Journal of Industrial and Systems Engineering
Vol. 16, No. 3, pp. 1-19
Summer (July) 2024

The Mean-Variance Cardinality Constrained Portfolio Selection using an Enhanced Genetic Algorithm with a Novel Crossover Operator

Hossein Ghanbari[1], Emran Mohammadi[1*], Farnaz Barzinpour[1], Alireza Paeizi[1]

[1] *Department of Industrial Engineering, Iran University of Science and Technology, Tehran, Iran*

Abstract

Portfolio selection, a cornerstone of financial engineering, focuses on optimizing asset allocation to maximize returns while minimizing risk. While Markowitz's mean-variance model is foundational, its reliance on unrealistic assumptions, such as perfect market conditions, limits its practical utility. Notably, it overlooks real-world constraints like cardinality (limiting the number of assets) and floor-ceiling (restricting allocation ranges). To address these issues, this study extends the classical model by incorporating these constraints, transforming it into a mixed-integer quadratic problem that requires advanced algorithms for solutions. Among metaheuristic approaches, genetic algorithms are favored for balancing solution quality and computational efficiency. However, standard genetic algorithms face challenges in managing complex constraints. This paper introduces a novel crossover operator to enhance the algorithm's performance, applying it to the S&P 100 dataset. Results confirm the proposed method's effectiveness in generating high-quality solutions, underscoring its potential for practical portfolio optimization.

Keywords: Portfolio selection problem, Markowitz's mean–variance framework, Cardinality constraint, Genetic algorithm, Crossover operator

[*] Corresponding Author
ISSN: 1735-8272,

1- Introduction

Investment plays a crucial role in the finance industry and has a significant influence on economic growth. Through capital market investments, investors can not only grow their wealth but also safeguard it against inflation-related losses, ensuring long-term financial stability (Bagheriyan et al., 2023; Fooeik et al., 2022; Ghanbari et al., 2022). There are numerous strategies investors can use to allocate their wealth in the capital market. Obviously, an effective investment strategy seeks to maximize returns while minimizing the risk of loss, ensuring a balanced approach to wealth growth and risk management (Abdollahi Moghadam et al., 2019; Aguilar-Rivera et al., 2015; Mohammadi et al., 2022; Sabar et al., 2018). Portfolio selection is one of the most common investment strategies for spreading the risk of an investment, which was first introduced by Markowitz (1952). In his classical work, Markowitz presented the mean-variance portfolio selection model which became a new paradigm of portfolio construction for investors by forming a portfolio as a bi-criteria optimization problem with a tradeoff between risk and return (Soleimani et al., 2009). In the last few decades, portfolio selection has drawn the attention of researchers and practitioners alike, and by technology advancements as well as the rise of computing power it is still a developing topic (Ghanbari et al., 2023).

Although the basic Markowitz mean–variance model has theoretical elegance and great significance, it is still met with skepticism among investment practitioners. This is mainly because the traditional portfolio selection model is simplified with unrealistic assumptions that are of little use in practice (Sadjadi et al., 2012). The Markowitz mean-variance model assumes a perfect market with no transaction costs or taxes, in which short selling is not allowed, but assets are infinitely divisible and therefore can be traded in any (non-negative) fraction (Lwin and Qu, 2013; Woodside-Oriakhi et al., 2011). However, in real-world scenarios, investors often face constraints such as cardinality and floor and ceiling (quantity) constraints. These constraints are valuable to investors and are considered by them in the actual portfolio selection process (Anagnostopoulos and Mamanis, 2011; Chang et al., 2000; Deng et al., 2012; Mitra et al., 2003). Although by considering additional criteria in the selection process, the new portfolios may be less profitable or riskier on the one hand, but on the other hand these new portfolios beat the old portfolios in other aspects that are important for investors. Therefore, to address these limitations, this paper extends the mean-variance portfolio selection model by incorporating cardinality and floor-ceiling (quantity) constraints. The cardinality constraint ensures that the portfolio includes a specified number of assets, while the floor-ceiling constraint regulates the allocation to each asset, restricting it within predefined bounds.

The great importance of the cardinality constrained portfolio selection problem has motivated many researchers to investigate different techniques to solve this problem. Some researchers have proposed various exact approaches to solve the portfolio selection problem with cardinality constraints (Bertsimas and Shioda, 2009; Bienstock, 1996; Borchers and Mitchell, 1994; Frangioni and Gentile, 2006; Gulpinar et al., 2010; Lee and Mitchell, 1997; Li et al., 2006; Shaw et al., 2008; Vielma et al., 2008). Nevertheless, these exact techniques are ineffective when applied to large-scale problems because they may fail to find an optimal solution in a reasonable amount of time, however, the solution to this problem with a sufficiently small number of variables can be solved by quadratic programming.

Extending the portfolio selection problem with cardinality constraint alone already transforms the classical quadratic optimization model to a mixed-integer quadratic problem which has been shown that belongs to the class of NP-hard problems (Bienstock, 1996; Moral-Escudero et al., 2006; Shaw et al., 2008). Since mixed-integer quadratic problem are hard to solve optimally, many researchers have applied different approximate algorithms such as meta-heuristics and hybrid meta-heuristics to the constrained portfolio selection problem (Chang et al., 2000, 2009; Crama and Schyns, 2003; Deng et al., 2012; Fernández and Gómez, 2007; Golmakani and Fazel, 2011; Jobst et al., 2001; Kellerer et al., 2000; Kendall and Su, 2007; Mansini and Speranza, 1999; Maringer and Kellerer, 2003; Schaerf, 2002; Shoaf and Foster, 1996; Soleimani et al., 2009; Speranza, 1996; Woodside-Oriakhi et al., 2011; Yu et al., 2006; Zhu et al., 2011). It is important to note that meta-heuristics cannot guarantee that a solution will be optimal, but they are effective in finding the optimal or near-optimal solutions in a reasonable time frame.

