

**Bachelor Thesis Project (BTP) Report**

**Shivam Pandey**

**Roll Number: 160010003**

**Month of Submission: May 2020**

**Supervisor: Prof. Somenath Biswas, Computer Science and Engineering**

*This report is submitted towards partial fulfillment of the requirements for the award of the Bachelor of Technology (B.Tech) degree in Computer Science and Engineering at the Indian Institute of Technology Goa.*

**Acknowledgements**

**Abstract**

**Study of Multi-Objective Optimization using Evolutionary Algorithms**

*Shivam Pandey*

**Introduction & Problem Statement**

Optimization is a method of finding and comparing feasible solutions, till no better solution can be found. A solution is termed as good or bad on the basis of an objective, which in real life can be the cost of manufacturing, fuel efficiency of a car, profit earned in business, etc. In real life, we usually come across optimization problems having several objectives. A multi-objective optimization problem (MOOP) entails more than one objective to be optimized, often conflicting objectives, and has a pareto-optimal set of solutions. Pareto-optimal set is defined as the *non-dominated***1** set of the entire feasible search space. Multi-Objective Evolutionary Algorithms (MOEAs) seek to approximate the *pareto-front***2** of the MOOP, by evolving a population of solutions, inspired from the biological evolution using selection, reproduction, recombination and mutation. Classical optimization methods can at best find one solution in one simulation run, thereby making those methods inconvenient to solve MOOPs. Evolutionary approaches to MOOPs, on the other hand, are capable of searching for multiple optimal solutions concurrently in a single run, due to their population-approach. This ability of an evolutionary algorithm (EA) to find multiple optimal solutions in one simulation run makes EAs unique in solving MOOPs.

My Bachelor Thesis Project (BTP) work focuses on studying and exploring existing MOEAs like Non-dominated Sorting Genetic Algorithm-II (NSGA-II) [2], and formulate possible improvement ideas along with combining the best ideas from existing MOEAs to propose a better algorithm, in terms of *quality of solutions***3**, running time of the algorithm, or preferably both.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**1***Among a set of solutions P, the non-dominated set of solutions P`, are those that are not dominated by any member of the set P.*

**2***The curve formed in the objective space by joining these pareto-optimal solutions is known as a* ***pareto front****.*

**3***The MOEA converges as* ***close*** *as possible to the pareto front and maintains a* ***good******diversity*** *amongst the solutions.*

**Literature Review**

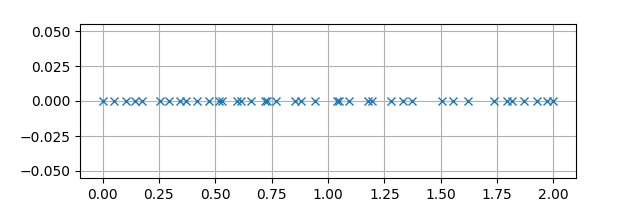
[1] In the initial phase of my work, the formulation of a MOOP was studied. Most MOEAs use the concept of *dominance* and *pareto-optimality* in their search. A solution *x* is said to dominate the other solution *y*, if both the following conditions are true:

• The solution *x* is no worse than *y* in all objectives.

• The solution *x* is strictly better than *y* in at least one objective.

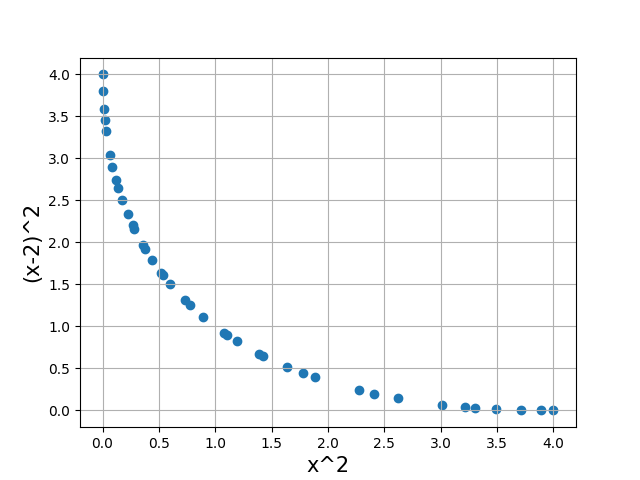
After studying the terminologies under the literature of EAs, the general structure of a genetic algorithm (GA) was studied and how they are implemented to solve an optimization problem, may it be a single-objective problem or a multi-objective problem.

