Svedberg Open
DISSEMINATION OF KNOWLEDGE

International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: https://www.svedbergopen.com/

Research Paper Open Access

Multi-Objective Evolutionary Optimization Algorithm for Profit-Maximization and Risk Minimization in Business Operations

Ergashev Rasulbek Sokhib ugli[1*], Krishnamurthy Kumar[2], Dr.C. Nallusamy[3], Roohee Khan[4], Yanglish Kosimova[5], Kattakul Kinjaev[6]

[1*] Turan International University, Namangan, Uzbekistan. E-mail: rasulbek.ergashev.00@gmail.com, https://orcid.org/0009-0000-2076-6642

[2]Department of Nautical Science, AMET University, Kanathur, Tamilnadu,India. E-mail: captrkkumar@ametuniv.ac.in, https://orcid.org/0009-0005-2439-0976

[3]Professor, Department of Information Technology, K.S.Rangasamy College of Technology, Tiruchengode, India. E-mail: nallusamyc@ksrct.ac.in, https://orcid.org/0000-0001-6100-0088

[4]Assistant Professor, Kalinga University, Naya Raipur, Chhattisgarh, India, E-mail: ku.roohee.khan@kalingauniversity.ac.in, https://orcid.org/0009-0009-8960-8840

[5]Teacher, Department of Physics, Jizzakh State Pedagogical University Jizzakh, Uzbekistan. E-mail: kyanglish@gmail.com https://orcid.org/0009-0003-0953-9731

[6]Lecturer, Department of Finance and Tourism, Termez University of Economics and Service, Termez, Uzbekistan. E-mail: samurai6356693@gmail.com, https://orcid.org/0009-0002-9315-1395

*Corresponding author: Email: rasulbek.ergashev.00@gmail.com

Abstract

The management of business operations is becoming more and more focused on achieving simultaneous optimization of conflicting objectives (profit maximization, risk minimization, regulatory compliance, and sustainability) while also managing complex operational constraints. Traditional single-objective optimization and multi-criteria optimization based on scalarization do not utilize the inherent trade-off structure present in these types of situations. Developed a Multi-Objective Evolutionary Optimization (MOEO) Algorithm for Business Operations (MOEO-BO) that uses the NSGA-III non-dominated sorted approach to optimize the following six objectives at once: net profit (f_1), Conditional Value-at-Risk (f_2), operational risk index (f_3), Return on Invested Capital (f_4), ESG compliance (f_5), and exposure to market volatility (f_6). The algorithm has been validated on 65 real-world business units within six industrial sectors, yielding a Hypervolume indicator of 0.847 and an Inverted Generational Distance of 0.0028, outperforming both NSGA-III (HV = 0.793), NSGA-II (HV = 0.761), MOEA/D (HV = 0.778), and SPEA2 (HV = 0.742). The MOEO-BO algorithm, when applied to businesses in the manufacturing, retail, financial services, healthcare, technology, and logistics sectors, resulted in an aggregate profit increase of 37.5%, with an overall risk reduction of 39.4% from baseline operational strategies. Ablation studies demonstrated that each of the following attributes contributed significantly to the solution quality: maintaining diverse reference points (+10.1% Hypervolume, 73.4% Generational Distance improvement), Augmented Lagrangian repairs (+3.9% Hypervolume), and using a six-objective formulation (+31.2% over the single-objective formulation). The MOEO-BO algorithm enables enterprise decision makers to create a Pareto-optimal strategy frontier, providing explicit means for managing trade-offs in the face of uncertainties.

Keywords: Multi-Objective Evolutionary Optimization, Pareto Optimality, NSGA-III, Business Operations Management, Profit Maximization, Risk Minimization, ESG Compliance; Augmented Lagrangian, Simulated Binary Crossover.

This is an open access article under CC BY 4.0, allowing unrestricted use with proper attribution, a license link, and indication of any changes made.

1. Introduction

Managing business operations is a multi-faceted challenge for firms today. Leaders must manage conflicting interests, such as maximizing profits (the primary fiduciary responsibility to shareholders), minimizing risks (in each of operational, financial, market, and regulatory dimensions), as well as other constraints imposed by environmental, social, and governance (ESG) requirements, regulatory capital, supply chains, etc., thus making the optimization process increasingly difficult.

The Classical method of addressing this complexity is through Scalarization - Transforming many objectives into one weighted aggregated objective [3], which, while computationally tractable, results in a single operating point that is only as good as the user's specification of weights, a specification generally not well specified. Goal programming [12][21] has the same drawbacks, as it requires precise aspiration levels that may be impossible or suboptimal to meet.

Multiobjective evolutionary algorithms (MOEAs) are a fundamentally new way of solving problems, where together we evolve a population of candidate solutions at the same time. They converge to an approximation of all the points on the entire Pareto front in one run, while at the same time providing stakeholders a clear, geometric trade-off surface with which to select points of operation, based on management's judgment, regulations, or analysis of different scenarios. The benefit of this approach is that it avoids the necessity of performing the optimization for each trade-off preference [5] [8].

MOEA has been an established way to manage engineering design and logistics activities; however, its use for managing operations within companies has not reached the same level. The current applications of MOEAs to operations management are generally very specific to an industry, such as those found in supply chain or production scheduling, and do not account for the same level of joint optimization that is necessary for SBOs. The paper improves the understanding of the ways that strategic business unit optimization can be improved using **multi-objective** evolutionary algorithms [4] [9] [11].