In recent years, several methods based on artificial intelligence have been applied to solve the portfolio selection problem. Among them, the genetic algorithm as a biologically inspired algorithm is one of the most widely used methods to improve the results of the constrained portfolio selection problem (Bavarsad Salehpoor and Molla-Alizadeh-Zavardehi, 2019; Doering et al., 2019; Kalayci et al., 2019). Genetic algorithms are stochastic, heuristic techniques based on the natural selection principles that can deal with many types of problems (D. Lin et al., 2005). The first application of the genetic algorithm to the basic mean-variance model was done by Shoaf and Foster (1996). In their paper in 1996, they found that the time complexity of the genetic algorithm is approximately $O(n \, log \, n)$ and is better than that of quadratic programming. After Shoaf and Foster (1996), many researchers have proposed genetic algorithms to solve the portfolio selection problem.

Chang et al. (2000) used genetic algorithms, tabu search, and simulated annealing to solve the portfolio selection problem with cardinality constraints. Lin and Liu (2008) proposed a genetic algorithm with three different models for portfolio selection problems with round lots. Chang et al. (2009) used the genetic algorithm to compare their heuristic approach to portfolio optimization problems in different risk measures with the mean-variance model under cardinality constraints. Soleimani et al. (2009) solved Markowitz's portfolio selection problem with three additional constraints (minimum transaction lots, cardinality constraints, and market capitalization) using a genetic algorithm. Sabar et al. (2018) introduced a multi-population genetic algorithm methodology to solve portfolio selection problems more effectively and more efficiently. Mittal and Srivastava (2021) proposed a constrained mean-variance-skewness portfolio optimization problem with an uncertain environment and solved it using an improved genetic algorithm. Konstantinou et al. (2022) presented a consistent and effective hybrid optimization scheme (sonar inspired optimization algorithm and genetic algorithm) for solving portfolio optimization problems with cardinality constraints. Experimental results showed that genetic algorithms perform better than other heuristic algorithms such as simulated annealing and tabu search. However, the standard genetic algorithm is not without its shortcomings, particularly when handling the complexity of constrained portfolio optimization.

To overcome these limitations, this paper proposes a novel crossover operator designed to enhance the genetic algorithm's performance, enabling it to generate high-quality solutions. This enhanced algorithm is then applied to the extended portfolio selection problem, using the S&P 100 dataset for empirical testing. The study focuses on the mean-variance portfolio selection model, incorporating two of the most widely used constraints: cardinality and floor-ceiling (lower and upper bounds). These constraints ensure the portfolio contains a predefined number of assets while restricting asset allocations to specific limits, reflecting realistic investment conditions.

The rest of this paper is structured as follows. Section 2 provides a detailed explanation of the expanded portfolio selection model, including the incorporation of cardinality and floor-ceiling constraints. In Section 3, we introduce the enhanced genetic algorithm, describing its novel crossover operator and how it is applied to solve the proposed model. Section 4 presents the experimental setup, along with the computational results obtained from testing the algorithm on the S&P 100 dataset. Finally, Section 5 offers concluding remarks and outlines potential directions for future research.

2- The Proposed Model

The portfolio selection problem, a fundamental challenge in financial decision-making, focuses on the optimal allocation of wealth across a set of available assets. This problem was first introduced by Markowitz (1952) in his pioneering work, where he assumed that investors are generally risk-averse. To address the inherent tradeoff between risk and return, Markowitz proposed the mean-variance model, which seeks to minimize portfolio risk (measured by variance) while targeting a specific desired level of return. This model revolutionized modern portfolio theory and laid the foundation for portfolio optimization, becoming a cornerstone of financial economics.

As discussed in this paper, we focus on an extended version of the mean-variance model that incorporates additional constraints to reflect real-world investment limitations. Specifically, we address the cardinality constraint, which limits the number of assets in the portfolio, and the floor and ceiling (lower and upper bounds) constraints, which regulate the allocation to each asset. These constraints are among the most commonly used in portfolio selection (Metaxiotis and Liagkouras, 2012). Therefore, our approach builds on these extensions to create a more realistic and practical framework for portfolio optimization. The extended mean-variance model with these constraints is defined as follows (Chang et al., 2000):

Parameters:

N: The number of available assets
μ_i: Expected return of asset i ($i = 1, 2, \dots, N$)
σ_{ij}: Covariance between asset i and asset j ($i = 1, 2, \dots, N$; $j = 1, 2, \dots, N$)
R^*: The given expected return
K: Desired number of assets to be hold in the portfolio
l_i: Minimum proportion of asset i ($i = 1, 2, \dots, N$)
u_i: Maximum proportion of asset i ($i = 1, 2, \dots, N$)

Variables:

w_i: Proportion of asset i ($i = 1, 2, \dots, N$)
Z_i: The status of asset selection ($Z_i = 1$ if asset i is chosen to be held, otherwise $Z_i = 0$)

Proposed Model:

$$Minimize \sum_{i=1}^{N}\sum_{j=1}^{N} w_i\, w_j\, \sigma_{ij} \tag{1}$$

Subjected to:

$$\sum_{i=1}^{N} w_i\, \mu_i = R^* \tag{2}$$

$$\sum_{i=1}^{N} w_i = 1 \tag{3}$$

$$\sum_{i=1}^{N} Z_i = K \tag{4}$$

$$l_i Z_i \le w_i \le u_i Z_i \qquad i = 1, 2, \dots, N \tag{5}$$

$$Z_i = \{0,1\} \qquad i = 1, 2, \dots, N \tag{6}$$

$$0 \le w_i \le 1 \qquad i = 1, 2, \dots, N \tag{7}$$

$$0 \le l_i \le u_i \le 1 \qquad i = 1, 2, \dots, N \tag{8}$$

Equation (1) aims to minimize the total variance (risk) of the portfolio, while equation (2) ensures that the portfolio achieves the desired level of return. Equation (3) guarantees that the entire budget is fully allocated across the selected assets. Equation (4) describes the cardinality constraint, which limits the number of assets held in the portfolio, while equation (5) describes the floor and ceiling (quantity) constraints which sets the proportion of assets held between a given range $[l_i, u_i]$. The decision variable Z_i in equation (6) represents the selection status of each asset, indicating whether an asset is included in the portfolio or not. Equation (7) emphasizes the fact that the proportion of assets must not be negative, which means that short selling is not allowed, and equation (8) defines variable ranges for asset proportions.

