

Article

Integrating Large Language Models and Optimization in Semi-Structured Decision Making: Methodology and a Case Study

Gianpaolo Ghiani ^{*,†} , Gianluca Solazzo [†]  and Gianluca Elia [†] 

Dipartimento di Ingegneria dell'Innovazione, Università del Salento, Via per Monteroni, 73100 Lecce, Italy; gianluca.solazzo@unisalento.it (G.S.); gianluca.elia@unisalento.it (G.E.)

* Correspondence: gianpaolo.ghiani@unisalento.it

† These authors contributed equally to this work.

Abstract: Semi-structured decisions, which fall between highly structured and unstructured decision types, rely on human intuition and experience for the final choice, while using data and analytical models to generate tentative solutions. These processes are traditionally iterative and time-consuming, requiring cycles of data gathering, analysis, and option evaluation. In this study, we propose a novel framework that integrates Large Language Models (LLMs) with optimization techniques to streamline such decision-making processes. In our approach, LLMs leverage their capabilities in data interpretation, common-sense reasoning, and mathematical modeling to assist decision makers by reducing cognitive load. They achieve this by automating aspects of information processing and option evaluation, while preserving human oversight as a crucial component of the final decision-making process. Another significant strength of our framework lies in its potential to drive the evolution of a new generation of decision support systems (DSSs). Unlike traditional systems that rely on rigid and inflexible interfaces, our approach enables users to express their preferences in a more natural, intuitive, and adaptable manner, substantially enhancing both usability and accessibility. A case study on last-mile delivery system design in a smart city demonstrates the practical application of this framework. The results suggest that our approach has the potential to simplify the decision-making process and improve efficiency by reducing cognitive load, enhancing user experience, and facilitating more intuitive interactions.



Citation: Ghiani, G.; Solazzo, G.; Elia, G. Integrating Large Language Models and Optimization in Semi-Structured Decision Making: Methodology and a Case Study. *Algorithms* **2024**, *17*, 582. <https://doi.org/10.3390/a17120582>

Academic Editor: Abdulsalam Yassine

Received: 16 November 2024

Revised: 10 December 2024

Accepted: 12 December 2024

Published: 16 December 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: semi-structured decisions; human-in-the-loop; knowledge discovery; large language models; last-mile logistics

1. Introduction

Starting from the seminal work of Nobel laureate Herbert A. Simon [1] and Gorry and Morton's influential paper [2], managerial decisions have been classified into *structured*, *semi-structured*, and *unstructured*. Structured decisions include those that follow a clear, predefined process or set of rules, often relying on established criteria or quantitative data (e.g., inventory management and production scheduling). Unstructured managerial decisions include those that are complex, novel, and lack a clear or predefined process. These decisions often require judgment, intuition, and creativity, as they deal with uncertain or ambiguous situations. Examples include strategic planning, crisis management, market entry decisions, and innovation initiatives. Various studies have indicated that a significant portion of them fall into the semi-structured category. For instance, in his landmark book [3], management guru Henry Mintzberg highlighted that the majority of decisions are neither entirely routine (structured) nor completely novel (unstructured). In particular, he emphasized that such a category is quite common in middle management, where decisions often involve both a structured component (e.g., financial analysis) and an unstructured part (e.g., judgment on market conditions).

Semi-structured managerial decisions depend on human intuition and experience for the final judgment, while utilizing data and analytical models to generate a sequence of tentative solutions that progressively incorporate the decision makers' preferences. Hence, they involve a combination of standardized procedures (which are peculiar to structured decisions) and human assessment (which is typical of unstructured decisions).

They are common in managerial and strategic contexts where decisions cannot be entirely algorithmic and require a nuanced approach. Examples include budgeting decisions (see, e.g., [4]), where standard procedures can be used for creating budgets, while managerial judgment is needed to allocate funds based on strategic priorities and future projections. Another example is marketing campaign design (see, e.g., [5]), where decisions about targeting demographics and allocating resources among various media require both data analysis (e.g., past campaign performance) and creative thinking. A further remarkable example is investment decisions (see, e.g., [6,7]), in which evaluating investment opportunities involves the quantitative analysis of financial data as well as the qualitative assessment of market conditions and future trends.