[2]An existing elitist MOEA, particularly known as NSGA-II, with computational complexity O (*M N*2) (here, M = number of objectives, N = population size), was studied and implemented on a two-objective optimization benchmark problem: Schaffer’s study (SCH)[3] (here, we took a minimization problem). The two objective functions considered for the minimization problem were *x* 2 and (*x* − 2)2 , where the decision variable space : *x* ∈ (−∞, +∞). It is evident that the optimal set of solutions for this problem would be *x* ∈ [0, 2], hence we would expect the algorithm to converge to the pareto-front (non-dominated solutions) in the objective space after sufficient number of generations are evolved. In fig.1 and fig.2 shown below, the NSGA-II was run for about 3000 generations on a population of 40 individuals.

****

**Fig. 1 Visualization of Decision Variable Space (or Data Space)**

In Fig.1, each of the crosses denote one individual in the final population the algorithm outputs, which is evidently close to the optimal solution set, i.e. *x* ∈ [0, 2].

****

**Fig.2 Visualization of Objective Space**

In Fig. 2, a reasonably good approximation to the pareto-front was obtained in the objective space, along with reasonable diversity which is expected from NSGA-II. Hence, the run of NSGA-II was successful on this particular MOOP. With this successful execution of NSGA-II, a concrete understanding was developed about the working of MOEAs.

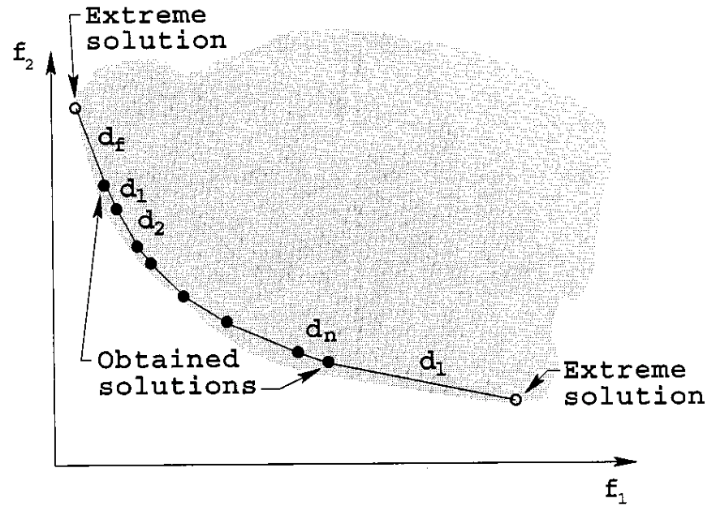
**Experimental Analysis & Improvement Idea**

The major experimental analysis throughout this survey of NSGA-II lies at considering the diversity of solutions in both, objective space and data space. In MOEAs, one of the most important factor which contributes to the obtained set of non-dominated solutions to be called “good” is *diversity* **4**. It should be noted that NSGA-II [2] does not take into account the diversity in *data space*, but only in *objective space*. This motivated further study to consider the diversity (individually and simultaneously) in both, data space and objective space, and compare the results with traditional NSGA-II. The expectation is that if we perform the evolution of each generation by taking into account the crowding distances in both, data space and objective space, then, we might obtain a good amount of uniformity in both the spaces.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

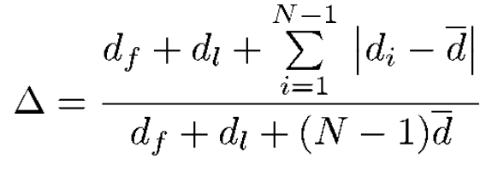
**4*Diversity*** *signifies how uniformly the solutions are spread throughout the search space.*

Before we move towards the implementation of the algorithm, we define the metric which will be used to measure the diversity in a given search space. With the use of this metric and graphical visualizations acquired from simulation run, we will give a detailed analysis of the results obtained.