1.1 Research Contributions

This paper makes five principal contributions:

1. A full MOEA system capable of supporting corporate decisions in six business-related objective areas, as well as Augmented Lagrangian constraint repair and reference-point diversity maintenance using NSGA-III.
2. First-ever successful simultaneous optimization of profit, CVaR, operational risk, ROIC, ESG, and market volatility for multi-SBU (Strategic Business Unit) operations.
3. Evaluation of 65 actual businesses within 6 different industries with five (5) years of historical financial data (2018-2023) while considering the effect of COVID-19.
4. Comparison to 7 competitive algorithms (NSGA-II, MOEA/D, SPEA2, ε-MOEA, all fair static scalar methods) using 4 types of Pareto-quality metrics.
5. Quantified contribution of components to solution quality performed through a systematic component-based removal experiment methodology.

1.2 Scope and Paper Organization

Multi-Objective Evolutionary Optimization for Business Optimization (MOEO-BO) is an evolutionary computation (EC) method designed for multi-objective business optimization (i.e., at the "business unit level" - i.e., resource allocation across product types, capital investment, pricing/risk hedge strategy, and allocation of work force) that occur on a weekly to quarterly time scale with evolutionary run times of minutes to hours being acceptable for operational purposes. Intraday trading or real-time process control is not included. Section 2 summarizes related research. Section 3 formalizes the multiple-objective business optimization problem. Section 4 describes the MOEO-BO algorithm. Section 5 provides details on the experimental setup and measurement techniques. Section 6 provides results. Section 7 discusses the limitations/implications for managers of this research. Section 8 provides closing remarks.

2. Literature Review

2.1 Classical Multi-Objective Optimization in Business

Business multi-objective optimization started with linear programming and the welfare economic concept of Pareto efficiency [13][22]. Satisficing was defined by means of aspiration levels for multiple objectives. Preference elicitation was provided with an axiomatic basis by multi-attribute utility theory, while structured pairwise comparisons were produced through the Analytic Hierarchy Process. Both are commonly employed today; however, both have the same main drawback, i.e., reducing the trade-off surface to a single point.

MOLP has been applied in operations management for production planning [18], supply chain design [16], and revenue management. However, many problems in practice are more often nonlinear (due to scales of economy, risk nonlinearity & portfolio effects), non-convex (due to combinatorial resource allocation), and high-dimensional (due to large SBU portfolio size), which result in suboptimal or infeasible MOLP solutions.

2.2 Evolutionary Multi-Objective Algorithms

MOEAs originated from VEGA & MOGA [10]. The original version of the NSGA uses non-dominated sorting and the distance of crowding. NSGA-II improved upon this method by using an $O(MN^2)$ sorting method to improve sorting efficiency. The NSGA-III extended this basic framework to a many-objective ($k > 3$) optimization problem using a structured reference point on a normalized hyperplane and therefore had superior diversity in the set of solutions for individuals with four or more objectives; the diversity principle of NSGA-III has specific applications in solution formulation (six objectives).

MOEA/D decomposes multi-objective problems into scalar subproblems using Chebyshev aggregation (that is, maximizing the use of the greatest distance from the ideal to ensure diversity) and therefore provides less robust diversity in solutions, but better computational efficiency [7][23]. The SPEA2 uses multiple methods of assigning fitness (based on the number of dominated individuals and the density of dominated individuals) to produce individuals with a consistent level of fitness. The ε-MOEA adapts its use of ε-dominance archiving to continually approximate the set of Pareto efficient solutions. Based on all of this, the practical comparisons of NSGA-III vs. NSGA-II consistently indicate superior performance on standard test suites for $k \geq 4$ objectives, thus demonstrating that the NSGA-III serves as the most suitable foundation for work.

2.3 MOEAs Applied to Business and Finance

Different types of businesses utilize various approaches of Multi-Objective Evolutionary Algorithms (MOEAs), such as improving the performance of supply chains [2] or optimizing portfolio allocations, production schedules, and energy procurement **tasks [**1]. The Non-Dominated Sorting Genetic Algorithm II (NSGA-II) algorithm was used to model a multi-period/multi-objective supply chain design with total cost and customer satisfaction as the two objectives. Some works within the supply chain domain incorporate risk factors into their supply chain networks via essentially the use of chance constraints [14], [15]. However, there is currently no single unified and validated framework that jointly optimizes the six business-domain objectives of profit, conditional value-at-risk (CVaR), operational risk, capital efficiency, ESG compliance, and market volatility that can be applied in multiple industries [20]. This lack of a joint decision-making framework to optimize all six objectives is the primary gap that MOEO-BO addresses.

3. Problem Formulation

3.1 Decision Variables

The decision space, n, is represented by $x=[x_1, x_2,...,x\_n]\top\in\Omega\subseteq\mathbb{R}^n$ (the decision vector), which contains the values of the n business variables used to describe resource allocations (in terms of capital investment), production, pricing and hedging in order to create a portfolio of NSBU Strategic Business Units; The feasible space, Ω, is defined by the limit of the constraints given in Section 3.3. For exploratory analysis, using factors between 48 and 120 to represent decision factors, based on the complexity of the sectors.

3.2 Objective Functions

The six objective functions are formalized in the table below. The MOEO-BO optimizes F(X) = [f1(x), f2(x), f3(x), f4(x), f5(x), f6(x)]. Two of these functions (f1 and f4) are maximized but negate to perform minimization. The other four (f2, f3, f6) is minimized. Thus, the problem is:

$$\boldsymbol{min\_\{x \in \Omega\}\ F(x) = [-f_1(x),\quad f_2(x),\quad f_3(x),\quad -f_4(x),\quad -f_5(x),\quad f_6(x)]}$$

The MOEO-BO problem formulation has six objective functions. The CVaR (Conditional Value At Risk) premium is set to 0.95 and accounts for the 95th percentile tail risk. The weights, λk, for the ESG objectives correspond to sector regulatory mandates. Therefore, objectives that are specified as "Maximize" in the specification are negated by the decision to minimize in the algorithm (Table 1).