The iterative nature of semi-structured decision making involves repeated cycles of gathering data, analyzing information, generating and evaluating options, and refining decisions [8]. In this scheme, managers continuously adjust and incorporate their preferences in the tentative decisions, based on the insights gained in the previous iterations. This repeated and cyclical process is inherently time-consuming as each iteration requires decision makers to generate tentative solutions, analyze them carefully, and possibly integrate new elements (e.g., penalty terms in the objective function and new constraints) in the decision model (see Section 2 for more details).

Preferences may reflect those of the organization's managers or incorporate the needs of various other stakeholders, including workers, customers, local communities, and others. Hereafter, we refer to the group involved in the decision-making process as the "decision makers". It is worth noting that all these subjects do not necessarily have to physically participate in the decision board. Their preferences may be expressed in person or can take the form of product reviews, formal complaint reports, etc., collected in the company information system.

In this paper, we propose a new algorithmic approach to semi-structured decision making, based on the integration of *Large Language Models* (LLMs) and *Optimization*, to assist managers in the decision-making process. This new approach has the potential to reduce the cognitive effort required to generate, evaluate, and refine options, and arrive at a decision. Another major strength of our framework lies in its potential to pave the way for a new generation of *decision support systems* (DSSs). Unlike traditional systems that depend on rigid and inflexible interfaces, our approach allows users to articulate their preferences in a more natural, intuitive, and adaptable way, significantly improving both usability and accessibility.

1.1. State of the Art

To the best of our knowledge, this is the first contribution dealing with the integration of LLMs and *optimization in semi-structured decision making*. However, several recent articles explore how LLMs can support decision-making processes in diverse ways.

A first line of research aims at using LLMs to provide recommendations based on textual documentation. This is useful in a number of sectors in which LLMs can assist professionals in navigating complex documents and generate contextually relevant insights. In particular, in the healthcare field, LLMs can analyze patient records, medical literature, clinical guidelines, and research papers to interpret symptoms and recommend personalized treatment plans or medical interventions (see, e.g., [9]). In the legal sector, LLMs can be used to process and recommend actions from complex legal documents, such as contracts, case law, or statutes (see, e.g., [10]). In the financial and banking sectors, LLMs can analyze financial reports, market news, and research documents to provide investment recommendations or insights into market trends (see, e.g., [11]).

A second line of research explores the use of LLMs in translating natural language descriptions of *structured* managerial decisions into optimization models. The goal is to increase the accessibility and usability of optimization solvers by allowing non-experts to interface with them using natural language. A number of papers appeared in this field after the Institute for *Operations Research and the Management Sciences* (INFORMS) [12] in partnership with *Meta AI* [13] launched the Natural Language for Optimization (NL4Opt) Competition in 2023 (see [14]), to bridge the gap between natural language processing (NLP) and optimization. Relevant contributions in this field include [15–21].

1.2. Structure of the Paper

The remainder of this paper is organized as follows. In Section 2, we model preference discovery in semi-structured decision making as an iterative process. In Section 3, we illustrate how an LLM can be integrated with an optimization algorithm to support a semi-structured decision-making process, depending on the type of optimization algorithm used (e.g., constructive procedure, parameterized heuristic, or model-based solver). Finally, the new approach is illustrated in Section 4 through a case study related to the design of a last-mile delivery system in a smart city context. Conclusions and future research directions follow in Section 5.

