Intelligent Automation & Soft Computing
DOI:10.32604/iasc.2022.023510
Article

Tech Science Press

On NSGA-II and NSGA-III in Portfolio Management

Mahmoud Awad[1], Mohamed Abouhawwash[1,2,*] and H. N. Agiza[1]

[1]Department of Mathematics, Faculty of Science, Mansoura University, Mansoura, 35516, Egypt
[2]Department of Computational Mathematics, Science, and Engineering (CMSE), Michigan State University, East Lansing, 48824, USA
*Corresponding Author: Mohamed Abouhawwash. Email: abouhaww@msu.edu
Received: 11 September 2021; Accepted: 27 October 2021

Abstract: To solve single and multi-objective optimization problems, evolutionary algorithms have been created. We use the non-dominated sorting genetic algorithm (NSGA-II) to find the Pareto front in a two-objective portfolio query, and its extended variant NSGA-III to find the Pareto front in a three-objective portfolio problem, in this article. Furthermore, in both portfolio problems, we quantify the Karush-Kuhn-Tucker Proximity Measure (KKTPM) for each generation to determine how far we are from the effective front and to provide knowledge about the Pareto optimal solution. In the portfolio problem, looking for the optimal set of stock or assets that maximizes the mean return and minimizes the risk factor. In our numerical results, we used the NSGA-II for the portfolio problem with two objective functions and find the Pareto front. After that, we use Karush-Kuhn-Tucker Proximity Measure and find that the minimum KKT error metric goes to zero with the first few generations, which means at least one solution converges to the efficient front within a few generations. The other portfolio problem consists of three objective functions used NSGA-III to find the Pareto front and we use Karush-Kuhn-Tucker Proximity Measure and find that The minimum KKT error metric goes to zero with the first few generations, which means at least one solution converges to the efficient front within a few generations. Also, the maximum KKTPM metric values don't show any convergence until the last generation. Finally, NSGA-II is effective only for two objective functions, and NSGA-III is effective only for three objective functions.

Keywords: Genetic algorithm; NSGA-II; NSGA-III; Portfolio problem

1 Introduction

Genetic algorithms (GA) have been widely employed as optimization and search methods in a variety of problem domains [1,2], including industry [3], architecture [4], and research, over the past ten years [5,6]. Their strong applicability, global outlook, and ease of usage are the key explanations for their high success rate.

Genetic algorithms are the most common algorithms for solving many real-life applications. These algorithms are inspired by natural selection. Genetic algorithms are population search algorithms, which

This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

introduced the idea of survival of the best fitness function. Most of these algorithms incorporate the genetic operation to obtain the new chromosome (solution). The basic genetic operations are selection, crossover, and mutation [7].

In the past, the portfolio optimization problem was designed to find the configuration of assets that generated the maximum expected return which was the main criterion. However, this design changed in 1952, a new variable with the expected return was introduced by Harry Markowitz that called the risk of each portfolio. Thereafter, analysts began to incorporate a risk-return trade-off in their models [8]. Harry Markowitz doesn't consider the real-world challenges as cardinality constraints, lower and upper bounds, substantial stock size, class constraint, round-lots constraint, computational power and time, pre-assignment constraint, and local-minima avoidance.

In this article, we solve the portfolio problem [9] using two genetic algorithms, NSGA-II and NSGA-III. The competing parameters in the portfolio dilemma are optimizing anticipated return and mitigating risk, also known as the Markowitz, mean-variance model [10].

1.1 Aim of the Study

This study aims to find the Pareto front for portfolio problems with two and three objective functions using the methods NSGA-II and NSGA-III which are simpler and easy to apply. Those methods can address portfolio optimization problems without simplification and with decent results in a fair amount of time, and it has a lot of practical applications. we obtained solutions for the portfolio models using NSGA-II and NSGA-III same as theoretical solutions.

1.2 Novelty and Contributions

The main contributions of this paper are as follows:

- A genetic algorithm can be used to find the Pareto front for portfolio optimization problems which are the same as those found in other approaches.
- A genetic algorithm can handle the portfolio optimization problems without simplification and with decent results in a fair amount of time.
- The two cases studied were presented to prove the applicability of the genetic algorithm.
- The framework of NSGA-II and NSGA-III are elaborated in algorithms 1 and algorithm 2.

1.3 Study Structure

In the following part of the article, we first give a literature review of NSGA-II, NSGA-III, and portfolio optimization problems in the second section. After that, we explain the concept of the NSGA-II method. The NSGA-III is discussed in the fourth section. The Karush-Kuhn-Tucker proximity measure (KKTPM) is analyzed in section five for multi-objective optimization problems [11,12]. In section six, evolutionary algorithms are used to solve a portfolio dilemma. Finally, in section seven, we bring the article to a close.

