

Soft optimization with generative heuristics: a framework integrating optimization, simulation, and large language models

Miguel Saiz^{1,*}, Laura Calvet², David Lopez-Lopez³ and Angel A. Juan^{4,5}

Abstract

Metaheuristics are widely used to solve combinatorial optimization problems often through the use of hybrid approaches such as simheuristics, which combine metaheuristics with simulation to address uncertainty. However, many real-world problems involve qualitative or semi-structured information that these methods do not handle well. To address this limitation, we introduce a soft optimization framework, which integrates quantitative search techniques with large language models to incorporate cognitive and interpretive capabilities. This framework, called generative heuristics, combines simheuristics with large language models to evaluate and generate candidate solutions based on both numerical performance and alignment with semi-structured goals. We apply this approach to a stochastic version of the project portfolio selection problem, where the objective is to maximize present value while also considering strategic alignment derived from textual project descriptions. Results show that the method can identify high-quality portfolios that balance hard constraints with soft ones and strategic criteria. The approach is general and can be extended to other stochastic combinatorial optimization problems involving mixed data inputs.

MSC: 68T20, 90-08, 90-10, 90B50.

Keywords: *Generative heuristics, large language models, simheuristics, decision-making, combinatorial optimization, project portfolio management, soft optimization.*

* Corresponding author: msaizgarcia@uoc.edu (Miguel Saiz)

¹ Computer Science Dept., Universitat Oberta de Catalunya, Barcelona, 08860, Spain.

² Telecommunications and Systems Engineering Dept., Universitat Autònoma de Barcelona, Sabadell, 08202, Spain.

³ General Management & Strategy Dept., ESADE Business School, Barcelona, 08034, Spain.

⁴ CIGIP-ValgrAI, Universitat Politècnica de València, Alcoy, 03801, Spain.

⁵ School of Engineering, UNIE Universidad, Madrid, 28029, Spain.

Received: December 2025

Accepted: July 2026

1. Introduction

Heuristics and metaheuristics stand as powerful methodologies extensively employed by both industry practitioners and academic researchers to tackle combinatorial optimization problems (COPs) in real-time (Amodeo, Talbi and Yalaoui, 2018). These problems span diverse domains including logistics and transportation (Reyes-Rubiano et al., 2019; Castaneda et al., 2022), finance (Doering et al., 2019), or bioinformatics (Calvet et al., 2023). With the inherent stochasticity in most real-world applications, there is an increasing interest in hybridizing metaheuristics and simulation, giving rise to what is termed as ‘simheuristics’ (Juan et al., 2025). This hybrid approach is gaining prominence as a potent tool for addressing stochastic COPs effectively. While the metaheuristic component focuses on creating and improving solutions, the simulation element assesses the performance of the most promising solutions under uncertainty scenarios. Consequently, simheuristics find extensive application in resolving problems characterized by stochastic quantitative parameters present in the objective functions, constraints, or both.

Nevertheless, many decision-making processes involve elements that cannot be captured purely through structured data or numerical models. These include qualitative or semi-structured sources such as textual reports, strategic plans, or stakeholder preferences. Addressing such problems requires an approach that integrates both quantitative and qualitative components. We refer to this broader class of problems as requiring ‘soft optimization’, where both hard numerical constraints and soft strategic or semantic objectives are considered. In project management, for example, selecting a portfolio of projects involves not only optimizing financial metrics but also ensuring alignment with strategic goals described in unstructured or semi-structured formats. Similar needs appear in finance, where investment decisions are guided by analysis of textual data from company documents, and in supply chain design, where supplier selection may involve qualitative assessments.

Large language models (Min et al., 2023), such as OpenAI’s generative pre-trained transformer (GPT) series (<https://chat.openai.com>), offer new capabilities to handle such unstructured information. These models are based on deep learning architectures and trained on large textual corpora, enabling them to understand and generate human-like responses. Large language models (LLMs) have shown strong performance in tasks involving language understanding, summarization, and decision support. At the same time, their use introduces challenges related to ethical considerations (Kapania et al., 2025), bias (Simpson et al., 2025), and societal impacts (Coeckelbergh, 2025). In this paper, we present a novel framework called ‘generative heuristics’, which addresses soft optimization by combining simheuristics with LLMs. The main contribution of this work is the integration of structured optimization techniques with LLM-based evaluation to tackle decision problems that involve both quantitative and qualitative information. Our framework extends simheuristics to include an LLM component, which enables the evaluation and generation of solutions based on non-numeric criteria, such as alignment with strategic objectives. We show the application of this framework to a stochastic variant of the project

portfolio selection problem (PPSP) (Mohagheghi, Meysam Mousavi and Mojtahedi, 2020). In this setting, projects have uncertain durations, costs, and returns. The goal is to maximize the expected present value of the selected portfolio while also achieving good alignment with a predefined strategic plan. The proposed methodology uses a genetic algorithm with Monte Carlo simulation (MCS) (Katoch, Chauhan and Kumar, 2021) to address the stochastic aspects, and a GPT-4.1 model to evaluate strategic alignment from project descriptions and plans. A computational experiment was conducted using a dataset from Saiz et al. (2026), which includes twenty project proposals and related structured data such as schedules, risk registers, and cash flows. The strategic plan contains ten environmental drivers used to guide the LLM evaluations. The results show that the method can generate a set of non-dominated portfolios that balance present value (PV) and strategic alignment. The outputs were verified using both the GPT-4.1 Mini model and the MAI-DS-R1 research model developed by Microsoft, confirming the robustness of the approach.

Overall, this work contributes a generalizable method for soft optimization, applicable to other domains that combine structured optimization with qualitative assessments. The combination of simheuristics and LLMs in a unified loop supports more nuanced decision-making in complex, uncertain environments. The remainder of the document is organized as follows. Section 2 reviews the related literature on hybrid metaheuristics, the use of LLMs in optimization, and project portfolio selection problems. Section 3 formally introduces the project portfolio selection problem addressed in this work. Section 4 presents the proposed generative heuristics framework. Section 5 describes the methodology used to solve the problem. Next, Section 6 reports the computational experiments, while Section 7 discusses the obtained results. Finally, Section 8 concludes the paper and outlines future research directions.

2. Literature review

This literature review is structured into three parts. First, we review hybrid metaheuristics that combine simulation and optimization under uncertainty. Second, we examine how LLMs are used in optimization beyond text generation. Third, we explore the modeling of the PPSP, including its evolution toward multi-objective formulations and the emerging role of LLMs in evaluating strategic criteria.

2.1. Hybrid metaheuristics

Hybrid metaheuristics combine different search strategies to overcome limitations of standalone algorithms, such as local optima and premature convergence. These methods have become increasingly relevant for solving complex, real-world problems. Talbi (2021) discusses the integration of machine learning (ML) with metaheuristics, offering a taxonomy based on where ML is applied—problem modeling, low-level operations, or high-level strategies. The paper highlights the need for unified frameworks in ML-augmented optimization. Rezk and Selim (2024) focus on metaheuristics for ensemble

pruning in machine learning, identifying research gaps in hybrid models for multiclass classification and hyperparameter tuning. Azevedo, Rocha and Pereira (2024) review hybrid optimization-ML methods for clustering and classification, showing that combining global search with pattern learning improves performance in noisy, high-dimensional settings. Boschetti and Maniezzo (2024) highlight matheuristics, which integrate mathematical programming with heuristics, and discuss emerging directions such as AI-based and quantum hybrids.

Simheuristics are designed to handle uncertainty by integrating simulation with metaheuristic search (Chica et al., 2020). These methods are especially useful in logistics and energy systems, where they assess solution feasibility and robustness. Dokeroglu, Kucukyilmaz and Talbi (2024) discuss hyper-heuristics that dynamically manage low-level heuristics and are suitable for multi-objective and ML-based problems. Sangaiah et al. (2023) propose a hybrid system combining ant colony optimization and fuzzy logic for dynamic resource allocation in cloud-edge environments, showing improved performance over conventional methods. Recent trends in hybrid metaheuristics emphasize adaptability (Martin et al., 2024), deep learning integration (Akay, Karaboga and Akay, 2022), and real-time applications (Adelantado et al., 2022). These developments support more intelligent and flexible decision-making in dynamic environments.

2.2. Use of LLMs in optimization

LLMs are increasingly integrated into optimization workflows, extending the idea of learnheuristics, hybrid methods that combine metaheuristics with learning. Calvet et al. (2017) introduced this term for heuristics that adapt through experience. Later works, such as Q-learnheuristics (Crawford et al., 2021) and simulation-based variants (Bayliss, 2021), emphasize learning-driven exploration. In this context, LLMs act as semantic optimizers that support goal alignment and interpretability. Recent studies explore LLMs in optimization roles beyond text generation. Tarczewski, Stojic and Dzielinski (2025) and Mustata et al. (2025) show how LLMs can interpret qualitative requirements and support multi-criteria decisions. Applications include UAV planning (Tian et al., 2025), smart cities (Kalyuzhnaya et al., 2025), and transport prediction (de Zarzà et al., 2023). LLMs are also used as adaptive agents (Cinkusz, Chudziak and Niewiadomska-Szynkiewicz, 2025) that handle ambiguity and align solutions with high-level goals.