Table 1: Multi-Objective Business Problem — Objective Function Definitions

Objective	Symbol	Mathematical Form	Optimization	Business Meaning
Net Profit	$f_1(x)$	$\Sigma_i (p_i - c_i) \cdot x_i - FC$	Maximize	Total margin minus fixed overhead
Conditional Value-at-Risk	$f_2(x)$	$CVaR\_\alpha = E[L \mid L \geq VaR\_\alpha]$	Minimize	Expected loss in worst α% scenarios
Operational Risk Index	$f_3(x)$	$\Sigma_j w_j \cdot r_j(x) / \Sigma_j w_j$	Minimize	Weighted composite of process risk scores
Capital Efficiency	$f_4(x)$	$ROIC = NOPAT / Invested\ Capital$	Maximize	Returns generated per unit of invested capital
ESG Compliance Score	$f_5(x)$	$ESG(x) = \Sigma_k \lambda_k \cdot g_k(x)$	Maximize	Regulatory & sustainability compliance index
Market Volatility Exposure	$f_6(x)$	$\sigma\_p(x) = \sqrt{(x^T \Sigma x)}$	Minimize	Portfolio-weighted earnings volatility

3.3 Constraint System

Eight constraints define the Omega feasible region (Mistakes are highlighted in the example): three types of soft constraints and five types of hard constraints. The hard constraints are evaluated using the domain-specific repair operator (Line 14 of Algorithm 1) after applying crossover and mutation. The soft constraints were added to the overall fitness evaluation via an Augmented Lagrangian penalty function. $\boldsymbol{\Phi(x, \lambda, \rho) = F(x) + \Sigma_j \left[\lambda_j \cdot max\left(0, g_j(x)\right) + \left(\frac{\rho}{2}\right) \cdot max\left(0, g_j(x)\right)^2\right]}$

*where λ_j are Lagra*nge multiplier estimates updated every T_ρ=50 generations, and ρ is the penalty parameter. This formulation avoids the infeasibility bias of pure death penalty methods while maintaining meaningful gradient information near constraint boundaries (Table 2).

Table 2: Business Constraint Classification and Enforcement Methods

Constraint Type	Mathematical Form	Category	Penalty Method
Capital Budget	$\Sigma_i c_i \cdot x_i \leq B\_max$	Hard / Inequality	Death penalty (infeasible elimination)
Regulatory Compliance	$ESG(x) \geq ESG\_min$	Hard / Inequality	Death penalty
Operational Capacity	$x_i \leq cap\_i \ \forall i \in I$	Hard / Box	Clipping in repair operator
Non-negativity	$x_i \geq 0 \ \forall i$	Hard / Box	Absolute value reflection
Risk Exposure Cap	$CVaR\_\alpha(x) \leq R\_max$	Soft / Inequality	Augmented Lagrangian penalty
Market Share Floor	$\Sigma_i x_i \geq D\_min$	Soft / Inequality	Augmented Lagrangian penalty
Diversification	$x_i / \Sigma_j x_j \leq 0.40 \ \forall i$	Soft / Proportion	Quadratic penalty $\varphi(x)^2$
Integer Product Mix	$x_i \in \mathbb{Z}$ for discrete units	Hard / Integrality	Rounding with feasibility check

Eight constraint types applied in MOEO-BO. Hard constraints are enforced via repair operators applied post-crossover. Soft constraints use Augmented Lagrangian penalization with adaptive multiplier updates.

4. MOEO-BO Algorithm Design

4.1 Algorithm Overview

MOEO-BO develops the NSGA-III by providing four domain-specific features through its generational evolutionary process: (1) mixed optimization directions within a six-dimensional business objective vector, (2) using an Augmented Lagrangian constraint repair operator instead of using feasibility-tournament selection, (3) adapting reference point positions using perturbations to keep diversity when objectives converge and become correlated, and (4) archiving elite populations consisting of the top 100 non-dominated solutions from multiple generations. The full pseudocode for these algorithms is outlined in Algorithm 1.

Algorithm 1: MOEO-BO — Multi-Objective Evolutionary Optimization for Business Operations

Input: Population size N, max generations G, crossover rate p_c, mutation rate p_m, business objectives

F = {f_1, f_2, ..., f_k}

Output: Pareto-optimal set P* of non-dominated business strategy solutions

1: Initialize population P_0 ← random_initialize(N) within feasible decision space Ω

2: Evaluate objectives F(x) for all x ∈ P_0

3: Rank P_0 using non-dominated sorting (NSGA-III front assignment)

4: for g = 1 to G do

5: Q_g ← tournament_selection(P_{g-1}, k=2) with Pareto rank & crowding distance

6: Q_g ← SBX_crossover(Q_g, p_c=0.90, η_c=20)

7: Q_g ← polynomial_mutation(Q_g, p_m=1/n, η_m=20)

8: Evaluate F(x) for all x ∈ Q_g (profit, CVaR, ESG metrics)

9: R_g ← P_{g-1} ∪ Q_g (combined parent + offspring pool)

10: Apply non-dominated sorting to R_g → fronts {F_1, F_2, ..., F_l}

11: P_g ← ∅; i ← 1

12: while |P_g| + |F_i| ≤ N do P_g ← P_g ∪ F_i; i ← i + 1 end while

13: Fill remainder of P_g from F_i using crowding-distance sort (descending)

14: Apply business constraint repair operator (Equation 6) to infeasible solutions

15: end for

16: P* ← non_dominated(P_G) — extract final Pareto-optimal front

17: return P*

Complete Pseudocode for MOEO-BO. Simulated Binary Crossover is applied using η_c=20. Polynomial mutation with pm=1/n. Repair of violated constraints in line 14 is done using clipping in the box constraints method and Augmented Lagrangian updates for the soft constraint violations. The steps that are specific to NSGA-III is highlighted in grey lines.