2 Literature Review

2.1 NSGA-II

There are many studies in NSGA-II. Deb et al. [13] have proposed NSGA-II which improves the iterative convergence rate while ensures population diversity by employing the fast non-dominated sorting approach. Kodali et al. [14] used NSGA-II to solve a problem that involves two objectives, four constraints, and ten decision variables of the grinding machining operation. Wang et al. [15] have used improved NSGA-II to solve multi-objective optimization of turbomachinery.

2.2 NSGA-III

There are some studies in NSGA-III. Deb and Jian [16] have proposed the first algorithm of NSGA-III to solve multi-objective optimization problems. Mkaouer et al. [17] are used NSGA-III to solve many-objective software remodularization. Zhu et al. [18] were studied an improved NSGA-III algorithm for feature selection used in intrusion detection. Yi et al. [19] were studied the behavior of crossover operators in NSGA-III for large-scale optimization problems.

2.3 Portfolio Problem

Markowitz [20] was proposed the portfolio problem, that it is looking for the expected mean-return is maximized (profit), and the risk is minimized. The factor in measuring risk is the variance of the portfolio return; the smaller the variance lower will be the risk. Michaud [21] has found that mean-variance theory has some limitations because asset volatility is required for constructing the model, and determining an asset's future volatility is challenging in practice. Momentum investment is a well-known quantitative investment strategy. Hong and Stein [22] show that this strategy, the momentum effect is used to reveal the price stickiness of stocks over a certain period; this information is then used to predict price trends and make investment decisions.

3 NSGA-II or Elitist Non-Dominated Sorting GA

The NSGA-II protocols [23] is the most used EMO procedure for finding multiple Pareto-optimal solutions in a multi-objective optimization problem, and it has the following features:

It employs three principles: 1. an apparent diversity-preserving mechanism; 2. an elitist principle; and 3. non-dominated alternatives are stressed.

Consider a community of size N, with parent and offspring populations P_t and Q_t. Making $R_t = P_t \cup Q_t$ by combining offspring and parent populations in the first process. R_t should be non-dominated sorted to distinguish various fronts F_i, $i = 1, 2, \ldots,$ etc. Set a new population $P_{t+1} = \varnothing$ and a counter $i = 1$ before $|P_{t+1}| + |F_i| < N$ is reached. $P_{t+1} = P_{t+1} \cup F_i$ and $i = i + 1$ are the steps to take. Then use the crowding-sort $(F_i, <_C)$ protocol to get the most distributed $(N - |P_{t+1}|)$ solutions by sorting the crowding distance values in the sort from F_i to P_{t+1}. To build an offspring population Q_{t+1} from P_{t+1}, use the crowded tournament array, crossover, and mutation operators.

Now we demonstrate a crowded tournament collection operator. The crowded comparison operator $(<_C)$ performs a comparison between two solutions and returns the tournament's winning answer. It is assumed that every solution i has two characteristics. The community has a local crowding distance (d_i) and a non-domination rating r_i.

Definition: If all of the above assumptions are true, the crowded tournament selection operator [24] compares two solutions (solution i and another solution j), and solution i wins the tournament. If $r_i < r_j$, it implies the solution i has a higher ranking. If $r_i = r_j$ and $d_i > d_j$, the solutions are of equal level, but solution i has a shorter crowding distance than solution j.

Crowding gap; To find the estimation density of solutions around a given solution i in the community, one takes the average distance between the two solutions on each side of solution i through each of the objectives. This d_i serves as the cuboid's estimated diameter, which is calculated by using the cuboid's nearest neighbors as vertices, a process known as crowding time. For each point in the set F, calculate the crowding distance as follows (crowding type $(F_i, <_C)$: First and foremost, First, set $d_i = 0$ for each i in the set. $l = |F|$ equals the number of solutions in F. Find the ordered indices vector $I^m = sort(f_m, >)$

for each objective function $m = 1, 2, \ldots, M$, or sort in the collection in the worst order of f_m. $d_{I_1^m} = d_{I_l^m} = \infty$ or assign a significant gap to the boundary solutions for $m = 1, 2, \ldots, M$, and all other solutions $j = 2$ to $(l - 1)$, assign:

$$d_{I_j^m} = d_{I_j^m} + \frac{f_m^{\left(I_{j+1}^m\right)} - f_m^{\left(I_{j-1}^m\right)}}{f_m^{max} - f_m^{min}} \tag{1}$$

The lowest and highest objective function values are denoted by I_1 and I_l, respectively. Algorithm 1, for generation t of NSGA-II procedure [25].