In multi-objective models, it is often impossible to optimize all objective functions simultaneously. Therefore, to address this issue it is needed to give priority to some objectives over others by using methods such as weighted sum approach. The weighted sum method combines multiple objectives into a single objective function by assigning a weight to each, reflecting their relative importance. As can be seen in equation (9), the quadratic objective function balances two competing objectives—return maximization and risk minimization—through a weighting parameter (λ). This parameter allows for a flexible trade-off between risk and return, depending on the investor's risk aversion.

$$Minimize\ \lambda \left[\sum_{i=1}^{N}\sum_{j=1}^{N} w_i\, w_j\, \sigma_{ij}\right] - (1-\lambda)\left[\sum_{i=1}^{N} w_i\, \mu_i\right] \tag{9}$$

Subjected to:

$$\sum_{i=1}^{N} w_i = 1 \tag{10}$$

$$\sum_{i=1}^{N} Z_i = K \tag{11}$$

$$l_i Z_i \leq w_i \leq u_i Z_i \qquad i = 1, 2, \dots, N \tag{12}$$

$$Z_i = \{0,1\} \qquad i = 1, 2, \dots, N \tag{13}$$

$$0 \leq w_i \leq 1 \qquad i = 1, 2, \dots, N \tag{14}$$

$$0 \leq l_i \leq u_i \leq 1 \qquad i = 1, 2, \dots, N \tag{15}$$

The λ weighting parameter, which balances risk and return, reflects the investor's risk aversion and is assigned different values between 0 and 1. Clearly, the sensitivity of the investor to the risk increases as λ increases from zero to one. When $\lambda = 1$, the investor adopts a highly conservative approach, with the objective function focused solely on minimizing risk. Conversely, when $\lambda = 0$ the investor is highly aggressive, prioritizing return maximization with no consideration for risk. For values of $0 < \lambda < 1$, the parameter achieves a trade-off, allowing for a balance between risk and return, depending on the investor's specific risk tolerance.

3- Utilizing Genetic Algorithm

Genetic algorithms are among the most widely studied and applied approaches in the evolutionary algorithm literature. They have demonstrated remarkable success in solving complex optimization problems and have received increasing attention over the last few decades. Genetic algorithms, originally introduced by Holland (1975), can be described as a heuristic search and optimization technique based on the famous Darwinian principle of natural selection and the well-known phrase “survival of the fittest”. In essence, individuals who are better adapted to their environment have a higher likelihood of surviving and reproducing, while less adapted individuals are gradually eliminated.

The genetic algorithm simulates these processes by taking an initial population of individuals and applying genetic operators to each reproduction. In this algorithm, an initial population consisting of chromosomes (several feasible solutions) is randomly generated. In the context of portfolio selection problems, each individual or “chromosome” in the initial population represents a potential solution, where the genes of the chromosome correspond to the proportions of assets in the portfolio. The initial population is typically generated randomly to ensure diversity in the search space. Once the initial population is formed, parents are selected based on their fitness to produce offspring, leading to a new generation that combines characteristics of both the parents and offspring. The fitness of each chromosome is evaluated using a fitness function, which measures how well the solution meets the objective of the optimization problem. In portfolio selection, this function typically reflects how well a given portfolio balances risk and return.

The genetic algorithm relies on two main operators—crossover and mutation—to introduce diversity and improve the quality of solutions in each generation. Crossover combines the genetic material of two parents to produce offspring, while mutation introduces random changes to individual chromosomes to explore new areas of the solution space. In each generation, the quality or desirability of the chromosomes (solutions) is evaluated using a fitness function derived from the objective function of the model. The higher the fitness value, the closer the solution is to optimizing the objective. The selection operator then ensures that chromosomes with higher fitness values have a greater probability of being selected for reproduction, increasing their likelihood of appearing in the next generation. This mechanism drives the algorithm toward progressively better solutions over time. The basic steps of a simple genetic algorithm can be summarized as follows (Chang et al., 2000):

Generate an initial population
Evaluate fitness of individuals in the population
Repeat:
Select parents from the population
Recombine (mate) parents to produce children using cross over and mutation operators
Evaluate fitness of the children
Replace some or all of the population by the children
Until a satisfactory solution has been found.

Figure 1. Structure of a simple genetic algorithm

In this section, we introduce the proposed genetic algorithm designed to address the portfolio selection problem under cardinality and floor-ceiling constraints. The algorithm follows the general framework of evolutionary algorithms, leveraging selection, crossover, and mutation operators to explore the solution space and optimize the portfolio's risk-return balance. The selection strategy and novel crossover operator have been carefully developed to ensure efficient search and high-quality solutions. Below, we outline the details of the selection process and the specific rules governing the proposed crossover operator.
The selection strategy is determined as follows:

- Evaluating The fitness of individuals in the population using the fitness function.
- Sorting the population regarding the fitness values.
- The top half of the fittest individuals are automatically passed on to the next generation.
- Selecting parents randomly from the current population.
- Each pair of parents produces offspring using crossover and mutation operators.
- These offspring, which constitute the remaining half of the population, then proceed to the next generation. In this way, the next generation is formed by combining the fittest individuals with the newly generated offspring.

The crossover operator proposed in this paper is a novel approach, drawing inspiration from the RAR operator introduced by Radcliffe (1991). The rules governing this crossover process are outlined as follows:

1. If both parents contain a specific asset in their portfolios, then that asset is transferred to the child's portfolio, with its allocation being randomly assigned between the values from the two parents.
2. If neither parent includes a specific asset, that asset will not be included in the child's portfolio.

3. If one parent holds a specific asset while the other does not, the final value of that asset in the child's portfolio is randomly assigned between having it or not (Note: If the cardinality constraint (K) has already been reached, the asset will not be assigned to the child's portfolio).

Throughout the crossover process, all constraints are dynamically monitored to ensure the child's portfolio remains feasible. If a constraint is violated, the portfolio is repaired, and the crossover operation is repeated until a valid solution is generated.