**Fig.3 Diversity Metric Δ (courtesy [2])**

The diversity metric Δ calculates the magnitude of spread achieved in the obtained solutions. The following metric [2] is used to measure the non-uniformity in the distribution:



It should be noted that for the most uniformly spread out set of non-dominated solutions, the numerator of Δ would become zero, and thus, Δ = 0. For any other distribution, we will obtain Δ > 0. Please note that Δ will always be non-negative, which is self-evident from its mathematical expression. So, it is to be realized that the more diverse (uniformly spread) the set of solutions will be, the closer the value of Δ will be to zero. Hence, smaller the value of Δ, more diverse is the solution set.

Now, to run our algorithm, we shall consider a slightly more complex MOOP, which is called as the *Min-Ex* problem. It is a two-objective, two-variable minimization problem, defined below:

**Min-Ex Problem**

*Minimize ƒ*1 (*x*1 , *x*2) = *x*1

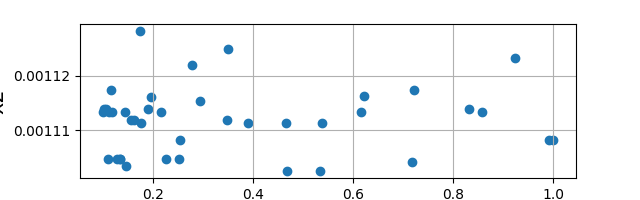
*Minimize ƒ*2 (*x*1 , *x*2) =(1 + *x*2) / *x*1

*subject to* 0.1 ≤ *x*1 ≤ 1 & 0 ≤ *x*2 ≤ 5

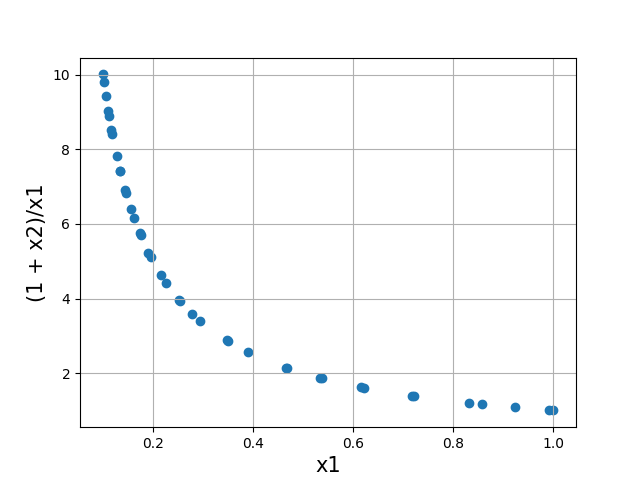
**Simulation Run**

We will run three different algorithms on the *Min-Ex* problem, which includes the traditional NSGA-II and the other two algorithms, which are slightly modified versions of the original NSGA-II. We define the following three names for three different algorithms to be implemented for experimentation:

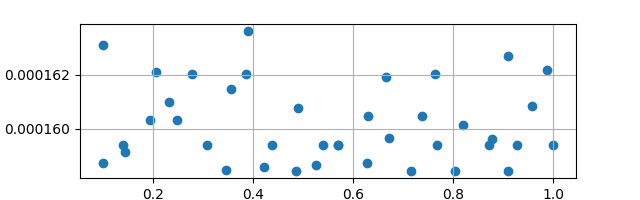
* *Version-I* 🡪 Traditional NSGA-II algorithm.
* *Version-II 🡪* Modified algorithm, which will evaluate the crowding distances only in *data space.*
* *Version-III 🡪* Modified algorithm, which will evaluate the crowding distances in both, *data space* and *objective space*.



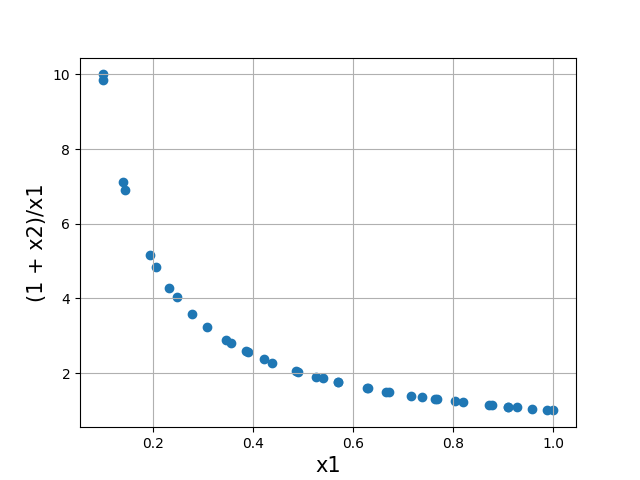
**Fig.4A Data Space (Run of *Version-I*)**