Algorithm 1: NSGA-II

```
Input: Problem Size, Population size, P mutation, P crossover
Output: Children
Population ← Initialize Population (Population size, Problem Size)
Evaluate against objective functions (Population)
Fast Nondominated Sort (Population)
Selected ← Select parents by rank (Population, Population size)
Children ← Crossover and mutation (Selected, P crossover, P mutation)
While (Stop Condition())
   Evaluate against objective functions (Children)
   Union ← Merge (Population, Children)
   Fronts ← Fast Nondominated Sort(Union)
   Parents ← ∅
   Front_L ← ∅
   For ( Front_i ∈ Fronts)
      Crowding distance assignment (Front_i)
      If ( Size (Parents)+ Size (Front_i) > Population size )
      Front_L ← i
         Break()
      Else
         Parents ← Merge (Parents, Front_i)
      End
   End
   If (Size (Parents)< Population size )
      Front_L ← Sort by rank and distance()
      For (P_1 to P_Population size − Size Front_L)
         Parents ← P_i
      End
   End
   Selected ← Select parents by rank and distance (Parents, Population size)
   Population ← Children
   Children ← Crossover and mutation (Selected, P crossover, P mutation)
End
Return (Children).
```

4 An Evolutionary Many-objective Optimization Algorithm Using Reference Point Based Non-Dominated Sorting Approach (NSGA-III)

NSGA-III begins [26] with a random population of N members and a series of widely spaced M-dimensional reference points H distributed over a unit hyper-plane with standard vector ones covering the entire R_+^M field. The hyper-plane (HP) is set up in such a way that it intersects all of the objective axes at the same time. The technique of Das and Dennis [27] is used to position $H = \binom{M+p-1}{p}$ reference points on the HP with $(p+1)$ points through any boundary. They choose the population size N to be the smallest multiplied by four greater than H, with the assumption that one population member would be obtained for all reference points.

The following procedures are carried out at descent t. Following the precept of non-dominated sorting, all of the population P_t is sorted into different non-domination levels, close to how it is done in NSGA-II. The children's population Q_t is generated by using standard mutation and recombination operators on the P_t population. Since only one population member is expected to be examined for any reference point, each selection procedure in NSGA-III is unnecessary, as every selection operator would allow competition to be established between different reference points. After that, a combined population $R_t = P_t \cup Q_t$ is formed. Then, starting from the first non-dominated front, points are selected for P_{t+1} one by one until no entire solutions from a full front can be used. This restriction is also common in the NSGA-II. Assume that there is a final front that can't be fully selected as F_L. Only a few solutions from F_L that choose to be selected from P_{t+1} use a niche-preserving operator, which we'll look at later. To begin, any population unit of P_{t+1} and F_L is normalized using the current population distribution, resulting in similar values for all reference points and objective vectors. The shortest perpendicular distance $(d())$ of each population unit with a reference line generated by connecting a supplied reference point with the origin is then used to correlate each component of P_{t+1} and F_L with a specific reference point. Then, using reference points in P_{t+1}, a cautious (NS) niching technique is used to pick certain F_L components that are associated with the minimum. The (NS) niching strategy ensures that a population factor is selected for each of the provided reference points [28]. A population variable that is compared with an unrepresented comparison or an under-represented point is quickly outperformed. With a constant tension to ensure non-dominated individuals, all phase is predicted to produce one population variable that correlates with any supplied reference point near the (POF) Pareto-optimal front, assuming that the genetic difference operators (mutation and recombination) will deliver specific solutions. Algorithm 2 summarizes the algorithm, which uses widely spaced comparison points to ensure a well-distributed series of trade-off points at the end. Algorithm 2, Generation t of NSGA-III procedure:

Algorithm 2: NSGA-III Approach

Input: H structured reference points Z^s or supplied aspiration points Z^a, parent population P_t.
Output: P_{t+1}.

1. $S_t = \emptyset,\ i = 1$.
2. Q_t =Recombination and Mutation P_t.
3. $R_t = P_t \cup Q_t$.
4. $(F_1,\ F_2,\ F_3,\ \ldots) =$ Non-dominated sort R_t.
5. Repeat.
6. $S_t = S_t \cup F_i$ and $i = i + 1$.
7. Until $|S_t| \geq N$.