As part of the proposed genetic algorithm, a reparation process is applied to ensure the portfolio remains feasible, particularly in cases where the sum of asset proportions does not equal 1, which is a key constraint in portfolio selection. If the sum of proportions is less than 1, the difference between the sum and 1 is added to the chromosome with the largest asset proportion. Conversely, if the sum of proportions exceeds 1, the surplus is subtracted from the chromosome with the smallest asset proportion, provided that this chromosome's value is at least equal to the gap. This reparation step ensures the portfolio remains balanced and meets the total investment constraint, preserving the feasibility of each solution.

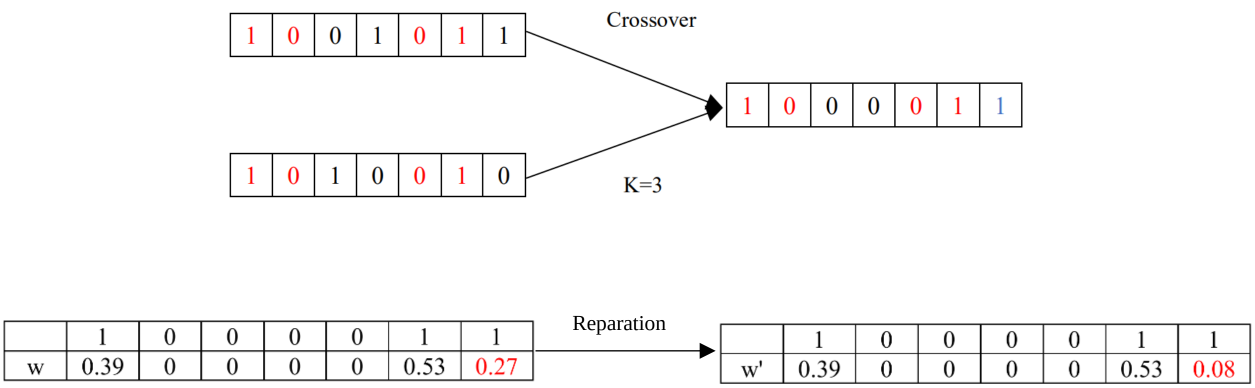


Figure 2. An example of (a) crossover operation and (b) reparation operations that are used in the proposed genetic algorithm

Section 4 will verify and validate this genetic algorithm implementation and provide further explanation of the genetic algorithm's operation and model aspects.

4- Computational Results

In this section, we first introduce the experimental setup, including the definition of experiments and parameter settings. Following that, we present the experimental results of the proposed genetic algorithm in detail, along with a comprehensive comparison to the exact mixed-integer quadratic programming solver.

4.1- Definition of Experiments

To evaluate computational results for a specific problem category, it is important to have benchmark datasets available for researchers to compare the efficiency of their algorithms and the quality of their solutions. The most commonly used publicly available datasets for portfolio selection problems are provided by Beasley (http://people.brunel.ac.uk/~mastjjb/jeb/orlib/portinfo.html). This dataset includes weekly prices from March 1992 to September 1997, along with covariance matrices and expected return vectors for different stock market indices. Our proposed algorithm was tested on this well-known publicly available benchmark problem using weekly prices from the US S&P 100 index between March 1992 and September 1997, and compared against the exact mixed integer quadratic programming solver in GAMS. The proposed genetic algorithm was implemented in Python and ran on a personal computer with an INTEL CORE i7-2630QM CPU @ 2.00GHZ.

4.2- Parameters Setting

This subsection outlines the parameter settings used in the proposed genetic algorithm. Those familiar with heuristic algorithms will know that assigning parameter values is often necessary and an essential part of the solution approach for the specific problem. In this section, we explain how to arrive at the parameter values by testing several different values using selected data sets. The parameters that needed to be decided for the proposed genetic algorithm were population size, probabilities of crossover, and mutation rates. The results indicated that our genetic algorithm converged after more than 60 iterations, with the best possibilities of cross-over and mutation rates being 0.8 and 0.3, respectively, and the population size performed acceptably at 100. The best parameter values for all algorithms are presented in Table 1 and were kept constant for all instances to test the techniques' robustness and scalability.

Table 1. Genetic algorithm parameter setting

Parameters	**Tested Range**	**Suggested Values**
Cross over rate	0.1-0.9	0.8
Mutation rate	0.1-0.9	0.3
Population size	50-200	100

4.3- Experimental Results

This subsection presents experimental results demonstrating the accuracy and efficiency of our proposed genetic algorithm. To achieve this, we conducted two sets of experiments aimed at evaluating the algorithm’s effectiveness in identifying practical optimal portfolios. We then compared the results of the genetic algorithm with those obtained from the model implemented in GAMS[†] software. In the first set of experiments, the lower and upper bounds for the asset proportions were set to $l_i = 0.05$ and $u_i = 0.40$, respectively. In the second set, these bounds were

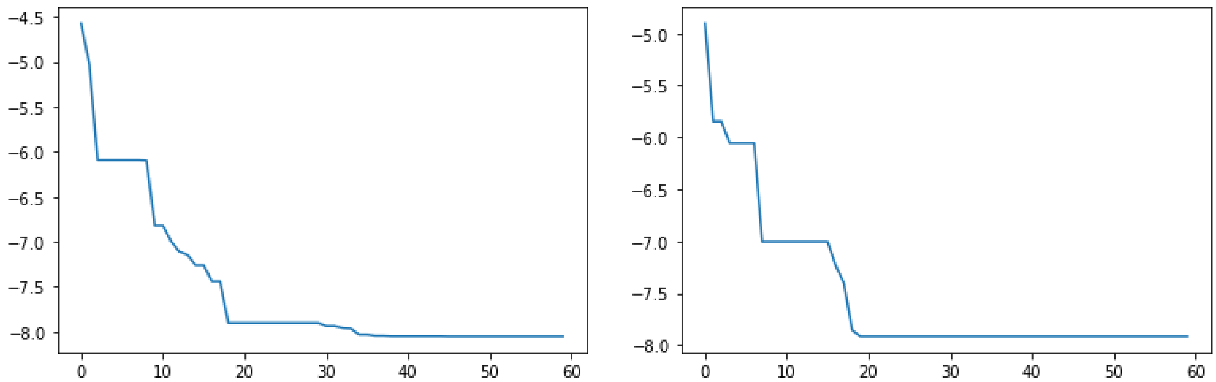
[†] General Algebraic Modeling System (GAMS)

adjusted to $l_i = 0.05$ and $u_i = 0.75$. Cardinality constraints were imposed by setting K to 5, 10, and 12, based on expert opinions and observations from various datasets that supported these values. This configuration led to the creation of six experimental scenarios, and the results of the proposed genetic algorithm for each scenario are presented below.