4.2 Genetic Operators

4.2.1 Simulated Binary Crossover (SBX)

The SBX operator generates children by mimicking the distribution of solutions generated by single-point crossover in the binary-encoded space of variables directly in the solution space of real-valued variables. Two parents' solutions, $x^{(1)}$ and $x^{(2)}$, are used to produce two children, $y^{(1)}$ and $y^{(2)}$, according to:

$$\boldsymbol{y^{(1,\,2)} = 0.5 \cdot \left[\left(1 \pm \beta_q\right) \cdot x^{(1)} + \left(1 \mp \beta_q\right) \cdot x^{(2)}\right]}$$

Increasing the value of η_c moves offspring towards the P, while decreasing the value of η_c results in greater crossover disruption (i.e., exploration). In the first 200 generations conditionally adjust η_c between 10 and 25 to achieve a balance between the objective of maximizing exploration in early generations and the objective of maximizing convergence in late generations.

4.2.2 Polynomial Mutation

With a probability of p_m=1/n (where n is the number of decision variables), each decision variable is altered independently by applying polynomial mutation through the application of an amendment (δ_q), which is chosen from a polynomial distribution with an index of η_m=20, allowing for bounded mutations to occur in the range of bounds (lb_i and ub_i) for each variable.

$$\boldsymbol{y_i = x_i + \delta_q \cdot (ub_i - lb_i)}$$

To prevent the box-constrained variables from being out of range, the operator respects the constraints from the start and thus reduces hard constraint repair efforts (Section 3.3).

4.2.3 Constraint Repair Operator

After applying crossover and mutation, any infeasible individuals is repaired using a domain-specific operator (Algorithm 1, line 14), whereby box-violated variables is clipped at their defined bounds. All positively allocated budgets that violate the budget constraint is proportionally scaled down to result in feasible solutions. All integrality violations for discrete variables is repaired via rounding in order to find the nearest feasible integer, at which time a check is performed to confirm that the allocation is still within the budget. If hard constraints relating to regulatory or ESG concerns cannot be repaired by means of a local adjustment, then the individual is re-initialized to the nearest constraint boundary, which is similar to the boundary repair procedure used in [18].

4.3 NSGA-III Non-Dominated Sorting and Reference Points

To conduct non-dominated sorting, the combined population RG = Pg-1 ∪ Qg is divided into multiple Pareto Fronts {F1,F2,...,Fl} by performing the standard dominance test and extending it to the case where maximization and minimization share the same objective direction (i.e., to find F1 as the current best solution to the actual Pareto Front P*).

Reference points are uniformly distributed across the six-dimensional Objective Space (M = 6) on a normalized hyperplane and are generated using [6] systematic reference point generation with H = 12 divisions per objective, or a total of C(H+M-1, M-1) = 6188 reference points. Each reference point attracts nearby solutions to that selection point and allows for geometric diversity across the six dimensions of the Pareto Front approximation. Reference Point Adaptation Procedure is used to create adaptive perturbations around reference points in directions of low solution density (A) every 100 generations to avoid premature convergence on a single subset of the Pareto Front.

4.4 Complexity and Convergence

The main expense associated with computation is the objective function evaluations: there are N=300 solutions, and each evaluation of a solution has six business metrics that involve historical simulation across three years. The empirical observations show that evaluating these solutions costs about 0.8–0.9 seconds per generation using a 16-core server (Intel Xeon Gold 6342, 2.80 GHz); thus, the total time to run G=500 generations is approximately 400–450 seconds. The cost of the non-dominated sorting operation is $O(M \cdot N^2)$ per generation, where M=6 and N=300, which represents less than 2% of the total runtime. Theoretical convergence guarantees exist for MOEAs under weak assumptions; however, empirical monitoring over 50-generation windows using hypervolume stagnation demonstrates convergence to a stable state by generation 350 for all six sectors in this study.

5. Experimental Setup

5.1 Dataset and Business Unit Portfolio

The empirical validation uses anonymized financial and operational data for 65 strategic business units (SBUs) drawn from six industry sectors across Europe, North America, and Latin America, provided under data-sharing agreements with seven multinational corporations. Data spans fiscal years 2018–2023 (six years), encompassing the COVID-19 disruption (2020) and the subsequent inflationary environment (2022–2023). Sector distribution: Manufacturing N=12, Retail & Distribution N=8, Financial Services N=15, Healthcare Operations N=10, Technology & SaaS N=9, Logistics & Supply Chain N=11.

Each of the SBUs has a list of features that includes P&L statements (quarterly), balance sheet position, historical returns for the last 5 years (used to estimate CVaR), operational KPIs (productivity, error rate, cycle time), ESG audit score, and how sensitive they are to market risk (Beta and correlation to the sector). In addition to this, decision variables is documented (between 48 and 120 per SBU), which encode capital allocation ratios, product-line mix weights, hedging ratios, and distribution of headcount across different business functions.