Prompt engineering plays a key role in controlling LLM behavior. Chain-of-thought (CoT) prompting encourages reasoning through intermediate steps. Goli and Singh (2024), Li et al. (2024), and Hebenstreit et al. (2024) report improved accuracy and coherence using CoT. This is echoed in diagnostics (Kosten, Nooralahzadeh and Stockinger, 2025), scientific reasoning (Lucas, Smith and Johnson, 2024), and education (Parker et al., 2025). CoT also boosts robustness under zero-shot settings (Sun, Zhang and Chen, 2024; Allen, Polat and Groth, 2024) and improves output quality in engineering (Chen et al., 2025) and energy (Chen et al., 2025) contexts. Alsofyani and Wang (2025) stress the role of prompt design in ensuring stability. Different prompting strategies (zero-shot, few-shot, and

instruction-tuned) affect LLM performance. Zero-shot uses the model’s pretraining alone, few-shot adds examples, and instruction-tuned prompts guide behavior with task-specific inputs. Chen and Li (2025) and Alsofyani and Wang (2025) show instruction-tuning improves output consistency. Guo et al. (2025) use meta-instruction prompting for SQL generation, while Zhang et al. (2024) apply CoT prompting in text-to-3D terrain mapping. These strategies enable LLMs to handle optimization tasks with qualitative inputs and ambiguous goals.

2.3. Modeling the PPSP

Initial PPSP models aimed to maximize present value or minimize costs under constraints (Saiz et al., 2022). As complexity increased, multi-objective models emerged to address strategic and qualitative factors. Mohagheghi, Mousavi and Vahdani (2017) and Calvo, Ivorra and Liern (2017) use fuzzy logic to combine profitability with strategic alignment. Ranjbar, Nasiri and Torabi (2022) model cost and time jointly, accounting for uncertainty and dependencies. These works shift focus from financial criteria toward broader stakeholder goals. Iranmanesh, Shakhshi-Niaei and Rastegar (2019) introduce complexity metrics, while Azevedo et al. (2024) and Cesarone, Martino and Carleo (2022) incorporate environmental, social, and governance criteria into the objective functions. Robustness and flexibility are now key requirements. Ranjbar et al. (2022) apply scenario-based methods, and Saiz et al. (2024) use simheuristics to manage uncertainty. These trends highlight a transition to more realistic and value-aligned portfolio models.

As PPSP models expand to include intangible factors, LLMs are gaining relevance in strategic evaluation. Zhuang and Wu (2025) show how LLMs balance performance and governance in corporate investments. Kim (2023) use ChatGPT for asset management, weighing returns against ethical and volatility constraints. Aiche, Winer and Cohen (2024) and Coelho e Silva, Fonseca and Castro (2024) apply LLMs in portfolio tasks involving cybersecurity and finance. Gopal (2024) show that transformers can capture non-obvious factor structures in portfolios. In public project evaluation, Saiz and Calvet (2025) propose a hybrid of LLMs and simheuristics for assessing alignment with high-level goals. Related studies (Sun et al., 2024; Tepelyan and Gopal, 2023) use generative and graph-based models to capture latent value and risk. These approaches show that LLMs can produce semantic alignment scores and explain portfolio quality using qualitative data, supporting their role in strategic decision-making.

3. Project portfolio selection problem

The stochastic PPSP seeks to maximize the expected present value of a project portfolio:

$$\max E[PV] = \max \sum_{i=1}^n x_i \cdot \sum_{t=1}^T \frac{R_{i,t}}{(1+W)^t}$$

Here, $R_{i,t}$ represents the reward generated by project i at time t , modeled as a random variable due to uncertainty. The discount factor W denotes the weighted average cost of capital (WACC), also treated as a random variable to account for financial variability. The binary decision variable $x_i \in \{0, 1\}$ indicates whether project i is selected ($x_i = 1$) or not ($x_i = 0$). The portfolio must satisfy a probabilistic budget constraint:

$$p \left(\sum_{i=1}^n x_i \cdot C_i \leq C_{\max} \right) \geq F_{\min}$$

This constraint ensures that the probability of the total cost of selected projects (each with a stochastic cost C_i) not exceeding the budget limit $C_{\max}$, is at least $F_{\min}$. The threshold $F_{\min}$ reflects the decision-makers' minimum acceptable confidence level, typically based on organizational or industry-specific risk tolerance. Each project cost C_i is defined as:

$$C_i = b_i + L_i \cdot f$$

where b_i is the base cost, f is a cost factor, and L_i is the project duration. The duration L_i is estimated using the critical path method (CPM) (Kelley Jr and Walker, 1959), incorporating stochasticity in the activity durations D_{ij} , which are modeled using triangular distributions.

Building upon this formulation of the stochastic PPSP, we introduce a secondary objective: maximizing the strategic alignment of the selected projects with the funding organization's strategic plan. This alignment is quantified using a strategic score S , ranging from 1 to 100. A solution to this extended problem includes the composition of the selected project portfolio, along with key performance indicators such as expected PV, confidence level, and strategic alignment. Additionally, the solution must provide a justification for each project selected, as well as an executive summary. Figure 1 illustrates a simplified representation of the stochastic PPSP.

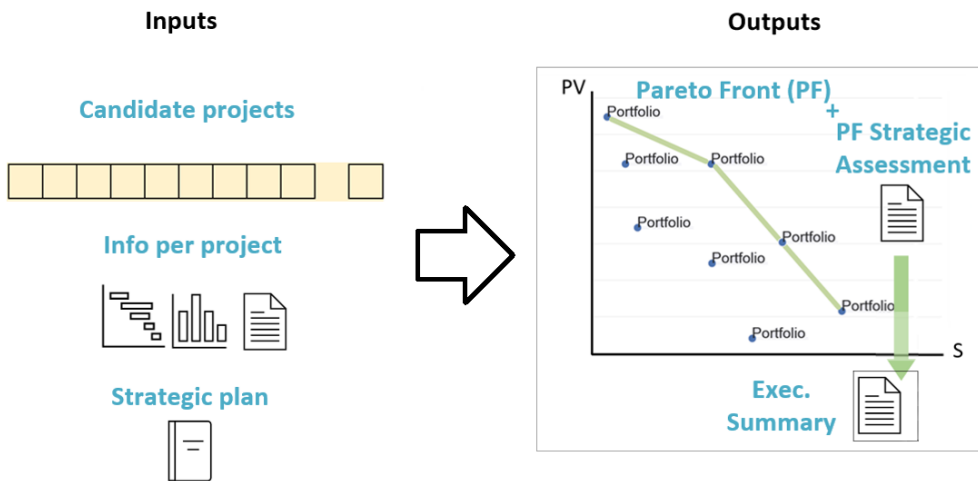


Figure 1. Representation of the stochastic project portfolio selection problem.

4. Generative heuristics

Combinatorial optimization problems involve finding the best solution from a discrete set of options under given constraints. These problems are common in areas such as logistics, scheduling, portfolio selection, and network design. Hybrid simulation-optimization (sim-opt) methods, introduced in the 1950s, have advanced significantly in recent decades due to improvements in computational resources and optimization techniques. In this context, the term ‘simheuristics’ was introduced to describe approaches that combine metaheuristics with simulation for solving stochastic COPs efficiently. Natural language processing (NLP) also began in the 1950s, initially using rule-based and statistical models. In the 1980s and 1990s, hidden Markov models and n-gram models were widely used. Later, the development of neural networks, especially after the introduction of backpropagation, led to major advances. Bengio, Ducharme and Vincent (2000) proposed feed-forward neural networks for language modeling, laying the foundation for deep learning in NLP. The emergence of large-scale models like GPT (starting in 2018) further transformed the field, with successive versions (GPT-2 in 2019, GPT-3 in 2020, GPT-4 in 2023) employing massive datasets and deep architectures. The convergence of progress in simheuristics and language models now allows for solving complex problems that involve both quantitative and qualitative data.

4.1. Framework description

The generative heuristics framework combines simheuristics with LLMs to address stochastic COPs involving semi-structured information, such as strategic plans or project reports. The primary objective is to identify robust solutions that simultaneously balance quantitative metrics (e.g., cost or return) with qualitative dimensions (e.g., strategic alignment). This dual optimization yields a Pareto set of non-dominated solutions, where any improvement in a quantitative objective necessitates a compromise in qualitative alignment, or vice versa. A granular breakdown of the framework is detailed in Algorithm 1.