Table 2. The results of experiments for the proposed genetic algorithm with $K = 5, 10, and$ 12 across different floor and ceiling constraints

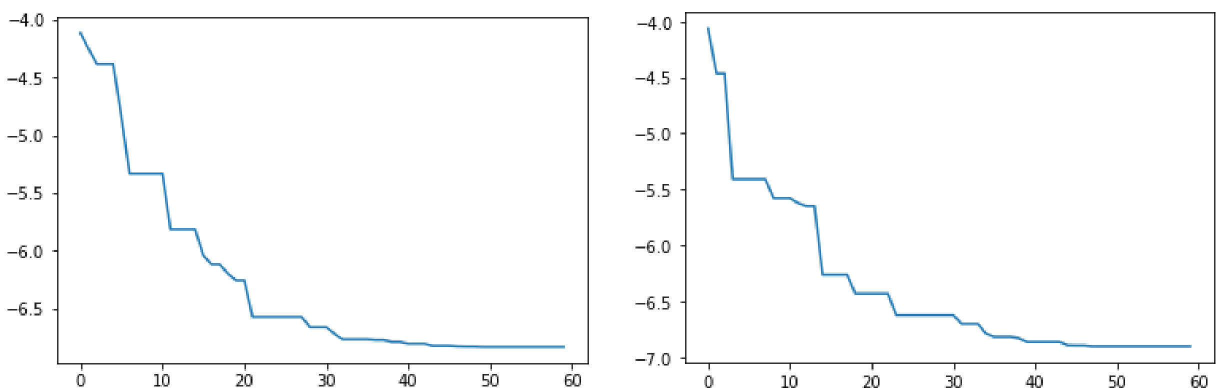
$\boldsymbol{K}$	$\boldsymbol{l_i}$	$\boldsymbol{u_i}$	$\boldsymbol{Objective\ Function}$	$\boldsymbol{w_i}$	
$K = 5$	$l_i = 0.05$	$u_i = 0.40$	8.05468993	$w_{14} = 0.080$	$w_{82} = 0.364$
				$w_{34} = 0.227$	$w_{89} = 0.118$
				$w_{42} = 0.214$	-
	$l_i = 0.05$	$u_i = 0.75$	7.93751583	$w_{14} = 0.060$	$w_{82} = 0.317$
				$w_{34} = 0.090$	$w_{89} = 0.181$
				$w_{42} = 0.355$	-
$K = 10$	$l_i = 0.05$	$u_i = 0.40$	6.82982376	$w_2 = 0.099$	$w_{43} = 0.094$
				$w_{14} = 0.119$	$w_{76} = 0.061$
				$w_{34} = 0.124$	$w_{82} = 0.158$
				$w_{36} = 0.066$	$w_{89} = 0.107$
				$w_{42} = 0.110$	$w_{93} = 0.060$
	$l_i = 0.05$	$u_i = 0.75$	6.90268329	$w_2 = 0.126$	$w_{43} = 0.068$
				$w_{14} = 0.089$	$w_{76} = 0.057$
				$w_{23} = 0.070$	$w_{82} = 0.113$
				$w_{34} = 0.166$	$w_{89} = 0.115$
				$w_{42} = 0.124$	$w_{93} = 0.073$
$K = 12$	$l_i = 0.05$	$u_i = 0.40$	6.35234111	$w_2 = 0.091$	$w_{43} = 0.072$
				$w_{14} = 0.088$	$w_{67} = 0.060$
				$w_{23} = 0.104$	$w_{76} = 0.068$
				$w_{34} = 0.094$	$w_{82} = 0.116$
				$w_{36} = 0.069$	$w_{89} = 0.087$
				$w_{42} = 0.072$	$w_{93} = 0.080$
	$l_i = 0.05$	$u_i = 0.75$	6.53988677	$w_2 = 0.084$	$w_{42} = 0.093$
				$w_{14} = 0.080$	$w_{43} = 0.075$
				$w_{20} = 0.062$	$w_{76} = 0.080$
				$w_{23} = 0.067$	$w_{82} = 0.112$
				$w_{34} = 0.135$	$w_{89} = 0.083$
				$w_{36} = 0.050$	$w_{93} = 0.078$

Figures 3-5 illustrate the convergence performance of the proposed genetic algorithm across all six scenarios. These graphs depict the algorithm's behavior over a series of iterations, highlighting how it progressively approaches the optimal solution. The results show that, in all six scenarios, the genetic algorithm demonstrates steady convergence toward optimal or near-optimal solutions. In particular, the algorithm achieves faster convergence in scenarios with tighter upper bounds ($u_i = 0.40$) and smaller cardinality constraints ($K = 5$), while scenarios with more relaxed constraints ($u_i = 0.75$) and larger K values require a higher number of iterations to reach a stable solution. Overall, the figures show that the proposed genetic algorithm efficiently balances exploration and exploitation, converging to high-quality solutions in a reasonable time frame, regardless of the parameter settings.