2. Preference Discovery in Semi-Structured Decision Making

A key aspect of semi-structured decision making is that preferences are not initially fully known to the decision makers themselves but are *discovered* through an iterative process in which the decision makers gain decisive insights on their preferences, goals, and constraints. In this *preference discovery* process, decision makers progressively enhance their understanding of what drives optimal decisions until a satisfactory solution is generated. The decision makers often start by “generating” an initial solution (let us say $\mathbf{x}^{(1)}$) whose *quantifiable* performance measures are evaluated analytically with a closed-form expression, a simulation, a system dynamics model, etc. (see [22] for a review of model-driven DSSs). Based on this initial solution, the decision makers reflect on their preferences (which we denote as π in the following), understanding which aspects (metrics) of the solution align with their values and which do not. The decision makers then identify qualitative factors that were not initially considered but are important for a more comprehensive solution. These new insights are used to generate a new solution that hopefully reflects better their preferences. The new solution is then assessed with regard to quantifiable metrics. This process is repeated, with a new solution $\mathbf{x}^{(h)}$ at each cycle h , embedding a more refined understanding of what is important to the decision makers. This iterative process helps in managing the complexity of real-world decisions by breaking down the problem and progressively building up a more complete solution. Figure 1 describes this iterative process.

When the decision is combinatorial (or continuous) in nature (see, e.g., [23,24]), feasible solutions cannot be enumerated because they are infinite or huge in number. In this case, the structured part of the decision-making process leads to an *optimization problem* in which only core quantitative factors are considered. Under these hypotheses, the semi-structured decision can be seen as the solution of an *optimization problem involving decision maker preferences*:

$$(\mathcal{P}) \quad \min z = f^\pi(\mathbf{x}) \quad (1)$$

s.t.

$$g_i(\mathbf{x}) = 0, \quad i = 1, \dots, m \quad (2)$$

$$h_r^\pi(\mathbf{x}) = 0, \quad r = 1, \dots, q_\pi \quad (3)$$

In this formulation, both the objective function $f^\pi(\mathbf{x})$ (assumed to be minimized) and constraints $h_r^\pi(\mathbf{x}) = 0$ depend on the preferences π of the decision makers and acknowledge the role of human tacit knowledge and contextual factors that affect the decision-making process.