The framework takes as input an instance of the stochastic COP. The workflow begins with a stochastic preprocessing phase that employs MCS to model and sample uncertain variables, thereby generating empirical distributions for the optimization process (Line 1). Subsequently, these empirical distributions are mapped into deterministic estimates, using expected or conservative values under a specified confidence level, to establish a deterministic proxy of the COP (Line 2). The LLM-driven module functions as a semantic evaluator, generating preliminary strategic assessments from qualitative descriptors (Line 3). These evaluations are cached and systematically reused, ensuring complete semantic consistency across solution evaluations while drastically minimizing computational overhead. Once this baseline is established, the framework enters its core hybrid, iterative optimization loop, which runs continuously until a stability-based stopping criterion is met (Lines 4-17). Within this loop, the search progresses through three distinct phases:

Algorithm 1 Proposed Integrated Framework Workflow**Require:** An instance of the stochastic COP**Ensure:** A Pareto set of robust, non-dominated solutions with risk/reliability metrics and narrative justifications

```

// Quantitative Models & Preliminary Assessment
1: Preprocess stochastic COP
2: Convert into deterministic COP
3: Generate preliminary evaluation of each individual option
4: while Stop condition reached? is N do
    // Simheuristic-driven search
5:   Generate a new solution using a simheuristic-driven search
6:   if promising solution? is Y then
7:     Store in set of promising solutions ( $Q_{HoF}$ )
8:   end if
9:   if  $Q_{HoF}$  big enough? is Y then
10:    Generate reduced solution space
11:   end if
    // LLM-driven search
12:   Generate aspirational solutions
13:   Generate set of most promising solutions
    // Cross-check
14:   Estimate quality / feasibility LLM generated candidates
15:   Estimate quality / feasibility of  $Q_{HoF}$ 
    // Pareto Update
16:   Update Pareto Frontier
17: end while
// Final Reporting
18: Perform risk / reliability analysis and report with justification

```

- **Simheuristic-driven Search (Lines 5-11):** The metaheuristic explores the deterministic solution space to identify feasible candidates that satisfy the constraints (Line 5). High-performing, non-dominated solutions are stored in a quantitative hall of fame (Q_{HoF} , Line 7). Once this archive reaches a target size, the framework analyzes these solutions to isolate frequent attributes and stratify the search space, generating a reduced solution space that focuses subsequent analysis (Line 10).
- **LLM-driven Search (Lines 12-13):** The LLM-driven search phase employs qualitative assessment to propose top ‘aspirational solutions’ (Line 12). These represent idealized configurations generated strictly from a qualitative perspective, even if their quantitative feasibility is initially ignored. Guided by these benchmarks, the LLM explores structural variants of these aspirational solutions, evaluating and ranking them to generate a refined set of the most promising candidates (Line 13).
- **Cross-Evaluation and Refinement (Lines 14-16):** To bridge both dimensions, the framework executes an automated cross-check mechanism. The framework quantitatively estimates the feasibility of the purely LLM-generated candidates (Line 14), while the LLM qualitatively scores the strategic value of the Q_{HoF} solutions (Line 15). High-quality, non-dominated solutions from both trajectories are filtered

through Pareto dominance to continuously update the running empirical Pareto frontier (Line 16).

Upon satisfying the stopping condition, the algorithm returns the final consolidated set of strategically strong, feasible candidates. As a final post-optimization step, the framework executes a detailed solution risk and reliability analysis on this set, utilizing the LLM to generate narrative justifications for the final recommendations.

As a simheuristic, this method operates under the proxy assumption that under moderate uncertainty, solutions performing well in the deterministic problem tend to maintain robust properties in the stochastic setting. Although not always exact, this assumption is highly effective in practice and significantly minimizes search effort. Under high uncertainty, this relationship may break down; however, the hybrid nature of the framework provides the necessary adaptability to mitigate this limitation. Consequently, this assumption enables the efficient identification of strong candidates for the stochastic COP.

5. Solving methodology

This section adapts the generative heuristics framework introduced in previous section to solve the stochastic PPSP. The execution flow begins with a stochastic preprocessing phase using MCS (Subsection 5.1) to model the cost and present value distributions for each individual project. These empirical distributions are subsequently mapped into deterministic point estimates. Following this, a multi-start genetic algorithm (GA) procedure is employed to maximize expected PV by evolving feasible portfolios, where hard budget and risk confidence thresholds enforce structural constraints (Subsection 5.2). This multi-start strategy is aimed to enhance exploratory diversity and mitigate premature convergence (Homayouni, Fontes and Gonçalves, 2023; López-Soto et al., 2017; Mao et al., 2021). Concurrently, an LLM manages the qualitative evaluation layer (Subsection 5.3), initiating with a preliminary strategic assessment of each individual project. Based on these evaluations, the LLM constructs an optimal portfolio for each target cardinality and expands the search space by proposing additional candidate solutions through a strategic neighborhood search. Every generated candidate is evaluated for feasibility and Pareto dominance, an iterative cycle that continues until either 10 feasible portfolios are discovered or 10 independent LLM iterations are completed.

In the final phase (Subsection 5.4), the GA-derived (PV-optimized) and LLM-derived (strategically optimized) portfolios undergo an automated cross-evaluation to construct the final Pareto frontier. Upon satisfying the terminal stopping criterion, a post-optimization verification step utilizes the LLM to independently reassess strategic alignment directly from the baseline data, ensuring the final recommendations remain robust. As with other metaheuristic approaches, the proposed method does not guarantee convergence to the global optimum or the absolute true Pareto front. Instead, the objective is to generate a high-quality approximation of the trade-off frontier within a limited computational budget. The resulting Pareto front should therefore be interpreted as a

practical decision-support frontier rather than a certificate of global optimality. To address the well-known risks of premature convergence and algorithmic stagnation, the algorithm implementation incorporates several explicit safeguards. These include a multi-start strategy, randomized crossover and mutation probabilities across independent runs, gene-level mutation operators, and a hall of fame (HoF) archive that preserves elite solutions across iterations. Collectively, these mechanisms maintain structural diversity throughout the search process and significantly reduce dependency on a single evolutionary trajectory.

Figure 2 illustrates the bi-objective PPSP. The process initiates with a simheuristic search maximizing expected PV to generate an elite HoF, which defines the reduced cardinality set (C_{red}). For each cardinality, the LLM identifies an aspirational portfolio based on its strategic text score. These aspirational configurations serve as reference anchors to evaluate all PV HoF solutions (red dots). Simultaneously, a strategic neighborhood is generated by expanding around these anchors to locate similarly aligned candidates; those meeting quantitative feasibility limits are retained (green dots), while infeasible ones are discarded. Consequently, this dual mechanism isolates two finalist trajectories: a simheuristic-led, LLM-assisted path (red dotted line) and an LLM-led, simheuristic-assisted path (green dotted line), intersecting to define the final empirical Pareto frontier (central black dot). The following subsections provide further details on the main steps of our methodology.

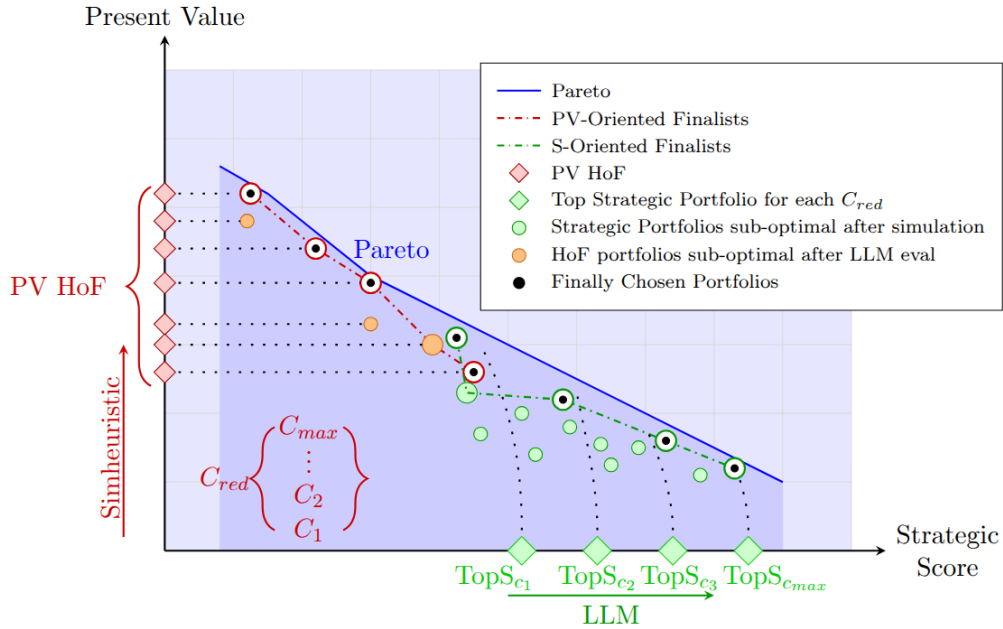


Figure 2. Graphical bi-objective representation of the specific stochastic PPSP procedure.

5.1. Stochastic preprocessing via Monte Carlo simulation

The framework initiates with a project-level stochastic preprocessing phase, formalized in Algorithm 2. For each project $i \in \{1, \dots, n\}$, independent simulation runs are executed over I iterations to derive empirical distributions and representative point estimates for both cost and expected PV. Specifically, the function `triangularSim` simulates project costs using triangular distributions derived from the baseline project cost, task schedules, and respective risk registers. Subsequently, the `computePV` function discounts the simulated cash flows at a designated weighted average cost of capital to construct the empirical distribution `PVDist`. Finally, the `pointestimate` function maps these empirical curves into deterministic values (`CostEstimate` and `PVEstimate`) based on a specified confidence level (CL). All generated distributions and point estimates are consolidated into project-level arrays, which are returned by the algorithm to feed the subsequent *simheuristic* search engine.