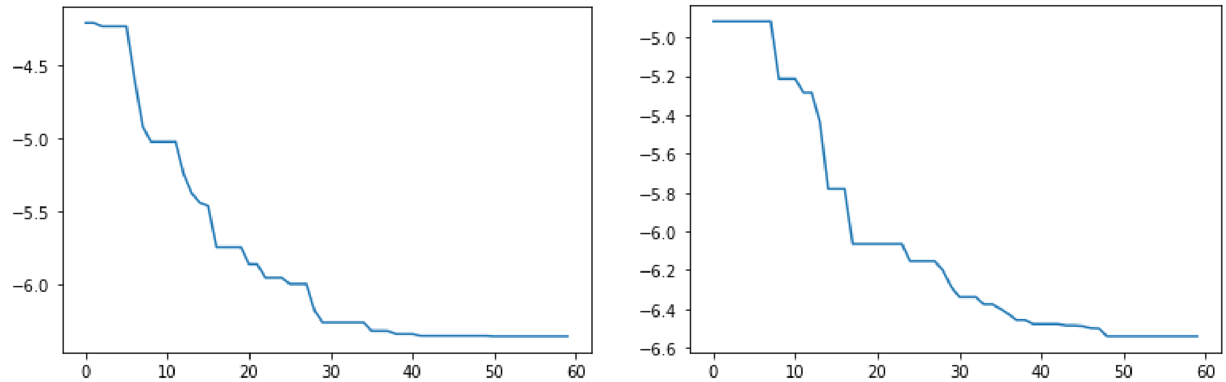
$(a)\ l_i = 0.05, u_i = 0.40$ $(b)\ l_i = 0.05, u_i = 0.75$

Figure 3. Comparison of the proposed genetic algorithm's converging performance for $K = 5$ under various floor and ceiling constraints



$(a)\ l_i = 0.05, u_i = 0.40$ $(b)\ l_i = 0.05, u_i = 0.75$

Figure 4. Comparison of the proposed genetic algorithm's converging performance for $K = 10$ under various floor and ceiling constraints



$(a)\ l_i = 0.05, u_i = 0.40$ $(b)\ l_i = 0.05, u_i = 0.75$

Figure 5. Comparison of the proposed genetic algorithm's converging performance for $K = 12$ under various floor and ceiling constraints

In order to verify the accuracy and efficiency of our proposed genetic algorithm, we conducted a series of computational experiments comparing the performance of our heuristic approach with the results from the mixed-integer quadratic programming solver in GAMS. The experiments were designed to assess both the accuracy of the solutions and the computational efficiency of the proposed algorithm under different conditions. We performed these experiments for all six scenarios, to reflect different real-world investment conditions. Table 3 presents a detailed comparison of the numerical outcomes obtained by both the proposed genetic algorithm and the exact mixed-integer quadratic programming solver.

Table 3. The results of experiments in GAMS and the proposed Genetic Algorithm for $K = 5, 10, and\ 12$ across different floor and ceiling constraints

$\boldsymbol{K}$	$\boldsymbol{l_i}$	$\boldsymbol{u_i}$	GAMS	Proposed Genetic Algorithm	Deviation (%)
$K = 5$	$l_i = 0.05$	$u_i = 0.40$	8.1667	8.0546	1.37 %
	$l_i = 0.05$	$u_i = 0.75$	8.1997	7.9375	3.19 %
$K = 10$	$l_i = 0.05$	$u_i = 0.40$	7.4980	6.8298	8.91 %
	$l_i = 0.05$	$u_i = 0.75$	7.5012	6.9026	7.98 %
$K = 12$	$l_i = 0.05$	$u_i = 0.40$	7.1335	6.3523	10.95 %
	$l_i = 0.05$	$u_i = 0.75$	7.1335	6.5398	8.32 %

The table highlights the objective function values, and the deviation from the optimal solutions. The results demonstrate that the genetic algorithm consistently produces high-quality solutions that are very close to the exact method's outcomes. These findings confirm the efficiency of our genetic algorithm, especially when dealing with larger, more complex portfolio selection problems, where an exact solution is computationally expensive or infeasible.

5- Conclusions and Future Research Directions

In this paper, we proposed an enhanced genetic algorithm to solve the mean-variance cardinality-constrained portfolio selection problem, incorporating novel crossover operators inspired by the

RAR operator. The complexity of real-world portfolio selection, including cardinality and floor-ceiling constraints, makes exact optimization approaches such as mixed-integer quadratic programming computationally expensive, especially for larger problem instances. Our genetic algorithm aims to provide a practical and efficient alternative for addressing these challenges. The experimental results demonstrated that the proposed genetic algorithm is not only effective in generating high-quality solutions but also significantly reduces computation times compared to the exact mixed-integer quadratic programming solver in GAMS. The algorithm consistently converged to near-optimal solutions across various scenarios with different cardinality and asset proportion constraints. This highlights its robustness and applicability in solving large-scale portfolio optimization problems, where exact methods struggle with scalability.

Finally, two areas are proposed for future research: future work could explore the integration of additional real-world factors, such as transaction costs or dynamic market conditions, to further enhance the applicability of the algorithm. Additionally, combining this genetic algorithm with other metaheuristics or hybrid approaches could provide further improvements in performance and solution quality.

References

Abdollahi Moghadam, M., Ebrahimi, S. B., & Rahmani, D. (2019). A two-stage robust model for portfolio selection by using goal programming. Journal of Industrial and Systems Engineering, 12(1), 1–17. https://www.jise.ir/article_76604.html

Aguilar-Rivera, R., Valenzuela-Rendón, M., & Rodríguez-Ortiz, J. J. (2015). Genetic algorithms and Darwinian approaches in financial applications: A survey. Expert Systems with Applications, 42(21), 7684–7697. https://doi.org/10.1016/J.ESWA.2015.06.001

Anagnostopoulos, K. P., & Mamanis, G. (2011). The mean–variance cardinality constrained portfolio optimization problem: An experimental evaluation of five multiobjective evolutionary algorithms. Expert Systems with Applications, 38(11), 14208–14217. https://doi.org/10.1016/J.ESWA.2011.04.233

Bagheriyan, M., Ghanbari, H., & Tohidi, H. (2023). The Impact of gold and oil prices on the Nasdaq Composite. 7th International Conference on Researches in Science & Engineering & 4th International Congress on Civil, Architecture and Urbanism in Asia.