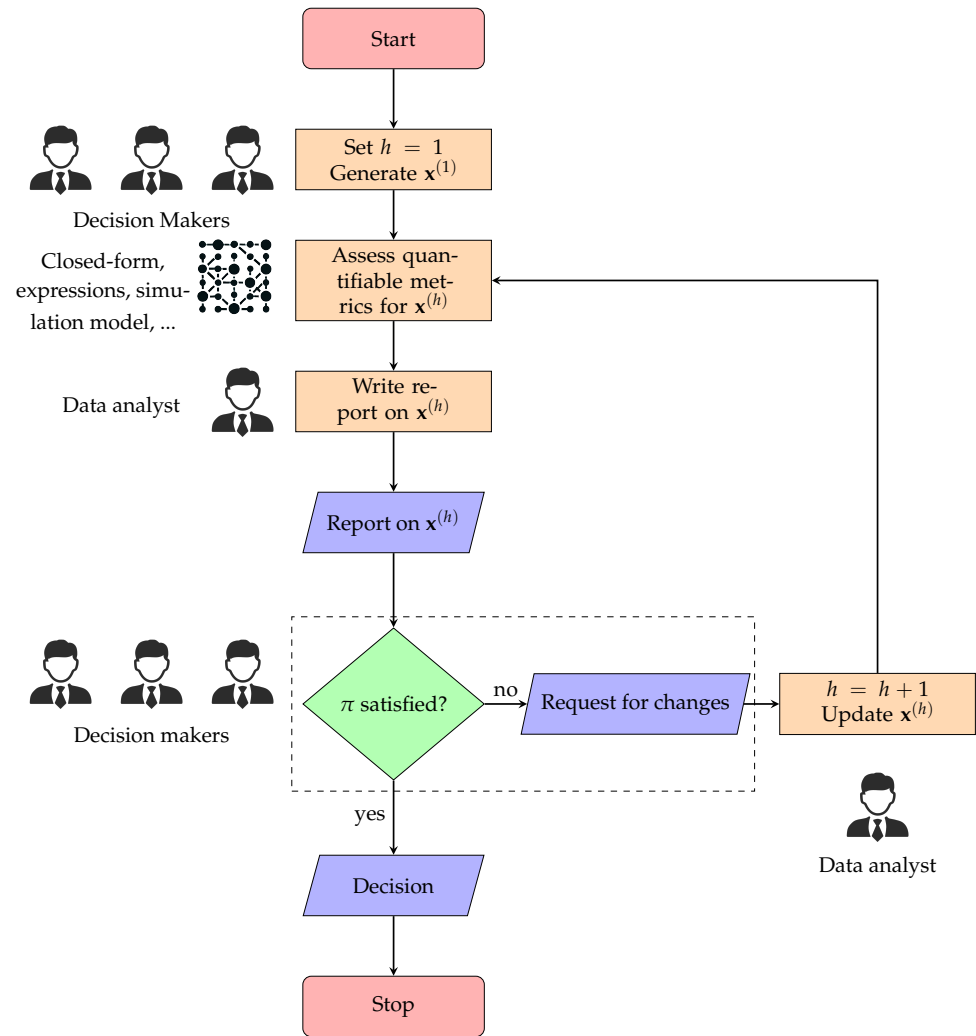


Figure 1. Preference discovery in semi-structured decision making.

More formally, at the initial iteration ($h = 0$), no preferences (or some initially known preferences) are considered (which is denoted as π_0 in the following) and only the structured component of the decision-making process is accounted for. Hence, the following optimization problem is solved

$$(\mathcal{P}^{(0)}) \quad \min z = f^{\pi_0}(\mathbf{x}) \tag{4}$$

s.t.

$$g_i(\mathbf{x}) = 0, \quad i = 1, \dots, m \tag{5}$$

where $f^{\pi_0}(\mathbf{x})$ represents an approximation of $f^\pi(\mathbf{x})$, in which no preferences are taken into account, and the constraints are independent of preferences.

Once a solution $\mathbf{x}^{(0)}$ of $P^{(0)}$ is obtained, it is analyzed by the decision makers and evaluated in light of their preferences. Whenever this solution does not consider some of the preferences, the decision makers devise a refined approximation $f^{\pi_1}(\mathbf{x})$ of the “true” objective function $f^\pi(\mathbf{x})$, and possibly an expanded set of constraints $h_r^{\pi_1}(\mathbf{x}), r = 1, \dots, q_{\pi_1}$. Then, the new approximation is solved, its solution analyzed, and so on. We now report the model at a generic iteration $h = 1, \dots$ for future reference in this article:

$$(\mathcal{P}^{(h)}) \quad \min z = f^{\pi_h}(\mathbf{x}) \tag{6}$$

s.t.

$$g_i(\mathbf{x}) = 0, \quad i = 1, \dots, m \tag{7}$$

$$h_r^{\pi_h}(\mathbf{x}) = 0, \quad r = 1, \dots, q_{\pi_h} \tag{8}$$

where (6) and (8) are approximations of (1) and (3), respectively, based on the latest updates π_h of the preferences π elicited by the decision maker. Figure 2 illustrates how the general iterative scheme in Figure 1 is specialized whenever the structured part of the decision-making process leads to an optimization problem.

This scheme is highly adaptable and can be tailored to different optimization approaches, whether using a constructive heuristic, a parameterized heuristic, or a model-based solver. For a constructive heuristic, the LLM can generate solutions directly, allowing the straightforward integration of on-the-fly preferences. With a parameterized heuristic, new preferences can be incorporated by adjusting specific parameters within the heuristic. For a model-based solver, updated preferences from decision makers may lead to adding or removing constraints or modifying weights within the objective function. Further details are provided in Section 3.