Algorithm 2 Project-Level Monte Carlo Simulation

Inputs:

n = number of projects
 for each project: base cost, schedule, risk register, cash flows
 I = number of simulation iterations
 CL = confidence level
 $wacc$ = weighted average cost of capital

Outputs:

`CostDist`, `PVDist`, `CostEstimate`, `PVEstimate`

```

1: for  $i = 1$  to  $n$  do
2:   CostSim  $\leftarrow$  triangularSim(baseCost, schedule, risk,  $I$ )
3:   PVDist  $\leftarrow$  computePV(cashFlows,  $wacc$ ,  $I$ )
4:   CostEstimate  $\leftarrow$  pointestimate(CostSim,  $CL$ )
5:   PVEstimate  $\leftarrow$  pointestimate(PVDist,  $CL$ )
6:   Append all results to project-level arrays
7: end for
8: return all distributions and point estimates

```

5.2. Simheuristic search

The optimization process is formalized in the multi-start GA procedure detailed in Algorithms 3 and 4. Implemented via the DEAP library (Fortin et al., 2012) with portfolios encoded as binary vectors, the multi-start routine (Algorithm 3) executes k independent runs to maximize solution-space coverage. In each run r , the algorithm randomizes operators and invokes the cardinality-targeted search engine (Algorithm 4, Line 4) to evaluate candidate configurations. This internal engine bounds the search domain by evaluating expanding sizes $(m + j)$ from the largest feasible cardinality m until resource constraints render solutions infeasible, explicitly defining the reduced space C_{red} . The elite solutions discovered across these targeted cardinality runs are stored, allowing

the algorithm to ultimately return a consolidated HoF (selectBest) containing the top h feasible, PV-optimized portfolios.

Algorithm 3 Multi-Start GA for expected PV Maximization

Inputs: distributions, budget B , confidence α , GA settings

Output: Consolidated HoF (top h individuals)

```

1: AllHoF  $\leftarrow []$ 
2: for  $r = 1$  to  $k$  do
3:   Randomize crossover/mutation settings
4:   Run GA; store HoF $r$ 
5:   Append results to AllHoF
6: end for
7: return selectBest(AllHoF,  $h$ )

```

▷ Algorithm 4

Algorithm 4 Cardinality-Targeted GA Subroutine

Inputs: same as Algorithm 3, target cardinality C

Output: Feasible portfolios and set C_{red}

```

1:  $j \leftarrow 1$ 
2: while true do
3:    $c \leftarrow m + j$ 
4:   Run GA with constraint  $c$ 
5:   if no feasible portfolios then
6:     break
7:   else
8:     Store feasible solutions
9:   end if
10:   $j \leftarrow j + 1$ 
11: end while

```

5.3. LLM search

The qualitative exploration phase utilizes an LLM model to integrate strategic alignment into the optimization loop. Initially, the LLM evaluates each project against the organization’s strategic plan, outputting a descriptive justification and a baseline textual score to anchor subsequent evaluations. Following this preliminary assessment, the LLM generates a single top-performing portfolio for each target cardinality defined within the reduced search space C_{red} . Each configuration is assigned a strategic score $S \in [1, 100]$ alongside a narrative justification, storing these selections as aspirational solutions. Using these aspirational portfolios as structural anchors, the framework initiates a generative local search. For each target size $c \in C_{red}$, the LLM is queried to propose 20 portfolio variations (Algorithm 5). Each generated candidate is subsequently mapped to the quantitative data layer to evaluate its expected PV and feasibility metrics. A local Pareto filter is then applied to retain exclusively non-dominated solutions. This interactive generation loop repeats for each cardinality target until either a minimum of 10 feasible portfolios are secured or a maximum threshold of 10 LLM queries is reached.

Algorithm 5 Strategic Neighborhood Exploration

Inputs: C , distributions, thresholds
Output: Pareto-filtered candidates

```

1: for each  $c \in C$  do
2:    $Feasible \leftarrow []$ 
3:   repeat
4:     Ask LLM for 20 portfolios of size  $c$ 
5:     Parse vectors and scores
6:     Evaluate feasibility
7:     Append feasible ones
8:   until enough found or 10 loops
9:   Apply Pareto filter
10:  Store results
11: end for

```

5.4. Pareto frontier, stop condition, and verification

In the ending phase, the GA-derived (PV-optimized) and LLM-derived (strategically optimized) portfolios are consolidated and re-evaluated by the LLM to construct the final Pareto frontier. Upon satisfying the stopping criterion, a post-optimization verification step utilizes the LLM model, operating exclusively on the raw strategic plan and project data, to independently reassess strategic alignment. Each candidate receives a final consensus score and textual justification. These results are structured within the ‘Finalists Table’ to support the comparison and review of the final recommendations.

6. Computational experiments

The implementation of the proposed approach was developed in Python, using specialized libraries including NumPy for numerical computations, DEAP for evolutionary algorithms, Matplotlib for visualization, and the Azure OpenAI API for LLM integration. All experiments were executed on a standard personal computer equipped with an Intel Core i7 CPU at 2.9GHz, 16GB of RAM, and Windows 10. The algorithm and supporting code are publicly available in a GitHub repository (https://github.com/weOptimize/MS5_genHeuristics_insideOptimizer) for reproducibility and further research.

6.1. Dataset

As the reference dataset for structured project information, we used the benchmark data generated by Saiz et al. (2026), which includes twenty realistic projects designed to support research in project portfolio optimization. Each project is defined by: a work breakdown structure (WBS) with 9 to 13 activities; a detailed schedule network with precedence relationships; triangular estimates of activity durations; a budget baseline from point estimates; a risk register with ten identified risks per project; risk occurrence

probabilities; triangular distributions for potential risk impacts; and expected yearly cash flows over a four-year horizon. This dataset has been enriched with a textual summary for each project (one or two sentences) describing its objectives and expected benefits, synthetically generated using Microsoft Copilot. Similarly, the strategic plan used to evaluate project alignment was also synthetically generated with Microsoft Copilot. It consists of ten strategic drivers related to environmental targets.

The utilization of a single benchmark dataset containing synthetically generated textual descriptions must be interpreted in light of the core objectives of this study. Rather than presenting an exhaustive empirical benchmark of competing semantic scoring mechanisms, the primary purpose of this computational experiment is to provide a highly controlled, fully reproducible proof of concept. This design illustrates how structured stochastic optimization data can be systematically paired with semi-structured strategic information inside the generative heuristics loop without the confounding noise typical of uncurated corporate documents.

6.2. Simulation parameters

The simulation parameters were selected to balance estimation stability with computational efficiency. Utilizing 2,000 simulation runs provides robust empirical distributions for project cost and PV while ensuring the experiment remains executable on standard computing hardware. The 75% confidence level applied during the project-level pre-processing stage represents a moderately conservative baseline for individual project planning. Conversely, the 90% portfolio confidence requirement imposes a stricter feasibility condition at the macro level. This asymmetric distinction reflects realistic corporate governance, where moderate uncertainties are tolerated at the individual project level, but the aggregated portfolio must satisfy a higher reliability threshold. Accordingly, the portfolio budget constraint was set at 9,000 monetary units (as a soft constraint), paired with a minimum portfolio confidence level of 90% (as a hard constraint). Thus, feasible portfolios must remain within budget with at least a 90% probability. PV calculations incorporated a WACC of 10%. While our underlying framework natively supports the stochastic modeling of cash flows and WACC, these parameters were treated deterministically in this validation study to cleanly isolate the effects of project selection under uncertainty.

6.3. GA parameters

The GA parameters were selected to establish a transparent, lightweight baseline for framework validation rather than to maximize performance through instance-specific tuning. A population size of 20 and a limit of 50 generations allow the search to explore diverse portfolio combinations while preserving rapid execution times. The tournament size of 3 and HoF size of 6 maintain a compact, high-quality elite set for subsequent LLM-based evaluation. To enhance exploration and reduce dependency on

fixed parameter choices, the implementation uses a multi-start strategy where crossover and mutation probabilities are randomized across independent restarts. This multi-start process is stability-driven: the global search terminates only when the elite HoF set remains unchanged for a predefined number of consecutive iterations. This design mitigates the risk of premature convergence and reduces reliance on a single evolutionary trajectory without requiring an expensive calibration phase. While systematic parameter tuning could yield further performance gains in a production deployment, it remains orthogonal to the primary contribution of this work, which focuses on the architectural integration of simulation, heuristic search, and LLM-driven semantic assessment.

6.4. LLM integration

The strategic assessment used Azure-hosted GPT-4.1 Mini with the following configuration: temperature = 0.1 (to ensure consistent evaluations), top- p = 0.90, and a maximum token count of 3,000 units. The LLM received as input the strategic plan (with numbered drivers) and summaries of the 20 competing projects. The assessment was conducted in multiple phases: (i) individual project evaluation; (ii) portfolio-level assessment per cardinality; (iii) strategic neighborhood exploration with 20 portfolios per iteration; (iv) consolidated evaluation of finalist portfolios; and (v) independent verification of the Pareto frontier. To extract structured data from natural language outputs, we implemented pattern recognition algorithms with a focus on parsing tables that contain strategic alignment scores.

To ensure the generative heuristic framework yields consistent and strategically aligned results, the LLM interactions follow a rigid, zero-shot structured prompting architecture. The prompting mechanism is bifurcated into two specialized operational roles: (i) a project evaluator role for individual candidate appraisal; and (ii) a portfolio architect role for macro-level neighborhood generation and scoring. To guarantee deterministic outputs and prevent formatting drift, all prompts enforce strict schemas for structural parsing, requiring explicit separation between quantitative metrics and qualitative justifications. The precise system instructions, role contextualizations, and representative prompt templates deployed across these evaluation layers can be found in the code repository, as well as summarized in subsection 6.5. While LLM-based evaluations exhibit inherent stochasticity across varying prompt structures, model versions, or repeated executions, the framework developed in this study treats the strategic alignment score as deterministic conditional on a rigid, highly constrained evaluation protocol. This protocol minimizes semantic evaluation noise by enforcing a fixed prompt template, a bounded scoring scale, and tightly controlled decoding parameters (specifically, a low temperature value of 0.1 and a top- p of 0.90) designed to maximize output consistency.