Bavarsad Salehpoor, I., & Molla-Alizadeh-Zavardehi, S. (2019). A constrained portfolio selection model at considering risk-adjusted measure by using hybrid meta-heuristic algorithms. Applied Soft Computing, 75, 233–253. https://doi.org/10.1016/J.ASOC.2018.11.011

Bertsimas, D., & Shioda, R. (2009). Algorithm for cardinality-constrained quadratic optimization. Computational Optimization and Applications, 43(1), 1–22. https://doi.org/10.1007/S10589-007-9126-9/METRICS

Bienstock, D. (1996). Computational study of a family of mixed-integer quadratic programming problems. Mathematical Programming, Series B, 74(2 PART A), 121–140. https://doi.org/10.1007/BF02592208/METRICS

Borchers, B., & Mitchell, J. E. (1994). An improved branch and bound algorithm for mixed integer nonlinear programs. Computers & Operations Research, 21(4), 359–367. https://doi.org/10.1016/0305-0548(94)90024-8

Chang, T. J., Meade, N., Beasley, J. E., & Sharaiha, Y. M. (2000). Heuristics for cardinality constrained portfolio optimisation. Computers & Operations Research, 27(13), 1271–1302. https://doi.org/10.1016/S0305-0548(99)00074-X

Chang, T. J., Yang, S. C., & Chang, K. J. (2009). Portfolio optimization problems in different risk measures using genetic algorithm. Expert Systems with Applications, 36(7), 10529–10537. https://doi.org/10.1016/J.ESWA.2009.02.062

Crama, Y., & Schyns, M. (2003). Simulated annealing for complex portfolio selection problems. European Journal of Operational Research, 150(3), 546–571. https://doi.org/10.1016/S0377-2217(02)00784-1

Deng, G. F., Lin, W. T., & Lo, C. C. (2012). Markowitz-based portfolio selection with cardinality constraints using improved particle swarm optimization. Expert Systems with Applications, 39(4), 4558–4566. https://doi.org/10.1016/J.ESWA.2011.09.129

Doering, J., Kizys, R., Juan, A. A., Fitó, À., & Polat, O. (2019). Metaheuristics for rich portfolio optimisation and risk management: Current state and future trends. Operations Research Perspectives, 6, 100121. https://doi.org/10.1016/J.ORP.2019.100121

Fernández, A., & Gómez, S. (2007). Portfolio selection using neural networks. Computers & Operations Research, 34(4), 1177–1191. https://doi.org/10.1016/J.COR.2005.06.017

Fooeik, A. M. L., Ghanbari, H., Bagheriyan, M., & Mohammadi, E. (2022). Analyzing the effects of global oil, gold and palladium markets: Evidence from the Nasdaq composite index. Journal of Future Sustainability, 2(3), 105–112. https://doi.org/10.5267/J.JFS.2022.9.010

Frangioni, A., & Gentile, C. (2006). Perspective cuts for a class of convex 0-1 mixed integer programs. Mathematical Programming, 106(2), 225–236. https://doi.org/10.1007/S10107-005-0594-3/METRICS

Ghanbari, H., Fooeik, A. M. L., Eskorouchi, A., & Mohammadi, E. (2022). Investigating the effect of US dollar, gold and oil prices on the stock market. Journal of Future Sustainability, 2(3), 97–104. https://doi.org/10.5267/J.JFS.2022.9.009

Ghanbari, H., Safari, M., Ghousi, R., Mohammadi, E., & Nakharutai, N. (2023). Bibliometric analysis of risk measures for portfolio optimization. Accounting, 9(2), 95–108. https://doi.org/10.5267/J.AC.2022.12.003

Golmakani, H. R., & Fazel, M. (2011). Constrained Portfolio Selection using Particle Swarm Optimization. Expert Systems with Applications, 38(7), 8327–8335. https://doi.org/10.1016/J.ESWA.2011.01.020

Gulpinar, N., le Thi, H. A., & Moeini, M. (2010). Robust investment strategies with discrete asset choice constraints using DC programming. Https://Doi.Org/10.1080/02331930903500274, 59(1), 45–62. https://doi.org/10.1080/02331930903500274

Holland, J. h. (1975). Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control and artificial intelligence. University of Michigan Press.

Jobst, N. J., Horniman, M. D., Lucas, C. A., & Mitra, G. (2001). Computational aspects of alternative portfolio selection models in the presence of discrete asset choice constraints. Quantitative Finance, 1(5), 489–501. https://doi.org/10.1088/1469-7688/1/5/301

Kalayci, C. B., Ertenlice, O., & Akbay, M. A. (2019). A comprehensive review of deterministic models and applications for mean-variance portfolio optimization. Undefined, 125, 345–368. https://doi.org/10.1016/J.ESWA.2019.02.011

Kellerer, H., Mansini, R., & Grazia Speranza, M. (2000). Selecting Portfolios with Fixed Costs and Minimum Transaction Lots. Annals of Operations Research, 99(1–4), 287–304. https://doi.org/10.1023/A:1019279918596

Kendall, G., & Su, Y. (2007). Imperfect evolutionary systems. IEEE Transactions on Evolutionary Computation, 11(3), 294–307. https://doi.org/10.1109/TEVC.2006.887348

Konstantinou, C., Tzanetos, A., & Dounias, G. (2022). Cardinality constrained portfolio optimization with a hybrid scheme combining a Genetic Algorithm and Sonar Inspired Optimization. Operational Research, 22(3), 2465–2487. https://doi.org/10.1007/S12351-020-00614-1/METRICS

Lee, E. K., & Mitchell, J. E. (1997). Computational Experience of an Interior-Point SQP Algorithm in a Parallel Branch-and-Bound Framework Proceedings of High Performance Optimization Techniques, 1997, Springer Verlag.