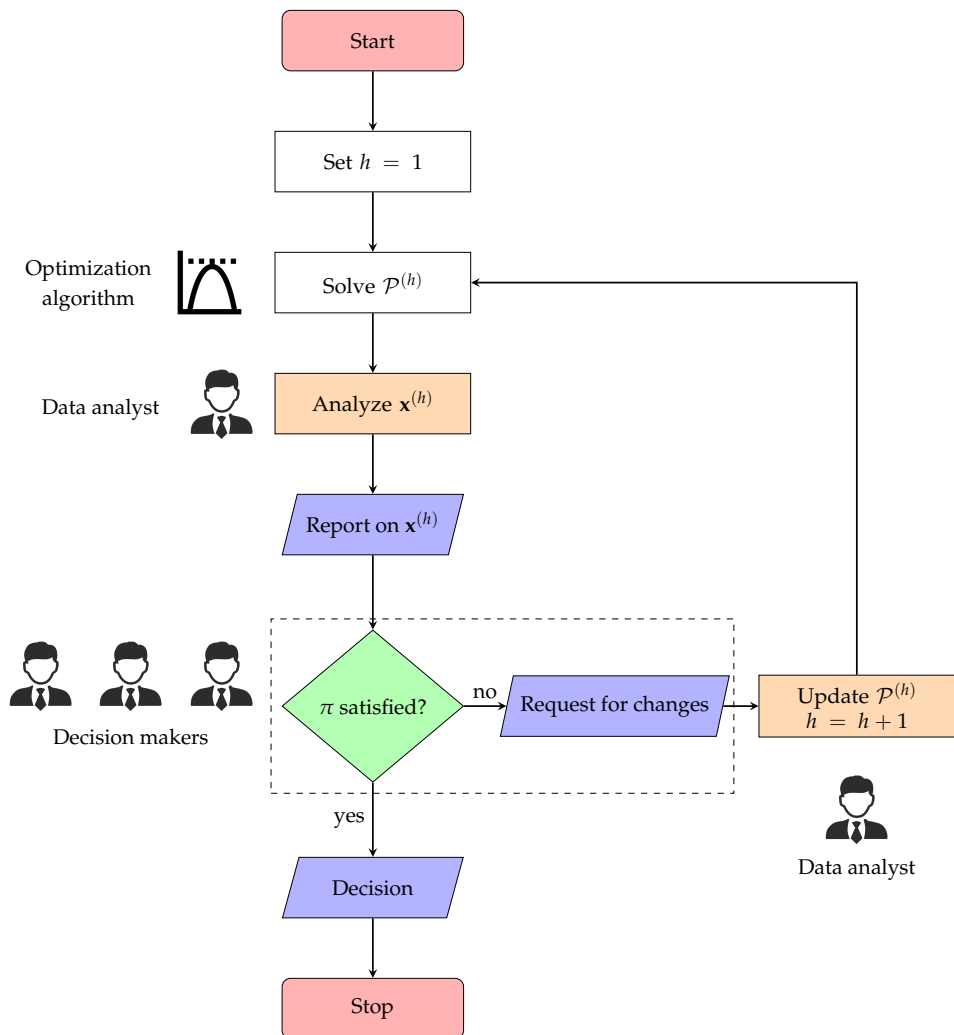


Figure 2. A semi-structured decision-making process whose structured component leads to an optimization problem.

3. Integrating the LLM and the Optimization Algorithm

In this section, we propose a novel framework that integrates *Large Language Models* with *optimization* techniques to streamline semi-structured decision-making processes. The discussion addresses cases where the optimization algorithm is a constructive heuristic (Section 3.1), a parameterized heuristic (Section 3.2), or a model-based solver (Section 3.3). First, we present an overview of the LLM features that are pertinent to the analysis that follows.

