70482	0.15006	2.82837	210000
0.70833	11.02401	0.15555	2.20021	510000

Table 3: Different Methods in Task 3

Methods	MOVD	SPA	HV
Random	0.25960	0.63343	3.04562E+148
Sort	0.13479	0.49409	1.19897E+185
Deap	0.25960	0.63343	3.04562E+148

PO	SPR	IGD	IGD+
0.63077	20.20094	0.25960	2.14068
0.69784	16.38085	0.13479	2.83942
0.75111	17.31053	0.08094	2.66839

3.3 Task 3

In this task we will explore the difference between different initialization methods. All methods use the parameter NG = 30000 and NEA = 1000. They all have a same initial population.

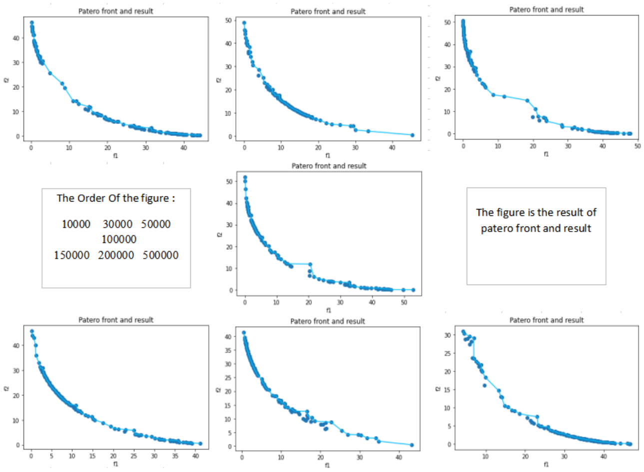

Figure 3: Result set and corresponding Pareto front in Task2

Figure 4: Evaluations of the solution set in Task2

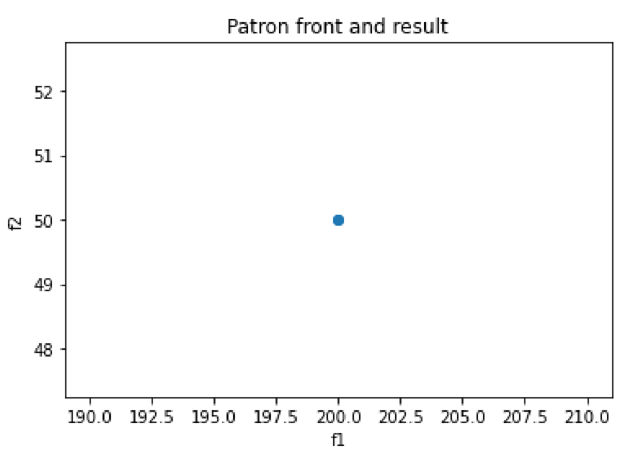

Figure 5: Over-sampling in Task3

From the table we can see that for this problem, the initialization methods order is: Tools better than sort and search better than randomly generate. But if we allocate too much budget, it will exist over-sampling.

4 ANALYSIS

We mentioned three topics in the review of literature. Li [2] thinks that the budget added on the initialization will increase the result and it will decrease when the budget keep increasing. This topic is prove by task 2. In task 2, the performance of the system increase when the cost of initial population is less than 50000 and it just decrease after that.

Saremi[3], however, had given the opinion that it will not effect the result significantly. Actually in the trade off of searching and initialization in task 1, we can see that different parameters do not have a very obvious trend of being better or worse. Because the solution of a multi-objective problem is evaluated and used in different situations. For example, if we want to arrange a timetable to lectures in the university, we just have to find a few optimal solutions in the population. In this case we need the convergence. Another case is that we want to find the best fit individuals in 50 different flowers under the same species, then the diversity is the priority. Different trade-off will given to different real problems. So when a problem is considering both the two criterion, then the trade-off is not too important in a proper range which is relatively balanced to the weight matrix of the different objective. The only problem we need to consider is to avoid extreme situation.

In task 3, it is obvious that using searching methods to initialize the population is better than random. However in the process of experiment, I found that if we search the initial population too much, the final result will display as a single point or like a small circle. This is over-sampling as the initial population is filled with the best fit individuals, which also meet my initial opinion.

5 CONCLUSION

In this report, I verify the literature review and my own ideas with three small experiments, which can still be improved. For example, I can use more algorithms to compare, and I can run more times of data to get the average. I also meet some difficulties on building evaluation functions of epsilon indicator. Though initially I want to use 8 fields of evaluation, I abandoned it.

There are still several directions to be explored in this field: how to generate initial populations in an efficient manner, while still achieving a good balance of diversity and quality. This could involve developing new initialization techniques or adapting existing ones to better suit the needs of MOEAs. Another direction that could be explored is the development of initialization methods that can adapt to the specific characteristics of the optimization problem being solved. This could involve using machine learning techniques to learn patterns in the objective function and generate more informed initial populations.We can also investigate the usage of hybrid approaches that combine MOEAs with other optimization techniques, such as local search or gradient-based methods, to improve the efficiency of the initialization process.

Many real-world optimization problems are subject to constraints, such as physical or resource limits. We need to develop initialization methods that are able to incorporate these constraints and generate feasible initial populations could be an important area of research.

REFERENCES

[1] Kaur, H., Singh, A., Singh, A. (2018). A hybrid strength Pareto evolutionary algorithm for multi-objective optimization. Engineering Applications of Artificial Intelligence, 72, 174-185.

[2] Li, Y., Zhang, Q., Suganthan, P. N. (2012). A study on the impact of initialization on the performance of multi-objective evolutionary algorithms. Evolutionary Computation, 20(2), 187-223.

[3] Saremi, S., Talatahari, S., Deb, K. (2018). A comprehensive study of multi-objective particle swarm optimization algorithms. Soft Computing, 22(7), 2033-2054.

[4] Deb, K., Pratap, A., Agarwal, S., Meyarivan, T. (2002). A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II. In: Proceedings of the International Conference on Parallel Problem Solving from Nature, pp. 849-858.

[5] Zitzler, E., Deb, K., Thiele, L. (2000). Comparison of Multiobjective Evolutionary Algorithms: Empirical Results. In: Evolutionary Computation, 8(2), pp. 173-195.

[6] Coello Coello, C.A., Pulido, G.T., Lechuga, M.S. (2004). Handling Multiple Objectives with Particle Swarm Optimization. In: IEEE Transactions on Evolutionary Computation, 8(3), pp. 256-279.

[7] Zhan, R., Gu, X., Liu, X. (2014). Hybridizing Particle Swarm Optimization with Genetic Algorithm for Multiobjective Optimization. In: IEEE Transactions on Evolutionary Computation, 18(4), pp. 557-570.

[8] Zhang, Q., Li, H., Suganthan, P.N. (2010). MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. In: IEEE Transactions on Evolutionary Computation, 11(6), pp. 712-731.