A known limitation of LLM-based evaluation is the possibility of hallucinated, unsupported, or inconsistent assessments. In the proposed framework, this risk is mitigated by using the LLM as a constrained evaluator rather than as an open-ended generator. The model is provided with the strategic drivers and project summaries, and it is asked to

assign bounded strategic-alignment scores using only the supplied information. Thus, the LLM is not expected to create new factual project attributes or infer missing data.

Table 1. *Summary of the LLM prompts used in the workflow.*

Step	Prompt	Main input	Main purpose
1	Initial project assessment	Strategic plan and all project descriptions	Identify the strategic drivers addressed by each project and describe the degree of alignment as very strong, strong, average, or weak. No numerical project score is requested.
2	Portfolio proposal by cardinality	Initial project assessment and required number of projects	Propose the most strategically aligned portfolio containing exactly the required number of projects.
3	Scoring of cardinality proposals	Initial project assessment and portfolios proposed in Step 2	Summarize the proposed portfolios and assign each one a strategic score from 1 to 100.
4	Scoring of GA portfolios	Initial project assessment, previous portfolio proposals, and GA HoF portfolios	Assign strategic scores from 1 to 100 to the portfolios produced by the genetic algorithm.
5	Strategic-neighborhood generation	Initial project assessment and required portfolio cardinality	Generate twenty strategically promising portfolios of the required size and assign a strategic score to each one.
6	Finalist scoring	Initial project assessment and consolidated finalist portfolios	Assess the strategic alignment of all finalists and produce a table with portfolio identifiers, compositions, and scores.
7	Final verification	Strategic plan, all project descriptions, and Pareto-frontier portfolios	Reassess each Pareto-frontier portfolio from 1 to 100 and provide a short justification based on the number and strength of the strategic drivers covered.

6.5. Prompting strategy

The LLM workflow uses seven prompt types, summarized in Table 1. The first prompt assesses the strategic alignment of all individual projects in qualitative terms. It identifies the strategic drivers addressed by each project and describes the alignment as very strong, strong, average, or weak. It does not assign numerical scores to individual projects. The resulting assessment is reused as context in the next stages. These stages propose portfolios for different cardinalities, score portfolios from 1 to 100, generate strategically

promising portfolios, and assess the portfolios produced by the optimization process. A final prompt checks the portfolios on the Pareto frontier using the strategic plan and all project descriptions again.

The prompts have three roles. Prompt 1 creates a qualitative assessment of the individual projects. Prompts 2 and 5 use this assessment to propose strategically promising portfolios. Prompts 3, 4, 6, and 7 assign or verify numerical portfolio scores. Keeping project assessment, portfolio generation, and portfolio scoring as separate tasks makes the role of each LLM call explicit. The LLM does not assign a numerical score to each individual project. Instead, the qualitative project assessment provides shared context for the later portfolio-level evaluations. Portfolio scores therefore reflect the combined coverage and strength of the strategic drivers rather than a simple sum of individual LLM project scores. The complete prompt templates are available in the public repository.

6.6. Experimental procedure and validation

The experimental evaluation followed the methodology and computational configurations established in the previous sections. To evaluate efficiency, execution times were benchmarked for each algorithmic stage. The end-to-end optimization process typically converged in under 6 minutes, with LLM API interactions accounting for approximately 50% of the total runtime. Beyond the integrated self-verification step designed to facilitate practical deployment, we performed an external cross-validation using a deep research LLM, specifically MAI-DS-R1, a Microsoft post-trained variant of DeepSeek-R1 fine-tuned on domain-specific corpora for complex dataset analysis. MAI-DS-R1 was deployed to independently assess the final Pareto-frontier portfolios alongside GPT-4.1. Crucially, both models preserved the identical ordinal ranking of the portfolios, indicating that the core qualitative ordering remains highly robust even when absolute strategic scores vary between architectures. Nevertheless, these LLM-generated scores must be interpreted strictly as decision-support indicators rather than absolute ground truth: in high-stakes managerial contexts, the resulting portfolios and text justifications should undergo human-in-the-loop validation by domain experts prior to implementation. The complete and unedited verification output from the deep research model is hosted in the public repository (https://github.com/weOptimize/MS5_genHeuristics.insideOptimizer/blob/main/deep_research_model_validat.txt) under the header ‘Validation response from LLM’.

Additionally, to assess the impact of evaluation noise on the ranking among the four reported Pareto-frontier portfolios, we conducted an internal robustness experiment across multiple runs (10 executions per model) using a suite of different architectures (GPT-4.1, GPT-4.1-mini, and GPT-5.4-mini). This experiment evaluates the stability of the ranking among these four portfolios, rather than the stability of the complete Pareto frontier. The results (see Figure 3) confirm the presence of minor score variability, e.g., occasional permutations between adjacent close ranks like 11 – 10 versus 10 – 11. However, the macro-qualitative ordering remains generally stable across runs, with the

dominant ranking (7 – 16 – 11 – 10) emerging as the most frequent ranking.

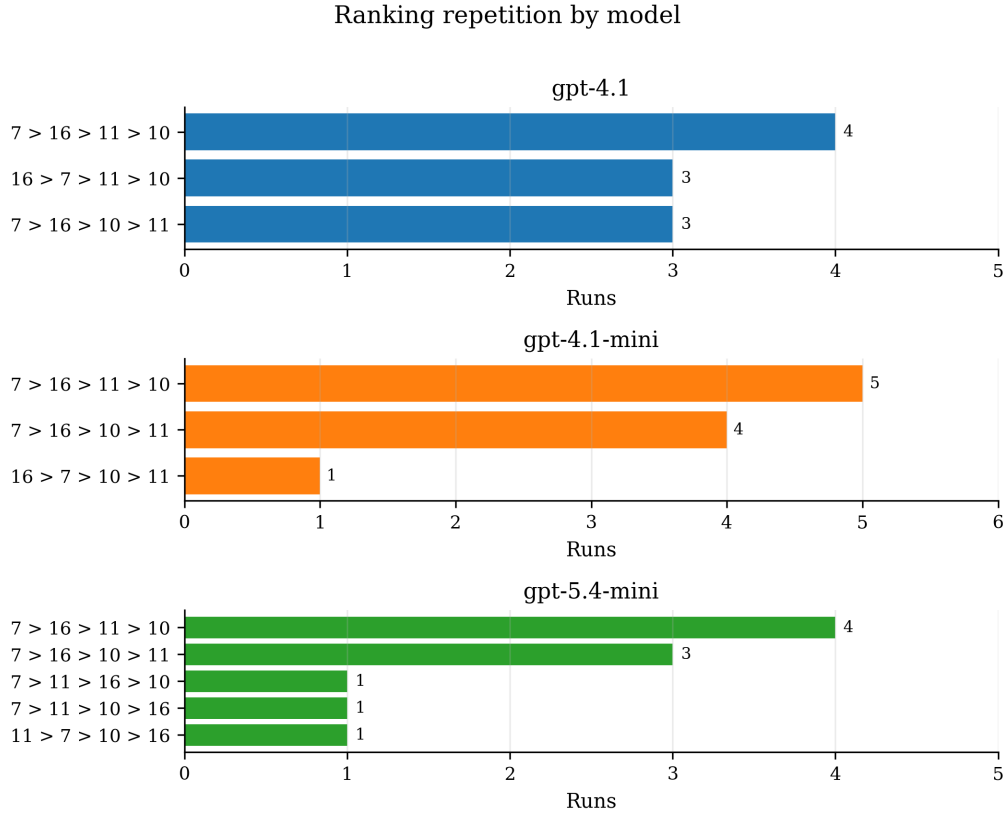


Figure 3. Frequency of rankings among frontier portfolios across three different models and ten runs each.

6.7. Comparison with a deterministic baseline

We compared the LLM assessments of the four reported Pareto portfolios with a simple and transparent weighted-keyword baseline. The baseline was built directly from the strategic plan. First, the text was converted to lowercase and split into individual words. Common words and general terms, such as “the”, “and”, “project”, “portfolio”, “strategic”, and “plan”, were removed because they provide little information about strategic alignment. The remaining words were ordered by how often they appeared in the strategic plan, and the 40 most frequent terms were retained. Thus, the terms were selected before scoring the projects and were not chosen to support any particular portfolio. Each retained term t received the following weight:

$$w_t = 1 + \log(f_t)$$

where f_t is the number of times the term appears in the strategic plan. This gives slightly more importance to terms that occur more often without allowing a few repeated terms to dominate the score. A project received the weight of a term when that term appeared in its description. Its raw keyword score was calculated as:

$$s_i = \frac{\sum_{t \in K} w_t I_{it}}{\sum_{t \in K} w_t}$$

where K is the set of 40 terms and $I_{it} = 1$ if term t appears in the description of project i , and $I_{it} = 0$ otherwise. The project scores were then normalized to the interval $[0, 100]$. Finally, the score of each portfolio was calculated as the mean score of its selected projects. Using the mean avoids rewarding a portfolio simply because it contains more projects. The keyword baseline reproduced the LLM ranking exactly: $7 > 16 > 11 > 10$. Spearman's rank correlation between the two rankings was $\rho = 1.00$. Table 2 reports the scores and ranks obtained with both methods.

Table 2. Comparison of the LLM and keyword-based portfolio assessments.