Large Language Models (LLMs) are *Generative AI* systems trained on extensive text datasets to understand and generate human language with high proficiency (refer to [25] for a comprehensive review of architectures and applications, along with a discussion of open issues and challenges). They use *deep learning* architectures, primarily *transformers*, to capture complex linguistic patterns and contextual information. LLMs exhibit the so-called *emergent properties*, unexpectedly sophisticated abilities that arise as the models scale in size, complexity, and volume of training data. These capabilities, often not explicitly programmed, include skills like (a) in-context learning, (b) common-sense reasoning, (c) mathematical problem-solving, and (d) data summarization and interpretation. These properties are of the utmost importance across various applications, particularly in advancing the next generation of DSSs, as detailed in the following. In these systems, LLMs have the potential to enhance the flexibility of optimization tools, making them more adaptive and responsive to complex, real-world semi-structured decision-making scenarios.

In our framework, LLMs (see Figure 3) aim at automating two phases of the decision cycle described in Figure 1: (a) the analysis and interpretation of the solution $\mathbf{x}^{(h)}$ of the current approximation (6)–(8); and (b) the update of problem $\mathcal{P}^{(h)}$ ($r = 1, \dots, q_{\pi_h}$) on the basis of the qualitative remarks expressed by the decision makers in a natural language. Both tasks are guided by a high-level description of the problem and a set of preliminary instructions provided to the LLM. Further details on this step are presented in the next subsection.

The interaction between the decision makers, the optimization solver, and the LLM is based on the *Prompt-Before-Prompt* (PBP) technique (also known as *pre-prompt*). PBP involves providing an initial prompt that helps to set the context, frame the subsequent prompts, or provide a set of instructions. In particular, the pre-prompt can give explicit directions or specify the desired format of the response, which helps the model to follow the guidelines more closely. This can lead to more accurate, coherent, and contextually appropriate responses from the LLM. In our framework, PBP is used to instruct the LLM about the preferences of the decision makers and how they relate to the data, variables, constraints and objective function of the optimization model. For this purpose, the decision is framed in terms of *entities* (such as customers, orders, and vehicles) and the *relationships* between them. This approach allows us to inform the LLM that, for example, customer requests are geographically grouped into districts, enabling the LLM to interpret solutions within the context of these districts. Additionally, it facilitates the automatic derivation of modifications to the optimization problem in response to updates in decision makers' preferences.

We now detail the three cases when optimization is approached using a constructive heuristic, a parameterized heuristic, or a model-based solver.