(Continued)

Algorithm 2 (continued)

8. The last front to be include $F_l = F_i$.
9. If $|S_t| = N$ then.
10. $P_{t+1} = S_t$, break.
11. Else.
12. $P_{t+1} = \cup_{j=1}^{l-1} F_j$.
13. Points to be chosen from $F_l : K = N - |P_{t+1}|$.
14. Normalize objectives and create a reference set Z^r: Normalize $(f^n,\ S_t, Z^r, Z^s, Z^a)$.
15. Associate each member s of S_t with a reference point: $[\pi(\text{s}), \text{d}(\text{s})]$ =Associate (S_t, Z^r); $\pi(s)$ =closest reference point, d = distance between s and $\pi(s)$.
16. Compute niche count of reference point $j \in Z^r$: $\rho_j = \sum_{s \in S_t/F_l} ((\pi(s) = j)\ ?1 : 0)$.
17. Choose K members one at a time from F_l to construct P_{t+1} : Niching $\left(K, \rho_j, \pi, d, Z^r, F_l, P_{t+1}\right)$.
18. End if.

5 Karush-Kuhn-Tucker Proximity Measure (KKTPM) for Multi-Objective Optimization

For a n-variable, M-objective optimization problem with J inequality constraints:

$$\min_{(X)} \quad \{f_1(X), f_2(X), \ldots, f_M(X)\},$$

$$\textit{Subject to} \quad g_j(X) \le 0,\ \ j = 1, 2,\ \ldots, J, \tag{2}$$

the Karush-Kuhn-Tucker optimality [29] conditions for Eq. (2) are given as follows:

$$\sum_{m=1}^{M} u_m \nabla f_m\left(X^k\right) + \sum_{j=1}^{J} u_j \nabla g_j\left(X^k\right) = 0, \tag{3}$$

$$g_j\left(X^k\right) \le 0,\ \ j = 1, 2, \ldots, J, \tag{4}$$

$$u_j g_j\left(X^k\right) \le 0,\ \ j = 1, 2, \ldots, J, \tag{5}$$

$$u_j \ge 0,\ \ j = 1, 2, \ldots, J, \tag{6}$$

$$u_m \ge 0,\ \ m = 1, 2, \ldots, M,\ \text{and } u \ne 0. \tag{7}$$

The u_m multipliers are not negative, but at least one of them cannot be empty. For the j-th inequality constraint, the parameter u_j is called Lagrange multiplier, and it is not even negative. A KKT point is a solution X^k that meets all of the above criteria. The inequality constraints $g_{J+2i-1}(X) = x_i^{(L)} - x_i \le 0$ and $g_{J+2i}(X) = x_i - x_i^{(U)} \le 0$ can be used to break inconstant the form $x_i^{(L)} \le x_i \le x_i^{(U)}$. There are some $J + 2n$ inequality limits for the previous issue of whether there are whole n pairs with particular inconstant boundaries.

The authentic analysis generated an output scalarization feature (ASF) for a given repeated (solution) X^k [29]. A matter of optimization:

$$\min_{(X)} \quad ASF(X,\ Z,\ W) = max_{m=1}^{M}\left(\frac{f_m(X) - z_m}{w_m}\right),$$

$$\textit{Subject to} \quad g_j(X) \le 0,\ \ j = 1, 2,\ \ldots, J. \tag{8}$$

The reference point $Z \in R^M$ was believed as a utopian point and the weight vector $W \in R^M$ is computed for X^k as views:

$$w_i = \frac{f_i\left(X^k\right) - z_i}{\left(\sum_{m=1}^{M} \left(f_m\left(X^k\right) - z_m\right)^2\right)^{1/2}} . \tag{9}$$

Thereafter, the KKTPM calculation process advanced for single-objective optimization problems to the ASF showed previously. So that the ASF formulation produce the objective function not differentiable, a smooth transformation of the ASF (a performance scalarization function) problem was made firstly by inserting slack variables x_{n+1} and reconstructing the initial problem as views:

$$\min \quad F(X, x_{n+1}) = x_{n+1},$$

$$\textit{Subject to} \quad \frac{f_i(X) - z_i}{w_i^k} - x_{n+1} \leq 0,\ i = 1, 2,\ \ldots, M \tag{10}$$

$$g_j(X) \leq 0,\ \ j = 1, 2,\ \ldots, J.$$

At this moment, the KKTPM optimization problem for the previous single-objective problem for $y = (X; x_{n+1})$ can be written as follows:

$$\min_{(\varepsilon_k,\ x_{n+1},\ u)} \quad \varepsilon_k + \sum_{j=1}^{J} \left(u_{M+j}\, g_j\left(X^k\right)\right)^2,$$

$$\textit{Subject to} \quad \left\| \nabla F(y) + \sum_{j=1}^{M+J} u_j \nabla G_j(y) \right\|^2 \leq \varepsilon_k,$$