5.2 Evaluation Metrics for Pareto Quality

Four standard Pareto quality indicators are computed for algorithm comparison:

- Hypervolume (HV) is the volume of the objective space that is dominated by the set of solutions in the Pareto approximation with respect to a reference point. The higher the value of HV, the better. The reference point for HV is set at 1.1 times the nadir vector along each of the three objective dimensions.
- The Inverted Generational Distance (IGD) is the average distance of the members of the true Pareto front (approximated by a 10,000 solution Monte Carlo reference set) to the nearest solution in the approximation. The lower the value of IGD, the better.
- Generational Distance (GD) is similar to IGD, but it measures the average distance from each of the solutions in the approximation to the nearest existing member of the true Pareto front. GD also penalizes dominated solutions. The lower the value of GD, the better.
- Spread (Δ) is a measure of the degree to which the solutions in the Pareto front are spaced along the boundary of the Pareto front. The average inter-solution distance, normalized to account for the extreme solutions, is used to compute Δ. The greater the value of Δ, the better.

5.3 Baseline Algorithms

Seven comparative baselines (NSGA-III, NSGA-II, MOEA/D, SPEA2, ε-MOEA Weighted Sum Scalarization (equal weights, conducted using L-BFGS-B), and Goal Programming (conducted using the simplex method; aspirations at 80th percentile of historical benchmarks)) are used in the experiments to compare MOEO-BO to. In the experiments, all bases have the same population size (N = 300) and the same number of generations (G = 500) as MOEO-BO. For the purpose of controlling for the selection method, the SBX and polynomial mutation parameters used by the epochs for NSGA-II, NSGA-III, and SPEA2 are the same as those used by the epochs for MOEO-BO. A Wilcoxon signed-rank test is used to determine the statistical significance of the differences between HV and IGD; this is controlled for by applying a Bonferroni correction, and a sample size of 30 independent runs is conducted for each of these tests.

5.4 Hyperparameter Configuration

As shown below in table 3, after performing OFAT experiments with a 30% holdout of all business units, the final hyperparameters were determined based on sensitivity analysis using a combination of three different levels of sensitivity (High=>5%, Medium=2-5%, and Low<2%).

Table 3: MOEO-BO Hyperparameter Configuration and Sensitivity Analysis

Parameter	Symbol	Value	Range Explored	Sensitivity
Population Size	N	300	[100, 500]	High
Max Generations	G	500	[200, 1000]	Medium
Crossover Rate	p_c	0.90	[0.70, 0.95]	Medium
SBX Distribution Index	η_c	20	[10, 40]	Low
Mutation Rate	p_m	1/n	Auto (1/dim)	High
Polynomial Mutation Index	η_m	20	[5, 50]	Low
Tournament Size	k	2	[2, 5]	Medium
Reference Points (NSGA-III)	H	12	[6, 20]	High
Lagrangian Penalty Weight	ρ	1.0	[0.1, 10]	Medium
Penalty Update Frequency	T_ρ	50 gen	[10, 100]	Low
Archive Size (Elitism)	|A|	100	[50, 200]	Low

Final hyperparameter values after OFAT sensitivity analysis on holdout business units. Red = High sensitivity (requires careful tuning); Amber = Medium; Green = Low sensitivity (robust to choice within range).

6. Results

6.1 Pareto Quality Comparison

Table 4 shows the mean of the four quality measures (Pareto front indicators) for the eight algorithms analyzed in this study, summarised over 30 independent runs on the whole dataset of 65 SBUs (65 Sample Business Units). MOEO-BO outperformed all other algorithms for the four measures of performance: HV = 0.847 (6.8% better than NSGA-III, 11.3% better than NSGA-II), IGD = 0.0028 (33.3% worse than NSGA-III), GD = 0.0031 (34% worse than NSGA-III), and Spread = 0.912 (4.7% better than NSGA-III). After applying the Bonferroni adjustment (Wilcoxon test), all pairwise comparisons were statistically significantly different at $p < 0.01$. Although the weighted-sum scalarization technique had a substantial timing advantage (44 sec versus 284 sec), HV = 0.681 (19.6% less than the best of the MOEO-BO algorithm) demonstrates the great degree to which scalarizations fail to provide thorough representations of the Pareto surface for a problem with six objectives.

Table 4: Pareto Quality Indicator Comparison — All Algorithms (30 Runs, 65 Business Units)

Algorithm	HV (↑)	GD (↓)	Spread (↑)	IGD (↓)	Profit (M$)	CVaR Reduc.	Runtime (s)
MOEO-BO (Proposed)	**0.847**	**0.0031**	**0.912**	**0.0028**	**$142.7M**	**−38.4%**	**284**
NSGA-III Baseline	0.793	0.0047	0.871	0.0042	$128.3M	−29.1%	261
NSGA-II	0.761	0.0059	0.834	0.0055	$119.6M	−24.3%	198
MOEA/D	0.778	0.0052	0.848	0.0049	$123.1M	−26.8%	312
SPEA2	0.742	0.0071	0.809	0.0067	$114.2M	−21.7%	247
ε-MOEA	0.729	0.0083	0.791	0.0078	$108.9M	−18.4%	219
Weighted Sum Scalarize	0.681	0.0124	0.723	0.0119	$98.4M	−11.2%	44
Goal Programming	0.658	0.0138	0.701	0.0131	$93.7M	−9.6%	38

The following table summarizes all results in terms of measures of performance means over thirty independent runs of the various methods used to solve the two-objective problem. The hypervolume (HV), inverted generational distance (IGD), generational distance (GD), and spread of those possible solutions are the four measures used to compare the comparative performances between MOEO-BO and NSGAIII at the $p < 0.01$ level (Wilcoxon signed rank test, Bonferroni corrected). The row(s) in bold typeface are those associated with the proposed approach of MOEO-BO.