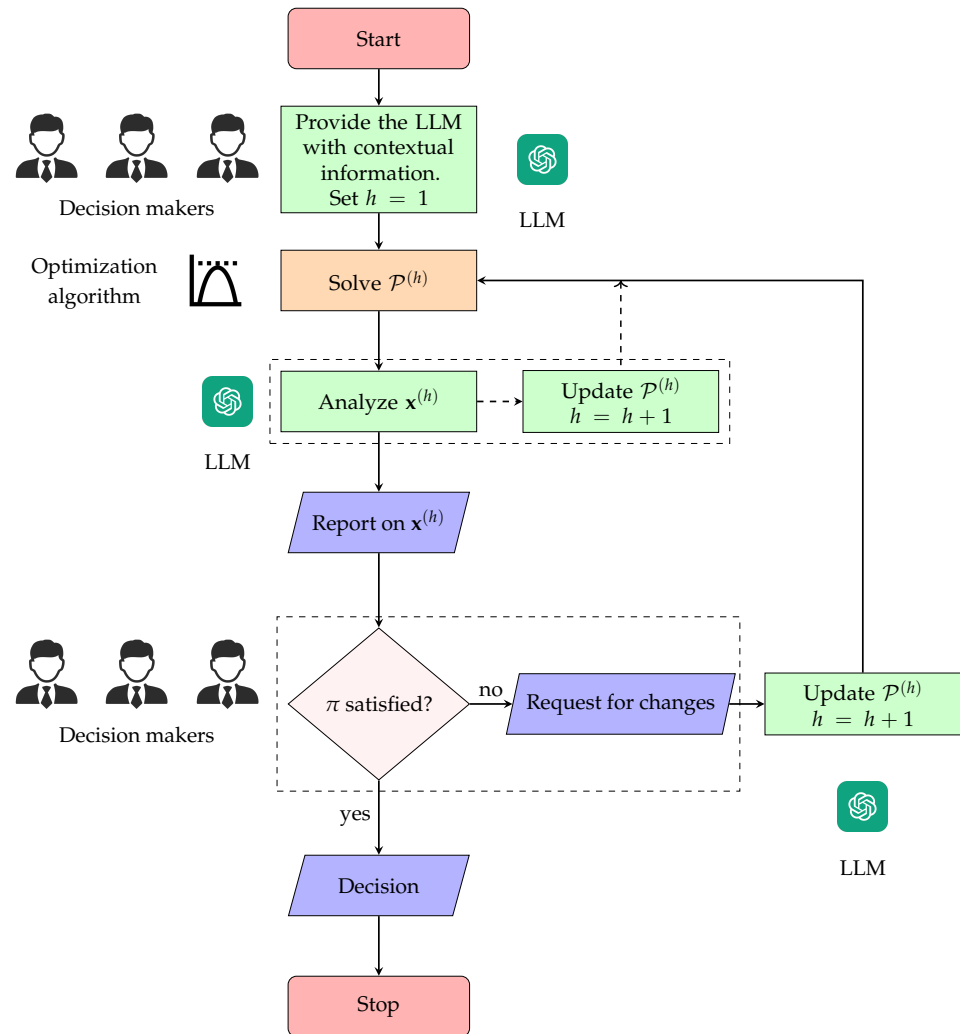


Figure 3. Integrating an LLM in a semi-structured decision-making process where the structured part is an optimization problem.

3.1. Integrating a Constructive Heuristic

Constructive heuristics are sequential algorithms that build a solution step-by-step, typically by making locally optimal choices at each stage. These heuristics start from an empty or partial solution and incrementally add elements based on a specific rule or criteria until a complete solution is formed. LLMs can implement a constructive heuristic if given a clear textual description of the step-by-step procedure. By processing instructions and contextual details about the problem, the LLM can follow the sequential steps required to build a solution, making decisions at each stage according to the described criteria. This allows the model to generate feasible solutions incrementally, as guided by the heuristic’s rules. The LLM can also generate the code for the heuristic in a specified programming language (e.g., Python), which can then be executed by an interpreter or runtime environment. By translating the heuristic’s logic into code, the model enables automated execution, allowing for efficient testing, adaptation, and integration within larger systems. This approach enables the seamless incorporation of on-the-fly user preferences into the solution construction process. By dynamically modifying steps based on real-time input, the system can tailor solutions more precisely to evolving user needs, enhancing flexibility and responsiveness within the heuristic framework.

3.2. Integrating a Parameterized Heuristic

Alternatively, if optimization is approached through a parameterized heuristic, new preferences may lead to adjustments of specific parameters within the heuristic itself. This is the case, e.g., when the problem is multi-objective in nature and the decision maker wants to explore the trade-off between n (conflicting) objectives, say z_1, \dots, z_n . In this case, the heuristic could be designed to optimize a weighted combination $z = w_1 x_1 + \dots + w_n x_n$ of the objectives z_1, \dots, z_n , with weights w_1, \dots, w_n . Here, an LLM could be utilized to adjust the parameters w_1, \dots, w_n based on customer preferences, effectively managing the trade-offs among these objectives.

3.3. Integrating a Model-Based Optimization Solver

If optimization is approached using a model-based solver, gathering new preferences from decision makers may result in adding or removing constraints, or adjusting weights within the objective function. Hence, a fundamental step of the decision cycle illustrated in Figure 3 is the derivation of a refined version of functions $f^{\pi_h}(\mathbf{x})$ and $h_r^{\pi_h}(\mathbf{x})$ ($r = 1, \dots, q_{\pi_h}$) based on the newly gained insights.