$$\sum_{j=1}^{M+J} u_j G_j(y) \geq -\varepsilon_k, \tag{11}$$

$$\frac{f_j(X) - z_j}{w_j^k} - \ x_{n+1} \leq 0,\ \ j = 1, 2,\ \ldots, M,$$

$$u_j \geq 0,\ \ j = 1, 2,\ \ldots, M + J.$$

The added term in the objective function permits a penalty correlated with the violation of the complementary slackness condition. The restrictions $G_j(y)$ are given below:

$$G_j(y) = \frac{f_j(X) - z_j}{w_j^k} - x_{n+1} \leq 0,\ \ j = 1, 2,\ \ldots, M, \tag{12}$$

$$G_{M+j}(y) = g_j(X) \leq 0,\ \ j = 1, 2,\ \ldots, J. \tag{13}$$

The optimal objective value ε_k^* to the above problem corresponds to the exact KKTPM. It is observed that for feasible solutions $\varepsilon_k^* \leq 1$, hence the exact KKTPM was defined as follows:

Exact KKTPM

$$\left(X^k\right) = \begin{cases} \varepsilon_k^*, & \textit{feasible } X^k, \\ 1 + \sum_{j=1}^{J} g_j\left(X^k\right)^2, & \textit{otherwise}. \end{cases} \tag{14}$$

6 Results Section

In this section, we will solve a portfolio problem in special cases using NSGA-II in the first model and NSGA-III in the second model. After that, we will show figures for each model. But, one must know the main portfolio problem. The portfolio is a set of assets or securities $(x_1, x_2, \ldots, x_n)$ chosen to minimize the risk and maximize the expected return. The risk is measure by the variance. The problem can be written as following [30]:

$$\max \quad \sum_{i}^{n} r_i x_i \ ,$$

$$\min \quad \sum_{i,j}^{n} x_j x_i \sigma_{ij},$$

$$\textit{Subject to} \quad \sum_{i}^{n} x_i = 1,$$

$$x_i \geq 0.$$

To illustrate the mechanism of the evolutionary algorithms and KKT proximity measure using an evolutionary multi-objective (EMO) algorithm, we thought of three and two-objective Portfolio problems. NSGA-II is used to solve two objective problems, while NSGA-III is used to solve three objective problems. We use the SBX recombination operator [31,32] with $p_c = 0.9$ and $\eta_c = 30$ in every problem, as well as the polynomial mutation operator [33,34] with $p_m = 1/n$ (where n is the number of variables) and $\eta_m = 20$ in every problem. In discussions about personal models, other criteria are listed.

6.1 Model-I for Portfolio Problem

Consider the three-security problems with expected returns vector and covariance matrix [35] given by:

$$(r_1, r_2, r_3) = (0.062, 0.146, 0.128) \text{ and}$$

$$\begin{bmatrix} \sigma_1^2 & \sigma_{12} & \sigma_{13} \\ \sigma_{12} & \sigma_2^2 & \sigma_{23} \\ \sigma_{13} & \sigma_{23} & \sigma_3^2 \end{bmatrix} = \begin{bmatrix} 0.0146 & 0.0187 & 0.0145 \\ 0.0187 & 0.0854 & 0.0104 \\ 0.0145 & 0.0104 & 0.0289 \end{bmatrix}.$$

Let $X = (x_1, x_2, x_3)^T$, where x_1, x_2, x_3 are the proportions of an asset invested in the following model-I and model-II. So model-I is [19,20]

$$\textit{max} \quad Er(X) = 0.062x_1 + 0.146x_2 + 0.128x_3$$

$$\textit{min} \quad Vr(X) = 0.0146x_1^2 + 0.0854x_2^2 + 0.0289x_3^2 + 2(0.0187x_1x_2 + 0.0145x_1x_3 + 0.0104x_2x_3)$$

$$\textit{subject to} \quad x_1 + x_2 + x_3 = 1,$$

$$x_1, x_2, x_3 \geq 0.$$

Fig. 1 shows the non-dominated points for model-I. In this figure, NSGA-II runs 200 generations with 100 population sizes. The obtained solutions exactly equal the previously exact obtained solutions. One advantage of applying genetic algorithms is that we obtain many solutions in a single run. Also, Fig. 2 represents the relation between generation number and KKT proximity measure. As shown from the figures, the KKT metric reduces with increasing the number of generations.