6.2 Business Performance by Sector

Table 5 illustrates MOEO-BO's out-of-sample business performance by sector (6 sectors in total). The metric of performance used in the comparison is the Pareto solution from the trade-off point at the 50th percentile (i.e., a balanced profit vs. risk preference) compared to each partner (corporate) performance prior to the deployment of MOEO-BO as assessed during their validation period.

Table 5: Out-of-Sample Business Performance by Industry Sector — MOEO-BO vs Baseline Strategy

Business Sector	Baseline Profit	MOEO-BO Profit	Profit Gain	Risk Reduction	ESG Score Δ	ROIC Improvement
Manufacturing (N=12)	$84.2M	$118.6M	+40.8%	−41.2%	+18.3 pts	+12.4%
Retail & Distribution (N=8)	$31.7M	$44.9M	+41.6%	−36.7%	+14.1 pts	+9.8%
Financial Services (N=15)	$203.4M	$271.8M	+33.6%	−44.8%	+21.7 pts	+15.2%
Healthcare Operations (N=10)	$67.3M	$93.1M	+38.3%	−38.9%	+29.4 pts	+11.7%
Technology & SaaS (N=9)	$118.9M	$164.3M	+38.2%	−33.1%	+16.2 pts	+14.6%
Logistics & Supply Chain (N=11)	$52.6M	$74.8M	+42.2%	−43.5%	+11.8 pts	+10.3%
AGGREGATE (All Sectors)	**$558.1M**	**$767.5M**	**+37.5%**	**−39.4%**	**+18.6 pts**	**+12.3%**

This graph shows how the companies perform at the 50th percentile for each business metric. It compares profitability (defined as the average annualized amount of profit over the five-year period) vs risk (defined as the CVaR improvement). The ESG performance is converted to a scale of 0 to 100 based on improvement in the triangle. Additionally, ROIC is improved compared to the average value of the sector based on the improvement of the triangle.

The Profitability Improvement for Financial Services is the largest absolute dollar increase ($68.4M) among all sectors; this indicates the capital-allocation flexibility of the sector and validates the positive impact of applying CVaR-constrained portfolio optimization to the amount of risk capital required. Logistics & Supply Chain has had the highest relative increase in profitability (+42.2%) as a result of MOEO-BO optimizing jointly, at the same time, route decision making, total inventory levels, and supplier diversification, all of which are simultaneously limited by available capacity. Finally, Healthcare Operations had the greatest improvement in ESG scores (+29.4 pts). This change in ESG scores is in alignment with the increased scrutiny placed on this sector by regulations surrounding the reporting of ESG metrics.

6.3 Ablation Study

The contribution of the MOEO-BO components to the overall performance, as summed up in table 6, was assessed by performing systematic removal experiments. The three components contributing most substantially to the overall performance of the algorithm were the six-objective formulation (whereby using a profit only base-case single objective formulation reduced hypervolume by 31.2%), NSGA-III reference points (whereby reverting to an NSGA-II-based selection method reduced hypervolume by 10.1%), and the polynomial based mutation strategy (where replacing with Gaussian mutation resulted in a −5.2% reduction in hypervolume). The Augmented Lagrangian Repair Operator provided a 3.9% increase in hypervolume; while this increase is small when combined with the aforementioned improvements, it becomes extremely large for Financial Services and Healthcare industries, where regulatory constraints restrict selection to >80% of potential candidate solutions.

Table 6: Ablation Study — Marginal Contribution of MOEO-BO Components

Configuration	HV (↑)	IGD (↓)	Profit (M$)	CVaR Δ	Δ Sharpe vs Full
Full MOEO-BO (Proposed)	**0.847**	**0.0028**	**$142.7M**	**−38.4%**	**—**
w/o NSGA-III Reference Points → NSGA-II	0.761	0.0055	$119.6M	−24.3%	−10.1%
w/o Augmented Lagrangian Repair	0.814	0.0041	$131.2M	−29.7%	−3.9%
w/o ESG Objective (f_5 removed)	0.831	0.0034	$148.1M	−35.6%	−1.9%
w/o CVaR Objective (f_2 removed)	0.822	0.0038	$151.4M	−18.2%	−2.9%
w/o Polynomial Mutation → Gaussian	0.803	0.0049	$124.8M	−27.1%	−5.2%
Single-Objective (Profit Only, GA)	0.583	0.0189	$158.9M	+12.6%	−31.2%
Single-Objective (Risk Only, GA)	0.541	0.0214	$79.3M	−51.2%	−36.1%

Thirty independent runs reveal the effect of each component of the full MOEO-BO system on the total system performance, with Δ Sharpe being the difference between the performance of the full model and the reduced model's HV. Single-objective GA based approaches optimize profit or risk only and illustrate the inherently constrained trade-off of a multi-objective formulation to balance this trade-off.

A particularly interesting result from the analysis conducted is that when comparing a single objective, i.e., profit only, the highest raw profit yielded was $158.9M; however, this came with a positive CVaR (+12.6% worse) as a result of the unconstrained maximization of profits, which increases tail risk. Conversely, a risk-only optimization results in a 51.2% reduction in the CVaR but at the expense of $63.4M in profits or −42% as compared to the unconstrained optimization of profits above. The MOEO-BO reaches a balanced Pareto front solution that yields $142.7M in profit with a CVaR reduction of −38.4%; therefore, this provides the best risk-adjusted return of all MOEO-BO solutions across a continuum of performance measures.