Li, D., Sun, X., & Wang, J. (2006). OPTIMAL LOT SOLUTION TO CARDINALITY CONSTRAINED MEAN–VARIANCE FORMULATION FOR PORTFOLIO SELECTION. Mathematical Finance, 16(1), 83–101. https://doi.org/10.1111/J.1467-9965.2006.00262.X

Lin, C. C., & Liu, Y. T. (2008). Genetic algorithms for portfolio selection problems with minimum transaction lots. European Journal of Operational Research, 185(1), 393–404. https://doi.org/10.1016/J.EJOR.2006.12.024

Lin, D., Li, X., & Li, M. (2005). A genetic algorithm for solving portfolio optimization problems with transaction costs and minimum transaction lots. Lecture Notes in Computer Science, 3612(PART III), 808–811. https://doi.org/10.1007/11539902_99

Lwin, K., & Qu, R. (2013). A hybrid algorithm for constrained portfolio selection problems. Applied Intelligence, 39(2), 251–266. https://doi.org/10.1007/S10489-012-0411-7

Mansini, R., & Speranza, M. G. (1999). Heuristic algorithms for the portfolio selection problem with minimum transaction lots. European Journal of Operational Research, 114(2), 219–233. https://www.academia.edu/24866407/Heuristic_algorithms_for_the_portfolio_selection_problem_with_minimum_transaction_lots

Maringer, D., & Kellerer, H. (2003). Optimization of cardinality constrained portfolios with a hybrid local search algorithm. OR Spectrum, 25(4), 481–495. https://doi.org/10.1007/S00291-003-0139-1/METRICS

Markowitz, H. (1952). PORTFOLIO SELECTION*. The Journal of Finance, 7(1), 77–91. https://doi.org/10.1111/J.1540-6261.1952.TB01525.X

Metaxiotis, K., & Liagkouras, K. (2012). Multiobjective Evolutionary Algorithms for Portfolio Management: A comprehensive literature review. Expert Systems with Applications, 39(14), 11685–11698. https://doi.org/10.1016/J.ESWA.2012.04.053

Mitra, G., Kyriakis, T., Lucas, C., & Pirbhad, M. (2003). A review of portfolio planning: Models and systems. Advances in Portfolio Construction and Implementation, 1–39. https://doi.org/10.1016/B978-075065448-7.50002-1

Mittal, S. K., & Srivastava, N. (2021). Mean-variance-skewness portfolio optimization under uncertain environment using improved genetic algorithm. Artificial Intelligence Review, 54(8), 6011–6032. https://doi.org/10.1007/S10462-021-09966-2

Mohammadi, S. E., Mohammadi, E., Makui, A., & Shahanaghi, K. (2022). Investigating unpopularity of utility-based approaches in portfolio optimization; introducing an extension to the UTASTAR method. Journal of Industrial and Systems Engineering, 14(2), 298–321. https://doi.org/10.2307/1909829

Moral-Escudero, R., Ruiz-Torrubiano, R., & Suárez, A. (2006). Selection of Optimal Investment Portfolios with Cardinality Constraints. 2006 IEEE International Conference on Evolutionary Computation, 2382–2388. https://doi.org/10.1109/CEC.2006.1688603

Najafi, S. E., Nozari, H., & Edalatpanah, S. A. (2022). Artificial Intelligence of Things (AIoT) and Industry 4.0–Based Supply Chain (FMCG Industry). A Roadmap for Enabling Industry 4.0 by Artificial Intelligence, 31-41.

Nozari, H. (2024). Green Supply Chain Management based on Artificial Intelligence of Everything. Journal of Economics and Management, 46, 171-188.

Nozari, H., Ghahremani-Nahr, J., Fallah, M., & Szmelter-Jarosz, A. (2022). Assessment of cyber risks in an IoT-based supply chain using a fuzzy decision-making method. International Journal of Innovation in Management, Economics and Social Sciences, 2(1).

Radcliffe, N. (1991). Forma Analysis and Random Respectful Recombination. Proceedings of the Forth International Conference. https://ci.nii.ac.jp/naid/10015779400/

Sabar, N. R., Turky, A., Leenders, M., & Song, A. (2018). Multi-population genetic algorithm for cardinality constrained portfolio selection problems. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 10860 LNCS, 129–140. https://doi.org/10.1007/978-3-319-93698-7_10/TABLES/6

Sadjadi, S. J., Gharakhani, M., & Safari, E. (2012). Robust optimization framework for cardinality constrained portfolio problem. Applied Soft Computing, 12(1), 91–99. https://doi.org/10.1016/J.ASOC.2011.09.006

Schaerf, A. (2002). Local Search Techniques for Constrained Portfolio Selection Problems. Computational Economics, 20(3), 177–190. https://doi.org/10.1023/A:1020920706534/METRICS

Shaw, D. X., Liu, S., & Kopman, L. (2008). Lagrangian relaxation procedure for cardinality-constrained portfolio optimization. Http://Dx.Doi.Org/10.1080/10556780701722542, 23(3), 411–420. https://doi.org/10.1080/10556780701722542

Shoaf, J., & Foster, J. A. (1996). The efficient set GA for stock portfolios. Proceedings of the Decision Science Institute, Orlando, 571–573. https://doi.org/10.1109/ICEC.1998.699758

Soleimani, H., Golmakani, H. R., & Salimi, M. H. (2009). Markowitz-based portfolio selection with minimum transaction lots, cardinality constraints and regarding sector capitalization using genetic algorithm. Expert Systems with Applications, 36(3), 5058–5063. https://doi.org/10.1016/J.ESWA.2008.06.007

Speranza, M. G. (1996). A heuristic algorithm for a portfolio optimization model applied to the Milan stock market. Computers & Operations Research, 23(5), 433–441. https://doi.org/10.1016/0305-0548(95)00030-5

Vielma, J. P., Ahmed, S., & Nemhauser, G. L. (2008). A Lifted Linear Programming Branch-and-Bound Algorithm for Mixed-Integer Conic Quadratic Programs. Https://Doi.Org/10.1287/Ijoc.1070.0256, 20(3), 438–450. https://doi.org/10.1287/IJOC.1070.0256

Woodside-Oriakhi, M., Lucas, C., & Beasley, J. E. (2011). Heuristic algorithms for the cardinality constrained efficient frontier. European Journal of Operational Research, 213(3), 538–550. https://doi.org/10.1016/J.EJOR.2011.03.030

Yu, L., Wang, S., & Lai, K. (2006). An integrated data preparation scheme for neural network data analysis. IEEE Transactions on Knowledge and Data Engineering, 18(2), 217–230. https://doi.org/10.1109/TKDE.2006.22

Zhu, H., Wang, Y., Wang, K., & Chen, Y. (2011). Particle Swarm Optimization (PSO) for the constrained portfolio optimization problem. Expert Systems with Applications, 38(8), 10161–10169. https://doi.org/10.1016/J.ESWA.2011.02.075