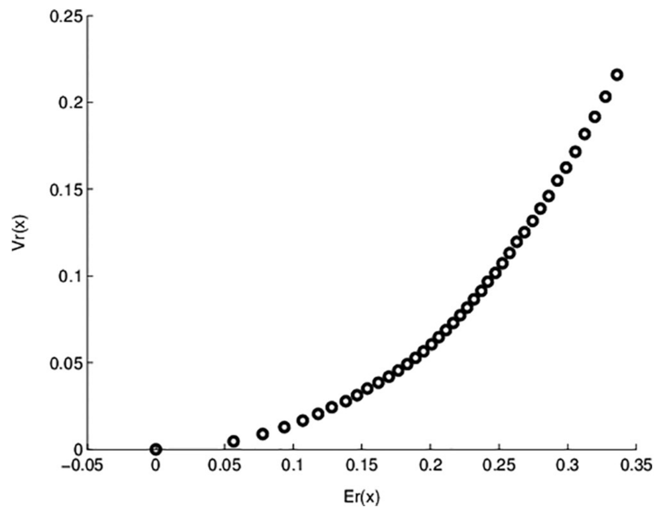

Figure 1: Pareto optimal points for model- I objective functions

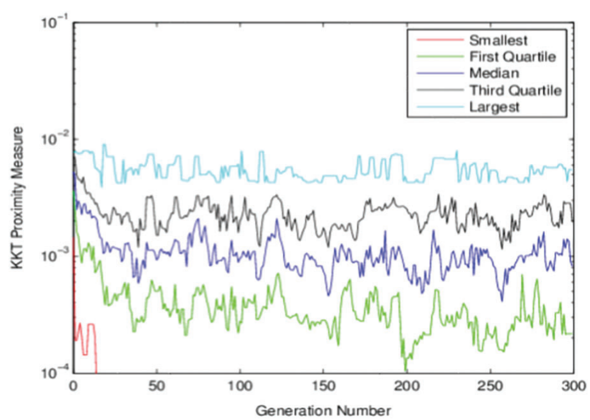

Figure 2: KKT Proximity measure *vs*. generation number for Model-I using NSGA-II

6.2 Model-II for Portfolio Problem [31]:

$max \quad En(X) = -(x_1 \log x_1 + x_2 \log x_2 + x_3 \log x_3)$

$max \quad Er(X) = 0.062x_1 + 0.146x_2 + 0.128x_3$

$min \quad Vr(X) = 0.0146x_1^2 + 0.0854x_2^2 + 0.0289x_3^2 + 2(0.0187x_1x_2 + 0.0145x_1x_3 + 0.0104x_2x_3)$

$subject\ to \quad x_1 + x_2 + x_3 = 1,$

$x_1,\ x_2,\ x_3 \geq 0.$

Fig. 3 shows the non-dominated points for model-II obtained by the NSGA-III algorithm. In this figure, NSGA-III runs 300 generations with 100 population sizes. The obtained solutions for this model by the proposed algorithm equal previously published results for this model. In Fig. 4, the relation between generation number and the KKT proximity measure is introduced. As shown from the figures, the KKT metric reduces with increasing the number of generations.

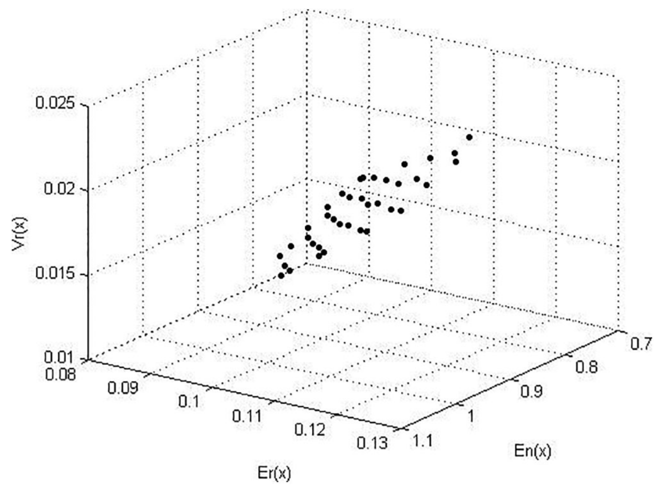

Figure 3: Pareto optimal points for model- II objective functions

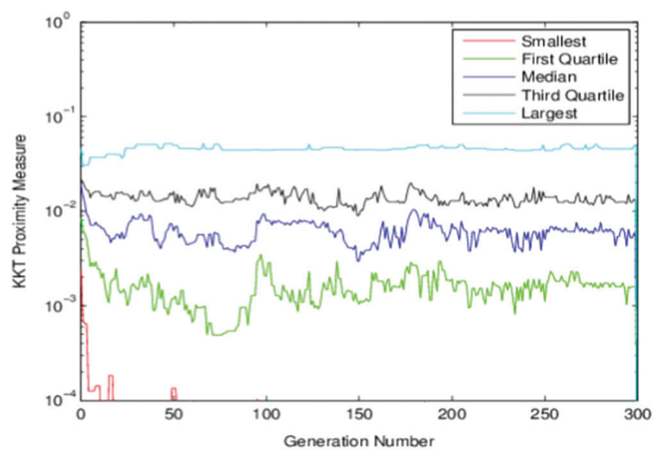

Figure 4: KKT Proximity measure *vs*. generation number for Model-II using NSGA-III