Portfolio	LLM score	LLM rank	Keyword score	Keyword rank
7	93	1	48.533	1
16	91	2	42.856	2
11	88	3	36.295	3
10	85	4	34.196	4

Table 3. Ten of the 40 terms used by the keyword baseline with the largest differences in contribution across portfolios.

Term	P7	P16	P11	P10	Range
Climate	1.196	0.652	0.359	0.399	0.837
Improve	0.701	0.287	0.946	1.052	0.765
Green	0.351	0.860	0.946	0.701	0.596
Energy	1.196	0.652	1.076	0.797	0.544
Infrastructure	0.000	0.463	0.000	0.000	0.463
Encourage	0.000	0.463	0.255	0.000	0.463
Environmental	0.566	0.926	1.018	0.848	0.452
Urban	0.283	0.694	0.255	0.283	0.440
Practices	0.701	0.287	0.315	0.351	0.414
Sustainable	0.701	0.287	0.315	0.701	0.414

Note: The values are components of the raw keyword score, not separate portfolio scores. All 40 terms were used to calculate the final scores. Only the ten terms with the largest variation across portfolios are shown.

To make the calculation easier to audit, Table 3 shows the ten terms that contributed most to differences between the four portfolio scores. These are not the only terms used: the final scores include all 40 terms. The ten terms shown are those whose contributions vary most across the four portfolios. For each term and portfolio, the table reports the term's weighted contribution. A larger value means that the term appears in a larger share of the portfolio's selected project descriptions. A value of zero means that the term does not appear in any of those descriptions. The range is the difference between the largest and smallest contribution across the four portfolios. Therefore, a larger range means that the term has more influence on differences between their keyword scores.

For example, "climate" makes a larger contribution to Portfolio 7 than to the other portfolios because it appears in a larger share of its selected project descriptions. In contrast, "infrastructure" contributes only to Portfolio 16. These examples allow the reader to inspect how the keyword matches affect the final ranking. The complete list of 40 terms, their frequencies, project matches, and portfolio-level contributions is available in the public repository. This comparison is limited to the four reported portfolios and does not repeat the optimization or rebuild the full Pareto frontier. It should therefore be interpreted as a consistency check of the reported portfolio assessments.

7. Results and analysis

The execution of our algorithm produced a set of non-dominated portfolios that form the Pareto frontier in the bi-objective space of expected present value and strategic alignment. Figure 4 displays this frontier, where each point represents a portfolio that cannot be improved in one objective without worsening the other. In the same figure, finalist portfolios not on the frontier are shown in grey. Likewise, Table 4 summarizes the cardinality, present value, strategic score, confidence level, and total cost for each non-dominated portfolio: IDs 7, 10, 11, and 16.

Table 4. *Pareto frontier portfolio analysis.*

ID	Selected Projects	Card.	Exp. PV	Strat. Score	Conf.	Cost
7	1, 2, 3, 6, 7, 10, 12, 15, 18	9	20,745	93	1.000	7,679
10	2, 3, 4, 6, 7, 8, 11, 12, 14	9	25,997	85	0.998	8,962
11	2, 3, 4, 6, 7, 8, 9, 11, 12, 17	10	25,678	88	0.994	8,979
16	1, 2, 3, 4, 6, 9, 12, 15, 16, 17, 19	11	24,696	91	0.905	9,033

The frontier includes four non-dominated portfolios, each representing a trade-off between financial return and strategic alignment. The portfolio with the highest expected PV (approximately 26,000 units) contains 9 projects and has a confidence level of 0.998, just below the 9,000-unit budget constraint. As we move along the Pareto curve, portfolios shift toward higher strategic alignment, culminating in a portfolio with a strategic score of

93 and a lower expected PV (20,800). This portfolio maintains full confidence (1.00) and a significantly lower cost (7,679 units), showing that strong strategic alignment can be achieved without exhausting the budget. Notably, portfolios with higher cardinalities (10-11 projects) occupy intermediate positions on the frontier. These options offer balanced performance in both objectives, with slightly lower confidence levels (0.994 and 0.905). This challenges the assumption that more projects automatically lead to better strategic outcomes and underscores the importance of careful project selection. The gap between the highest-PV solution (highlighted by a red dashed line) and the highest-strategy solution (green dashed line) illustrates the inherent trade-off between these objectives. This would be missed by a single-objective optimization. The visualization provides decision-makers with a clear overview of the available trade-offs, enabling more informed selection based on organizational goals and risk tolerance. The breakdown of execution times is shown in Table 5. These results confirm the feasibility and scalability of our hybrid approach. It is worth noting that the proposed framework does not require training or fine-tuning an LLM; the model is utilized exclusively at inference time as a semantic evaluator. Consequently, the intensive computational burden and data dependencies of LLM training are external to our optimization procedure. The framework is model-agnostic and computationally lightweight; in our experiments using the Azure-hosted GPT-4.1 Mini model, the complete workflow executed in under six minutes with a total monetary cost below one euro. The integration of stochastic simulation, GAs, and LLMs, allows for capturing both financial and strategic dimensions of portfolio selection under uncertainty. The resulting Pareto frontier offers a concise and interpretable set of high-quality portfolios for decision-makers.

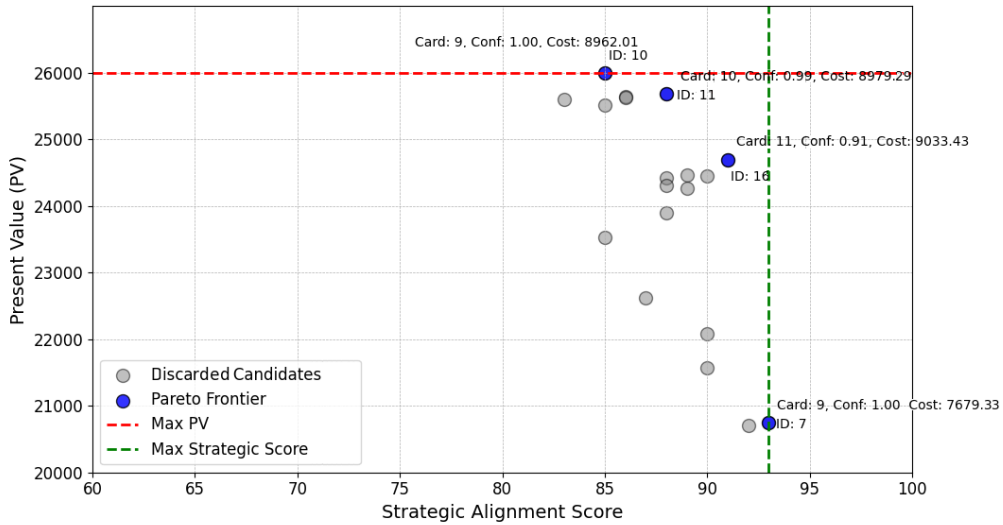


Figure 4. All finalist portfolios and extracted Pareto frontier.

Table 6 presents the results of a dual verification process using GPT-4.1 and MAI-DS-R1. Both models preserved the original ranking of portfolios, reinforcing the con-

Table 5. Computational times to execute the PPSP generative heuristic.

Step	Time (seconds)
Point estimate finished	161.7
Optimization step (GA)	14.7
Optimization step (higher cardinalities)	28.3
GPT-4 strategic assessments	58.2
Exploration of strategic neighborhood	79.3
Consolidation of final Pareto result	< 0.1
Total execution time	342.3

Table 6. Portfolio analysis with original results and dual verification scores.

ID	Selected Projects	Original Strat. Score	GPT-4.1 Verif.	MAI-DS-R1 Verif.
7	1, 2, 3, 6, 7, 10, 12, 15, 18	93	95	95
10	2, 3, 4, 6, 7, 8, 11, 12, 14	85	75	70
11	2, 3, 4, 6, 7, 8, 9, 11, 12, 17	88	78	72
16	1, 2, 3, 4, 6, 9, 12, 15, 16, 17, 19	91	80	82

GPT-4.1 Justifications

- **ID 7:** Covers all 10 strategic drivers with many strong contributions, including renewable energy, energy efficiency, sustainable agriculture, coastal protection, and climate resilience.
- **ID 10:** Covers 7 drivers well; strong in circular economy, green infrastructure, energy efficiency, renewable energy, and public transportation; lacks coverage in sustainable agriculture, coastal protection, and climate resilience.
- **ID 11:** Slightly improves on ID 10 with better support for climate education and energy efficiency; still lacks sustainable agriculture and coastal protection.
- **ID 16:** Covers 8 drivers; strong in coastal protection and circular economy, but weaker in water management and sustainable agriculture.

MAI-DS-R1 Justifications

- **ID 7:** Covers all 10 drivers with high-impact projects; minor weaknesses in P1/P3.
- **ID 10:** Misses Drivers 5, 7, and 10; includes low-impact projects (P8, P14).
- **ID 11:** Slightly better than ID 10 due to P9 but still misses key drivers.
- **ID 16:** Strong on coastal resilience and renewables but omits water/agriculture.

sistency of the approach. While verification scores differ slightly from original values, which is expected due to zero-shot evaluation and model differences, the relative order remained stable. This supports the conclusion that the proposed approach reliably captures key trade-offs.