7. Discussion

7.1 Managerial Decision Support Framework

For the business executive, MOEO-BO's primary value is in providing a way to determine a structured set of profit/risk tradeoffs (the Pareto-optimal frontier or P* – as opposed to relying on negotiation between individual functional areas) by utilizing an analytically-based surface of profit/risk tradeoffs. The way to understand the Pareto front is as a two-dimensional risk/return scatter plot (similar to the Markowitz efficient frontier), with the color of each point representing ESG and ROIC performance measures, which allows the board or investment committee to operate at a specific point that meets both their fiduciary responsibility and risk tolerance level.

Table 3 illustrates a 37.5% increase in the total profit margin and a 39.4% reduction in risk from the new method as it is applied against existing operational methods, not from the theoretical optimum. This indicates that corporations are using resources in a manner that significantly underutilizes available capacity, possibly due to functional optimizations being performed in silos, legacy plans that are not based on current realities, or not using sufficient quantitative tools in the strategic planning process. The largest improvement can be in sectors where there are many complex operations and where cross-functional relationships have an impact (i.e., Logistics, Manufacturing), in line with the idea that using an evolutionary search to solve such problems give the greatest benefit compared to using heuristics developed by humans because of the constraints imposed on their search and many dimensions to search within.

7.2 Sensitivity to Market Regime

Evaluating algorithm robustness across the multiple phases of market regimes relies on the evaluation dataset that incorporates both the COVID-19 disruption period (2020) and the inflation shock (2022). In this dataset, MOEO-BO outperformed NSGA-III, particularly during the acute COVID shock phase (q1-q2 2020), where the Sharpe ratio was improved by 29.4% due to the tail risk exposure (CVaR) constraints placed on the investment objective. Furthermore, the ESG investment objective (f_5) impacted SBU's in the energy sector with respect to their ability to withstand high carbon exposure due to regulatory pressures on profitability, limiting their profitable configurations. The explicit ESG formulation associated with MOEO-BO thus provided an earlier indication of high-profit, compliant configurations than any other method without an explicit ESG formulation.

7.3 Limitations

It should be noted that these three main limitations are significant. First, the business objective functions (Table 1) rely on accurate estimation of input parameters (expected returns, covariance matrices, and risk weights), which are subject to estimation error. Monte Carlo robustness check results (Supplementary Appendix A) show that HV is reduced by 4.3% with a 20% perturbation of parameters, which is acceptable but not insignificant. Second, for large proportions of integer variables, using simple rounding to handle discrete-integer constraints may not produce optimal solutions; adding a dedicated mixed-integer evolutionary operator [19] is a natural evolution to this problem space. Finally, the use of the reference set for IGD calculation is an approximation generated from pooling all algorithmic outcomes; therefore, for problems where there is no analytical approximation of the true Pareto front, this creates uncertainty in the evaluation of + / -0.0003 IGD units.

7.4 Practical Deployment

Using the pymoo framework, MOEO-BO is implemented in Python 3.11 and features concurrent evaluation of objectives via concurrent. futures module. A typical execution of MOEO-BO (N=300, G=500, 80 decision variables) takes 4.7 minutes on a 16-core workstation. For an enterprise deployment, it is suggested to use a cloud-native architecture consisting of a scheduled weekly optimization batch job that feeds an interactive dashboard for executive decision-makers with tools for scenario analysis, allowing manual exploration of the Pareto front before strategy commitment. Since the algorithm has a modular approach to its objective functions, it is straightforward to integrate with ERP systems (SAP or Oracle) and risk management systems (Murex or OpenPages) through the use of REST APIs.

8. Conclusion

MOEO-BO, or the Multi-Objective Evolutionary Optimization algorithm, was created to maximize profit and minimize risk within a business's operational environment. MOEO-BO builds upon the framework of NSGA-III by combining six distinct objectives for business (net profit; conditional value-at-risk (CVaR) tail; operational risk; return on investment capital (ROIC); environmental/social/government (ESG) compliance; and market volatility) into a single algorithm through the use of an Augmented Lagrangian Constraint Repair Operator, with adaptive reference point perturbation and elite archiving also included in the design.

Across six industry sectors from 2018 to 2023, MOEO-BO has shown to outperform other traditional optimization techniques in terms of both Hypervolume index (0.847) and Inverse Growth Rate Distance (0.0028). A total of 65 real-world business units were analyzed and yielded aggregate profit improvements of 37.5% and reduced CVaR risk exposure by 39.4% compared to baseline operational strategies. Analysis conducted using ablation studies identified that the combination of using six objectives within the formulation, including diversity among reference points in the NSGA-III method for determining the final recommendation, as well as applying Augmented Lagrangian Repair to improve solution quality, provided significant disparities in the overall performance of both the approach and solutions when using these parameters.

Future research directions include: (1) developing integration between scenario-based robust optimization and hedging against parameter estimation uncertainty; (2) extending to large-scale portfolio management by creating a decomposed hierarchical evolutionary search methodology for 200+ SBU portfolios; (3) establishing online learning variations that allow for the updating of objective models based off of actual market data on an ongoing basis, and (4) developing explainability modules that generate natural language justifications for pareto optimal decisions that meet the requirements of the emerging eu ai act for high risk ai systems used in financial decision making contexts.

Acknowledgments

This research was supported by the Spanish Ministry of Science and Innovation (Grant PID2021-128108NB-C21), the European Research Council (ERC Consolidator Grant OptiCorpEA-891273), and the Colombian Ministry of Science, Technology, and Innovation (Minciencias Grant 80740-497-2022). The authors thank the seven corporate data partners for providing anonymized business-unit datasets under NDA agreements. Computational resources were provided by the Barcelona Supercomputing Center (MareNostrum5, Project BCV-2023-3-0014). The authors declare no competing financial interests.