The minimum KKT error metric goes to zero with the first few generations, which means at least one solution converges to the efficient front within a few generations. Also, the maximum KKTPM metric values don't show any convergence until the last generation.

7 Conclusion

The solutions found in genetic algorithms are the same as those found in other approaches, and they are as effective. The genetic algorithm, on the other hand, is simpler and easier to apply. A genetic algorithm can address portfolio optimization problems without simplification and with decent results in a fair amount of time, and it has a lot of practical applications. NSGA-II and NSGA-III are used to address portfolio problems in models I and II. We measure the smallest, first quartile, median, third quartile, and highest KKTPM values as a function of generation, and the figure shows that KKTPM values decrease with generation. The obtained solutions for the portfolio models using the genetic algorithms same as theoretical solutions. NSGA-II is effective only for two objective functions, and NSGA-III is effective only for three objective functions. NSGA-II can be solving real-life optimization problems with two objective functions, and NSGA-III can be solving real-life optimization problems with three objective functions. In the future direction of this work, we will extend the proposed algorithms with more real-life applications with many objective functions.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that no conflicts of interest occurred regarding the publication of the paper.

References

[1] A. T. Khan, X. Cao, S. Li, B. Hu and V. N. Katsikis, "Quantum beetle antennae search: A novel technique for the constrained portfolio optimization problem," *Science China Information Sciences*, vol. 64, no. 5, pp. 1–14, 2021.

[2] D. E. Goldberg, "Genetic algorithms in search, optimization, and machine learning," *Reading: Addison-Wesley* vol. 3, no. 2, pp. 25–45, 1989.

[3] J. Patalas-Maliszewska, I. Pająk and M. Skrzeszewska, "Ai-based decision-making model for the development of a manufacturing company in the context of industry 4.0," in *2020 IEEE Int. Conf. on Fuzzy Systems (FUZZ-IEEE)*, Glasgow, Scotland, UK, IEEE, pp. 1–7, 2020.

[4] S. A. Darani, "System architecture optimization using hidden genes genetic algorithms with applications in space trajectory optimization," *Michigan Technological University*, vol. 12, no. 2, pp. 5–24, 2018.

[5] T. Harada and E. Alba, "Parallel genetic algorithms: A useful survey," *ACM Computing Surveys (CSUR)*, vol. 53, no. 4, pp. 1–39, 2020.

[6] Z. Drezner and T. D. Drezner, "Biologically inspired parent selection in genetic algorithms," *Annals of Operations Research*, vol. 287, no. 1, pp. 161–183, 2020.

[7] S. Katoch, S. S. Chauhan and V. Kumar, "A review on genetic algorithm: Past, present, and future," *Multimedia Tools and Applications*, vol. 80, no. 5, pp. 8091–8126, 2021.

[8] J. González-Díaz, B. González-Rodríguez, M. Leal and J. Puerto, "Global optimization for bilevel portfolio design: Economic insights from the Dow Jones index," *Omega*, vol. 102, no. 5, pp. 102353, 2021.

[9] A. Goli, H. K. Zare, R. Tavakkoli-Moghaddam and A. Sadeghieh, "Hybrid artificial intelligence and robust optimization for a multi-objective product portfolio problem Case study: The dairy products industry," *Computers & Industrial Engineering*, vol. 137, no. 4, pp. 106090, 2019.

[10] H. Markowitz, "Portfolio selection: Efficient diversification of investments," *Cowles Foundation Monograph*, vol. 16, no. 3, pp. 24–45, 1959.

[11] K. Deb, "Karush-kuhn-tucker proximity measure for convergence of real-parameter single and multi-objective optimization," *Numerical Computations: Theory and Algorithms NUMTA*, vol. 27, no. 3, pp. 23–34, 2019.

[12] M. Abouhawwash, M. Jameel and K. Deb, "A smooth proximity measure for optimality in multi-objective optimization using Benson's method," *Computers & Operations Research*, vol. 117, no. 2, pp. 104900, 2020.