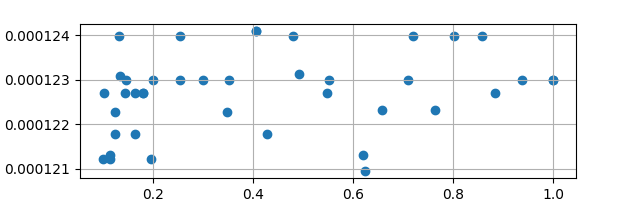
**Fig.4B Objective Space (Run of *Version-I*)**



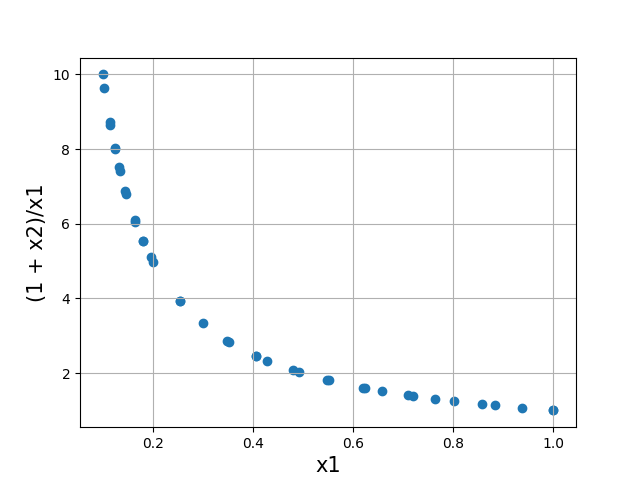
**Fig.5A Data Space (Run of *Version-II*)**



**Fig.5B Objective Space (Run of *Version-II*)**



**Fig.6A Data Space (Run of *Version-III*)**



**Fig.6B Objective Space (Run of *Version-III*)**

**Discussion and Results**

We ran each version of the algorithm five times over the *Min-Ex* problem and then calculated the mean diversity metric in both the spaces for each version. Please note that we have used the population size of 40, mutation rate of 0.2 and evolved for 3000 generations in a single simulation run.

The following table shows the values of *mean diversity metric in objective space* (Δo) and *mean diversity metric in data space* (Δd) :

**Table 1: Mean diversity metric in data space and objective space when run on *Min-Ex* problem with a population of 40 individuals and evolving for 3000 generations.**

|  |  |  |
| --- | --- | --- |
| **Min-Ex** | **Δd** | **Δo** |
| *Version-I* | 0.9640  (Fig.4A) | 0.6971  (Fig.4B) |
| *Version-II* | 0.5556  (Fig.5A) | 1.084  (Fig.5B) |
| *Version-III* | 0.8492  (Fig.6A) | 0.8798  (Fig.6B) |

From table 1, it can be diagnosed that the diversity metrics Δd and Δo took quite promising values. Firstly, we will only discuss about the set of solutions obtained in data space from all three versions. We would have expected that the most diverse data space is obtained from the run of *Version-II,* since it takes into account the crowding distances amongst data space only. This is very much evident from Fig.5A and the value of Δd(*Version-II*). Even by visual analysis, it is clearly seen that the most uniformly spread out obtained data space is of Fig.5A , followed by Fig.6A and Fig.4A.

**References**

[1] K. Deb, Multiobjective Optimization Using Evolutionary Algorithms. Chichester, U.K.: Wiley, 2001.

[2] K. Deb, A. Pratap, S. Agarwal and T. Meyarivan, “A fast and elitist multi-objective genetic algorithm: NSGA-II”, Proc. Parallel Problem Solving from Nature VI, 2000

[3] J. D. Schaffer, “Multiple objective optimization with vector evaluated genetic algorithms,” in Proceedings of the First International Conference on Genetic Algorithms, J. J. Grefensttete, Ed. Hillsdale, NJ: Lawrence Erlbaum, 1987, pp. 93–100.