8. Conclusions and future work

This paper introduces a novel framework, referred to as generative heuristics, that combines stochastic simulation, evolutionary optimization, and LLMs to address complex COPs involving both structured and semi-structured inputs. We frame this type of problem under the concept of ‘soft optimization’, where traditional methods fall short due to the presence of qualitative, textual, or ambiguous data elements that cannot be easily modeled with conventional objective functions. These semi-structured components require contextual interpretation and semantic reasoning, capabilities now feasible through the integration of LLMs. The proposed framework fills a gap in the literature by embedding LLMs directly into the optimization pipeline. This allows for the generation and evaluation of solutions that go beyond numeric performance, supporting the identification of portfolios or configurations that align with intangible objectives such as policy compliance, strategic alignment, or organizational values. To the best of our knowledge, this is the first attempt to combine simulation-based uncertainty modeling, metaheuristic optimization, and LLM-driven evaluation into a unified approach for decision support in soft optimization settings.

A structured review of existing research motivated and contextualized our approach, covering hybrid metaheuristics, LLM integration in optimization workflows, and recent advances in project portfolio selection modeling. The stochastic PPSP was selected as a representative use case to illustrate the framework’s capabilities. Within this context, we implemented a complete end-to-end process combining MCS for estimating uncertainty in costs and returns, a GA for portfolio generation, and a series of LLM-based evaluations to assess strategic alignment across varying portfolio sizes. The final output was filtered using a bi-objective Pareto analysis to identify solutions that optimally balance financial performance and strategic relevance. The computational experiments identified four non-dominated portfolios, whose performance was robust across both quantitative and qualitative dimensions. An external verification step was carried out using two distinct LLM configurations (GPT-4.1 and MAI-DS-R1) to ensure consistency and validate the semantic evaluations. These results confirmed that the proposed method is computationally efficient, cost-effective, and able to produce high-quality solutions that reflect both hard and soft decision criteria. The full procedure executed in under six minutes and at a cost of less than one euro.

This work contributes to the emerging area of AI-assisted optimization by showing that LLMs can serve not only as natural language generators, but also as semantic evaluators and decision-support tools within formal optimization frameworks. Enforcing rigid evaluation protocols and low-temperature decoding, allows the framework to treat the

LLM as a constrained evaluator rather than an open generator, mitigating hallucination risks. The generative heuristic framework provides a flexible structure to model problems where data cannot be fully reduced to numeric inputs, and where solution quality depends on both performance metrics and contextual fit. Future research may extend this framework to other problems such as vehicle routing, scheduling, or network design, particularly in settings where contextual or policy-related considerations play a role. Additional extensions could include the use of dynamic budget updates, progress monitoring over time, or expanded LLM-based assessments for risks, dependencies, and stakeholder alignment. The weighted-keyword comparison provides an initial consistency check for the Pareto frontier portfolios. Future research should extend this analysis to the complete candidate set, additional rule-based methods, expert assessments, and real-world datasets. Regarding the verification of semantic performance within the framework's evaluation layer, two major research lines emerge. First, the LLM-based strategic scores could be further benchmarked against further baselines, such as dictionary-based keyword matching procedures, traditional rule-based scoring matrices, or expert-defined multicriteria rubrics. Such comparisons will help isolate the exact marginal contribution of the LLM and identify the specific operational thresholds where advanced semantic evaluation justifies its computational overhead. Second, the generalizability of the framework should be tested on expanded datasets featuring organic, real-world strategic plans, uncurated project proposals, and diverse managerial assessments to validate its scalability and robustness in practical industrial decision-making contexts. Finally, conducting exhaustive sensitivity analyses under extensive prompt perturbations and developing real-time semantic noise filters remain vital directions to further bound LLM evaluation variability.

Declaration of generative AI and AI-assisted technologies in the writing process

During the preparation of this work the authors used GPT4.1 mini in order to improve the readability and language of the manuscript. After using this tool/service, the authors reviewed and edited the content as needed and take full responsibility for the content of the published article.

Acknowledgement

This work has been partially supported by the Spanish Ministry of Science, Innovation and Universities/AEI (BS-INV/GRC-24008_01 AIDigitalPrefer_PID2023, PID2025-169 830NB-I00, PID2022-138860NB-I00, AIA2025-163553-C44), and the Generalitat Valenciana (CIBEST 2026 135, CIAICO 117 2024).

References

- Adelantado, F., Ammouriova, M., Herrera, E., Juan, A. A., Shinde, S. S. and Tarchi, D. (2022). Internet of vehicles and real-time optimization algorithms: Concepts for vehicle networking in smart cities. *Vehicles* 4(4), 1223–1245.
- Aiche, A., Winer, Z. and Cohen, G. (2024). Constructing cybersecurity stocks portfolio using AI. *Forecasting* 6(4), 53.
- Akay, B., Karaboga, D. and Akay, R. (2022). A comprehensive survey on optimizing deep learning models by metaheuristics. *Artificial Intelligence Review* 55(2), 829–894.
- Allen, B., Polat, F. and Groth, P. (2024). Shroom-indelab at semeval-2024 task 6: Zero- and few-shot llm-based classification for hallucination detection. In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*, pp. 839–844.
- Alsofyani, M. and Wang, L. (2025). Evaluating ChatGPT's strengths and limitations for data race detection in parallel programming via prompt engineering. *The Journal of Supercomputing* 81(6), 776.
- Amodeo, L., Talbi, E.-G. and Yalaoui, F. (2018). *Recent developments in metaheuristics*. Springer.
- Azevedo, B. F., Rocha, A. M. A. and Pereira, A. I. (2024). Hybrid approaches to optimization and machine learning methods: a systematic literature review. *Machine Learning* 113(7), 4055–4097.
- Bayliss, C. (2021). Machine learning based simulation optimisation for urban routing problems. *Applied Soft Computing* 105, 107269.
- Bengio, Y., Ducharme, R. and Vincent, P. (2000). A neural probabilistic language model. *Advances in Neural Information Processing Systems* 13.
- Boschetti, M. A. and Maniezzo, V. (2024). Contemporary approaches in matheuristics an updated survey. *Annals of Operations Research* 343(2), 663–700.
- Calvet, L., Benito, S., Juan, A. A. and Prados, F. (2023). On the role of metaheuristic optimization in bioinformatics. *International Transactions in Operational Research* 30(6), 2909–2944.
- Calvet, L., de Armas, J., Masip, D. and Juan, A. A. (2017). Learnheuristics: Hybridizing metaheuristics with machine learning for optimization with dynamic inputs. *Open Mathematics* 15(1), 261–280.
- Calvo, C., Ivorra, C. and Liern, V. (2017). Fuzzy portfolio selection with non-financial goals. *Annals of Operations Research* 258, 153–173.
- Castaneda, J., Ghorbani, E., Ammouriova, M., Panadero, J. and Juan, A. A. (2022). Optimizing transport logistics under uncertainty with simheuristics: Concepts, review and trends. *Logistics* 6(3), 42.
- Cesarone, F., Martino, M. L. and Carleo, A. (2022). Does ESG impact really enhance portfolio profitability? *Sustainability* 14(4), 2050.

- Chen, H., Zhu, J., Wang, W., Zhu, Y. and Xi, L. (2025). Triplet-based contrastive method enhances the reasoning ability of large language models. *The Journal of Supercomputing* 81(4), 1–20.
- Chen, L., Darko, A., Zhang, F., Chan, A. P. and Yang, Q. (2025). Can large language models replace human experts? Effectiveness and limitations in building energy retrofit challenges assessment. *Building and Environment* 276, 112891.
- Chen, P. and Li, Z. (2025). Length instruction fine-tuning with chain-of-thought (LIFT-COT): Enhancing length control and reasoning in edge-deployed large language models. *Electronics* 14(8), 1662.
- Chica, M., Juan, A. A., Bayliss, C., Cerdón, O. and Kelton, W. D. (2020). Why Simheuristics?: Benefits, limitations, and best practices when combining metaheuristics with simulation. *SORT: Statistics and Operations Research Transactions* 44(2), 0311–334.
- Cinkusz, K., Chudziak, J. A. and Niewiadomska-Szynkiewicz, E. (2025). Cognitive agents powered by large language models for agile software project management. *Electronics* 14(1), 87.
- Coeckelbergh, M. (2025). LLMs, Truth, and Democracy: An overview of risks. *Science and Engineering Ethics* 31(1), 1–13.
- Coelho e Silva, L., Fonseca, G. d. F. and Castro, P. A. L. (2024). Transformers and attention-based networks in quantitative trading: a comprehensive survey. In *Proceedings of the 5th ACM International Conference on AI in Finance*, pp. 822–830.
- Crawford, B., Soto, R., Lemus-Romani, J., Becerra-Rozas, M., Lanza-Gutiérrez, J. M., Caballé, N., Castillo, M., Tapia, D., Cisternas-Caneo, F., García, J., Astorga, G., Castro, C. and Rubio, J.-M. (2021). Q-learnheuristics: Towards data-driven balanced metaheuristics. *Mathematics* 9(16), 1839.
- de Zarzà, I., de Curtò, J., Roig, G. and Calafate, C. T. (2023). LLM multimodal traffic accident forecasting. *Sensors* 23(22), 9225.
- Doering, J., Kizys, R., Juan, A. A., Fito, A. and Polat, O. (2019). Metaheuristics for rich portfolio optimisation and risk management: Current state and future trends. *Operations Research Perspectives* 6, 100121.
- Dokeroglu, T., Kucukyilmaz, T. and Talbi, E.-G. (2024). Hyper-heuristics: A survey and taxonomy. *Computers & Industrial Engineering* 187, 109815.
- Fortin, F.-A., Rainville, F.-M. D., Gardner, M.-A., Parizeau, M. and Gagné, C. (2012, jul). DEAP: Evolutionary algorithms made easy. *Journal of Machine Learning Research* 13, 2171–2175.
- Goli, A. and Singh, A. (2024). Frontiers: Can large language models capture human preferences? *Marketing Science* 43(4), 709–722.
- Gopal, A. (2024). Neuralfactors: A novel factor learning approach to generative modeling of equities. In *Proceedings of the 5th ACM International Conference on AI in Finance*, pp. 99–107.