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

[14] S. P. Kodali, R. Kudikala and K. Deb, "Multi-objective optimization of surface grinding process using NSGA II," in *2008 First Int. Conf. on Emerging Trends in Engineering and Technology (IEEE)*, IEEE, pp. 763–767, 2008.

[15] X. D. Wang, C. Hirsch, S. Kang and C. Lacor, "Multi-objective optimization of turbomachinery using improved NSGA-II and approximation model," *Computer Methods in Applied Mechanics and Engineering*, vol. 200, no. 9–12, pp. 883–895, 2011.

[16] 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*, vol. 18, no. 4, pp. 577–601, 2014.

[17] W. Mkaouer, M. Kessentini, A. Shaout, P. Koligheu, S. Bechikh *et al.,* "Many-objective software remodularization using NSGA-III," *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol. 24, no. 3, pp. 1–45, 2015.

[18] Y. Zhu, J. Liang, J. Chen and Z. Ming, “An improved NSGA-III algorithm for feature selection used in intrusion detection,” *Knowledge-Based Systems*, vol. 116, no. 12, pp. 74–85, 2017.

[19] J. H. Yi, L. N. Xing, G. G. Wang, J. Dong, A. V. Vasilakos *et al.,* “Behavior of crossover operators in NSGA-III for large-scale optimization problems,” *Information Sciences*, vol. 509, no. 15, pp. 470–487, 2020.

[20] H. Markowitz, “Portfolio selection,” *Journal of Finance*, vol. 7, no. 1, pp. 77–91, 1952.

[21] R. O. Michaud, “The Markowitz optimization enigma: Is 'optimized' optimal,” *Financial Analysts Journal*, vol. 45, no. 1, pp. 31–42, 2018.

[22] H. Hong and J. C. Stein, “A unified theory of underreaction, momentum trading, and overreaction in asset markets,” *Journal of Finance*, vol. 54, no. 6, pp. 2143–2184, 1999.

[23] K. Deb, “Multi-objective optimization using evolutionary algorithms,” John Wiley & Sons, Vol. 16, 2001.

[24] H. Chen, K. P. Wong, D. H. M. Nguyen and C. Y. Chung, “Analyzing oligopolistic electricity market using coevolutionary computation,” *IEEE Transactions on Power Systems*, vol. 21, no. 1, pp. 143–152, 2006.

[25] X. S. Yang, “Nature-inspired optimization algorithms,” *Academic Press*, vol. 14, no. 3, pp. 1–45, 2020.

[26] 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*, vol. 18, no. 4, pp. 602–622, 2014.

[27] H. Seada and K. Deb, “U-nsga-iii: A unified evolutionary optimization procedure for single, multiple, and many objectives: Proof-of-principle results,” in *Int. Conf. on Evolutionary Multi-Criterion Optimization*, Cham, Springer, pp. 34–49, 2015.

[28] T. Huang, Y. J. Gong, S. Kwong, H. Wang and J. Zhang, “A niching memetic algorithm for multi-solution traveling salesman problem,” *IEEE Transactions on Evolutionary Computation*, vol. 24, no. 3, pp. 508–522, 2019.

[29] K. Deb, M. Abouhawwash and H. Seada, “A computationally fast convergence measure and implementation for single-, multiple-, and many-objective optimization,” *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 1, no. 4, pp. 280–293, 2017.

[30] E. Ahmed and A. S. Hegazi, “On different aspects of portfolio optimization,” *Applied Mathematics and Computation*, vol. 175, no. 1, pp. 590–596, 2006.

[31] M. Abouhawwash and A. Alessio, “Develop a multi-objective evolutionary algorithm for PET image reconstruction: Concept,” *IEEE Transactions on Medical Imaging*, vol. 40, no. 8, pp. 2142–2151, 2021.

[32] M. Abouhawwash and K. Deb, “Reference point based evolutionary multi-objective optimization algorithms with convergence properties using KKTPM and ASF metrics,” *Journal of Heuristics, Springer*, vol. 27, no. 4, pp. 575–614, 2021.

[33] K. Deb and M. Abouhawwash, “An optimality theory-based proximity measure for set-based multi-objective optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 4, pp. 515–528, 2016.

[34] M. Abouhawwash, H. Seada and K. Deb, “Towards faster convergence of evolutionary multi-criterion optimization algorithms using Karush-Kuhn-Tucker optimality based local search,” *Computers & Operations Research*, vol. 79, no. 3, pp. 331–346, 2017.

[35] B. Samanta and T. K. Roy, “Multi-objective portfolio optimization model,” *Tamsui Oxford Journal of Mathematical Sciences*, vol. 21, no. 1, pp. 55, 2005.