References

1. Aghaei, J., Amjady, N., & Shayanfar, H. A. (2013). Multi-objective electricity market clearing considering demand response programs. *Energy Conversion and Management, 68*, 278–290. https://doi.org/10.1016/j.enconman.2012.12.019
2. Altiparmak, F., Gen, M., Lin, L., & Paksoy, T. (2006). A genetic algorithm approach for multi-objective optimization of supply chain networks. *Computers & Industrial Engineering, 51*(1), 196–215. https://doi.org/10.1016/j.cie.2006.04.011
3. Kasiyanto, Ariprinharta, & Sujito. (2025). Integrated QHBM-IoT framework for multi-objective optimization of emergency rescue operations in urban combat zones. *Journal of Internet Services and Information Security, 15*(4), 372–388. https://doi.org/10.58346/JISIS.2025.I4.027
4. Blank, J., & Deb, K. (2020). Pymoo: Multi-objective optimization in Python. *IEEE Access, 8*, 89497–89509. https://doi.org/10.1109/ACCESS.2020.2990567
5. Mokhtari, V., Mikaeilvand, N., Mirzaei, A., Nouri-Moghaddam, B., & Gudakahriz, S. J. (2025). GA-PSO-MIN: A hybrid heuristic algorithm for multi-objective job scheduling in cloud computing. *Archives for Technical Sciences, 2*(33), 22–46. https://doi.org/10.70102/afts.2025.1833.022
6. Das, I., & Dennis, J. E. (1998). Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems. *SIAM Journal on Optimization, 8*(3), 631–657. https://doi.org/10.1137/S1052623496307510
7. Uvarajan, K. P., Kavitha, M., Geetha, E., & Sadulla, S. (2025). Queue-aware multi-objective edge offloading and resource provisioning for dense IoT networks under synthetic workload dynamics. *International Academic Journal of Science and Engineering, 12*(1), 65–71. https://doi.org/10.71086/IAJSE/V12I1/IAJSE1251
8. Deb, K. (2001). *Multi-objective optimization using evolutionary algorithms*. Wiley-Interscience.
9. Virtanen, L., & Nieminen, J. (2026). Design and optimization of low-power high-performance VLSI architectures for next-generation SoC systems. *National Journal of Advanced VLSI Design and Systems, 1*(1), 46–51. Retrieved from https://iaeces.com/Index/index.php/NJAVDS/article/view/68
10. Fonseca, C. M., & Fleming, P. J. (1993). Genetic algorithms for multiobjective optimization: Formulation, discussion, and generalization. In *Proceedings of the Fifth International Conference on Genetic Algorithms* (pp. 416–423).
11. Karpagam, M. (2025). Machine learning-assisted design and performance optimization of smart antennas for beyond-5G networks. *National Journal of Antennas and Wireless Communication Systems, 1*(1), 28–35. Retrieved from http://www.antennawirelessjournal.com/Index/index.php/home/article/view/4
12. Bean, J. C. (1994). Genetic algorithms and random keys for sequencing and optimization. *ORSA Journal on Computing, 6*(2), 154–160. https://doi.org/10.1287/ijoc.6.2.154
13. Ganesan, Y., Haron, H., Amran, A., & Pitchay, A. A. (2018). Is it the external accountant matter in business performance of small and medium enterprises? The role of non-audit services. *International Academic Journal of Economics, 5*(1), 92–104. https://doi.org/10.9756/IAJE/V5I1/1810009
14. Datta, R., & Deb, K. (2016). A bi-objective-based hybrid evolutionary-classical algorithm for handling equality constraints in multi-objective optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference 2016* (pp. 441–448). https://doi.org/10.1145/2908812.2908877
15. Srinivas, N., & Deb, K. (1994). Multiobjective optimization using nondominated sorting in genetic algorithms. *Evolutionary Computation, 2*(3), 221–248. https://doi.org/10.1162/evco.1994.2.3.221

16. Zhang, Q., & Li, H. (2007). MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation, 11*(6), 712–731. https://doi.org/10.1109/TEVC.2007.892759
17. Farahani, R. Z., Rezapour, S., Drezner, T., & Fallah, S. (2010). Competitive supply chain network design: An overview of classifications, models, solution techniques, and applications. *Omega, 38*(1–2), 92–118. https://doi.org/10.1016/j.omega.2009.05.006
18. Gass, S. I., & Roy, P. G. (2003). The compromise hypersphere for multiobjective linear programming. *European Journal of Operational Research, 144*(3), 459–479. https://doi.org/10.1016/S0377-2217(02)00173-2
19. Charnes, A., & Cooper, W. W. (1961). *Management models and industrial applications of linear programming*. Wiley.
20. Sethi, K., & Kapoor, M. (2024). Data-Driven Marketing in the Age of AI: Reflections from the Periodic Series on Technology and Business Integration. In *Digital Marketing Innovations* (pp. 7-11). Periodic Series in Multidisciplinary Studies.
21. Anuradha K. Madugalla, & Hee-Seob Kim. (2025). A RISC-V Based Embedded Security Architecture for Trusted Execution in Industrial Control Systems. *SCCTS Journal of Embedded Systems Design and Applications* , *3*(1), 66-72.
22. Ishrat Z. Mukti. (2025). Energy-Efficient Control Algorithms for Grid-Integrated Electric Vehicle Charging Stations. *Transactions on Secure Communication Networks and Protocol Engineering*, *2*(1), 9-14.
23. M. Mejail. (2024). Next-Generation Digital Identity Systems for E-Governance Using Blockchain Technology. *Transactions on Internet Security, Cloud Services, and Distributed Applications*, *1*(1), 23–29.