- Guo, C., Tian, Z., Tang, J., Li, S. and Wang, T. (2025). Multi-pattern retrieval-augmented framework for Text-to-SQL with Poincaré-Skeleton retrieval and meta-instruction reasoning. *Information Processing & Management* 62(3), 103978.
- Hebenstreit, K., Praas, R., Kieseewetter, L. P. and Samwald, M. (2024). A comparison of chain-of-thought reasoning strategies across datasets and models. *PeerJ Computer Science* 10, e1999.
- Homayouni, S. M., Fontes, D. B. and Gonçalves, J. F. (2023). A multistart biased random key genetic algorithm for the flexible job shop scheduling problem with transportation. *International Transactions in Operational Research* 30(2), 688–716.
- Iranmanesh, S., Shakhshi-Niaei, M. and Rastegar, H. (2019). A multi-objective optimization model for project portfolio selection considering aggregate complexity: A case study. *International Journal of Optimization in Civil Engineering* 9(2), 291–311.
- Juan, A. A., Ammouriova, M., Faulin, J., Panadero, J. and Ferone, D. (2025). Simheuristics. In *Handbook of heuristics*, pp. 335–361. Springer.
- Kalyuzhnaya, A., Mityagin, S., Lutsenko, E., Getmanov, A., Aksenkin, Y., Fatkhiev, K., Fedorin, K., Nikitin, N. O., Chichkova, N., Vorona, V. and Boukhanovsky, A. (2025). LLM agents for smart city management: Enhancing decision support through multi-agent ai systems. *Smart Cities* 8(1), 19.
- Kapania, S., Wang, R., Li, T. J.-J., Li, T. and Shen, H. (2025). 'I'm Categorizing LLM as a Productivity Tool': Examining Ethics of LLM Use in HCI Research Practices. *Proceedings of the ACM on Human-Computer Interaction* 9(2), 1–26.
- Katoch, S., Chauhan, S. S. and Kumar, V. (2021). A review on genetic algorithm: Past, present, and future. *Multimedia Tools and Applications* 80, 8091–8126.
- Kelley Jr, J. E. and Walker, M. R. (1959). Critical-path planning and scheduling. In *Papers presented at the December 1-3, 1959, eastern joint IRE-AIEE-ACM Computer Conference*, pp. 160–173.
- Kim, J. H. (2023). What if ChatGPT were a quant asset manager. *Finance Research Letters* 58, 104580.
- Kosten, C., Nooralahzadeh, F. and Stockinger, K. (2025). Evaluating the effectiveness of prompt engineering for knowledge graph question answering. *Frontiers in Artificial Intelligence* 7, 1454258.
- Li, M., Zhou, H., Yang, H. and Zhang, R. (2024). RT: A retrieving and chain-of-thought framework for few-shot biomedical named entity recognition. *Journal of the American Medical Informatics Association* 31(9), 1929–1939.
- López-Soto, D., Angel-Bello, F., Yacout, S. and Alvarez, A. (2017). A multi-start algorithm to design a multi-class classifier for a multi-criteria ABC inventory classification problem. *Expert Systems with Applications* 81, 12–21.
- Lucas, T., Smith, J. and Johnson, E. (2024). Reasoning with large language models for medical question answering. *Journal of the American Medical Informatics Association* 31(9), 1964–1975.

- Mao, J.-y., Pan, Q.-k., Miao, Z.-h. and Gao, L. (2021). An effective multi-start iterated greedy algorithm to minimize makespan for the distributed permutation flow-shop scheduling problem with preventive maintenance. *Expert Systems with Applications* 169, 114495.
- Martin, X. A., Herrero, R., Juan, A. A. and Panadero, J. (2024). An agile adaptive biased-randomized discrete-event heuristic for the resource-constrained project scheduling problem. *Mathematics* 12(12), 1873.
- Min, B., Ross, H., Sulem, E., Veyseh, A. P. B., Nguyen, T. H., Sainz, O., Agirre, E., Heintz, I. and Roth, D. (2023). Recent advances in natural language processing via large pre-trained language models: A survey. *ACM Computing Surveys* 56(2), 1–40.
- Mohagheghi, V., Mousavi, S. M. and Mojtahedi, M. (2020). Project portfolio selection problems: Two decades review from 1999 to 2019. *Journal of Intelligent & Fuzzy Systems* 38(2), 1675–1689.
- Mohagheghi, V., Mousavi, S. M. and Vahdani, B. (2017). A new multi-objective optimization approach for sustainable project portfolio selection: a realworld application under interval-valued fuzzy environment. *Iranian Journal of Fuzzy Systems* 13(6), 41–68.
- Mustata, M., Iliescu, D., Mavriş, E. and Jude, C.-R. (2025). ChatGPT-assisted decision-making: An in-depth exploration of the Human-AI interaction. *International Journal of Human-Computer Interaction* 41(5), 421–438.
- Parker, M. J., Anderson, C., Stone, C. and Oh, Y. (2025). A large language model approach to educational survey feedback analysis. *International journal of artificial intelligence in education* 35(2), 444–481.
- Ranjbar, M., Nasiri, M. M. and Torabi, S. A. (2022). Multi-mode project portfolio selection and scheduling in a build-operate-transfer environment. *Expert Systems with Applications* 189, 116134.
- Reyes-Rubiano, L., Ferone, D., Juan, A. A. and Faulin, J. (2019). A simheuristic for routing electric vehicles with limited driving ranges and stochastic travel times. *SORT* 1, 3–24.
- Rezk, S. S. and Selim, K. S. (2024). Metaheuristic-based ensemble learning: An extensive review of methods and applications. *Neural Computing and Applications* 36(29), 17931–17959.
- Saiz, M. and Calvet, L. (2025). Combining LLMs and Simheuristics: An application to the project portfolio selection problem. In *Decision Sciences - Second Decision Science Alliance International Summer Conference, DSA ISC 2024, Valencia, Spain, June 6-7, 2024, Proceedings, Part I*, pp. 3–13. Springer.
- Saiz, M., Calvet, L., Juan, A. A. and Lopez-Lopez, D. (2024). A simheuristic for project portfolio optimization combining individual project risk, scheduling effects, interruptions, and project risk correlations. *Computers & Industrial Engineering* 198, 110694.
- Saiz, M., Lopez-Lopez, D., Calvet, L. and Juan, A. A. (2026). A genetic algorithm simheuristic for solving the stochastic project portfolio selection problem with portfolio reliability constraints. *International Transactions in Operational Research* 33(6), 4039–4071.

- Saiz, M., Lostumbo, M. A., Juan, A. A. and Lopez-Lopez, D. (2022). A clustering-based review on project portfolio optimization methods. *International Transactions in Operational Research* 29(1), 172–199.
- Sangaiah, A. K., Javadpour, A., Pinto, P., Rezaei, S. and Zhang, W. (2023). Enhanced resource allocation in distributed cloud using fuzzy meta-heuristics optimization. *Computer Communications* 209, 14–25.
- Simpson, S., Nukpezah, J., Brooks, K. and Pandya, R. (2025). Parity benchmark for measuring bias in llms. *AI and Ethics* 5(3), 3087–3101.
- Sun, H., Liu, X., Bian, Y., Zhu, P. and Cheng, D. (2024). Asymmetric graph-based deep reinforcement learning for portfolio optimization. In *Machine Learning and Knowledge Discovery in Databases*, pp. 174–189. Springer.
- Sun, W., Zhang, L. and Chen, M. (2024). Improving intermediate reasoning in zero-shot chain-of-thought for text classification. *Neurocomputing* 550, 119–130.
- Talbi, E.-G. (2021). Machine learning into metaheuristics: A survey and taxonomy. *ACM Computing Surveys (CSUR)* 54(6), 1–32.
- Tarczewski, T., Stojic, D. and Dzielinski, A. (2025). Large language model-based tuning assistant for variable speed pmsm drive with cascade control structure. *Electronics* 14(2), 232.
- Tepelyan, R. and Gopal, A. (2023). Generative machine learning for multivariate equity returns. In *Proceedings of the Fourth ACM International Conference on AI in Finance*, pp. 159–166.
- Tian, Y., Lin, F., Li, Y., Zhang, T., Zhang, Q., Fu, X., Huang, J., Dai, X., Wang, Y., Tian, C., Li, B., Lv, Y., Kovács, L. and Wang, F.-Y. (2025). Uavs meet llms: Overviews and perspectives towards agentic low-altitude mobility. *Information Fusion* 122, 103158.
- Zhang, G., Chen, L., Zhang, Y., Liu, Y., Ge, Y. and Cai, X. (2024). Translating words to worlds: Zero-shot synthesis of 3D terrain from textual descriptions using large language models. *Applied Sciences* 14(8), 3257.
- Zhuang, X. and Wu, Y. (2025). Large language models in corporate investment: An analysis of ChatGPT’s transformative applications, strategic impacts, and future opportunities. *IT Professional* 27(2), 21–27.