As with all human activities that depend on intuition and experience, this step defies easy formalization. Nonetheless, a common feature is that decision makers conceptualize their decisions in terms of “entities” and semantic features, rather than solely focusing on decision variables and constraints. We will elucidate this concept in the following.

Model (1)–(3), and its approximations (6)–(8), are *extended* representations ([26]) in which the decision is expressed by a fixed vector of variables \mathbf{x} . On the other hand, humans analyze data and reason about decisions in terms of problem components (“entities”), such as orders, customers, and vehicles, and their relationships. Thus, we need a representation that is closer to what humans do, namely, a *compact* representation in which objects and their various relationships can be described explicitly. As the authors of [26] note: “Almost everything that humans express in natural language concerns objects and their relationships”. For this reason, we reformulate model (1)–(3), by using an *algebraic modeling language* (e.g., AMPL [27], GAMS [28], or OPL [29]). Such languages rely on sets of objects (referred to as *entities* from now on), as is also very common in the context of computer science [30]). Entities are classified as *fundamental* or *derived*, where a derived set is defined as subsets or Cartesian products of other sets. Then, each variable, constraint, and parameter of the MIP model is *indexed* by one or more entities in the corresponding algebraic modeling. For instance, in a last-mile distribution problem, entity sets may include a set of orders O and a set of vehicles V , in which case a binary variable indicating whether an order $o \in O$ is assigned to a vehicle $v \in V$ will be referred to as x_{ov} instead of some x_j . Similarly, a constraint indicating that the capacity of a vehicle $v \in V$ cannot be exceeded by a given amount will be labeled by index v . Based on this labeling, humans (and LLMs) find easy-to-interpret data and solutions, and relate them to their preferences.

4. Case Study

We apply the framework presented in Section 3 to a case study in last-mile logistics, focusing on the design of an innovative delivery system. In this application setting, the objective is to determine optimal locations for parking slots (which serve as transshipment points) and to establish efficient vehicle routes and schedules. This decision-making process is semi-structured, as it involves balancing the preferences of multiple stakeholders and addressing conflicting objectives. In this case study, the role of the LLM was twofold: (1) to serve as a constructive heuristic for the location selection component, embedding stakeholder preferences, and (2) to adjust the parameters of a multi-objective metaheuristic [31] for optimizing vehicle routing and scheduling.

4.1. Decision Problem

Designing city logistics involves developing a comprehensive strategy to efficiently and sustainably manage the movement of goods within urban areas. The goal is to minimize

costs, reduce congestion and environmental impacts, and ensure timely deliveries [32]. The case study is related to the design of an innovative last-mile distribution system based on a combination of traditional vans and *Autonomous Delivery Robots (ADRs)* [33]. ADRs are self-driving vehicles intended to transport goods from one location to another without human intervention. These robots utilize advanced technologies such as artificial vision, sensors, and GPS to navigate and perform their delivery tasks efficiently. Different companies have carried out pilot initiatives for testing ADRs: for example, *Starship Technologies* is running a self-driving robot service in the city of London [34], as well as *e-Novia* and *Twinswheel* in different cities [35]. Recently, *Aramex* launched a self-driving delivery robot [36], while *FedEx* experimented its six-wheeled autonomous delivery robot, called *Roxo SameDay Bot* [37].

In our application setting, the decision makers want to design a last-mile distribution system that utilizes a fleet of vans, each equipped with a sub-fleet of ADRs, to drop off small parcels at customer locations. Deliveries can be performed by directly visiting customers with the van or by parking the van at designated locations (*parking slots*, PSs), from which the final leg of the delivery is completed either on foot by the driver or by ADRs. ADRs can autonomously navigate sidewalks at pedestrian speed to deliver parcels to customers' doors.

Formally, the problem is defined on a directed multigraph $G = (V, A)$ where V is the vertex set and A is the arc set. The vertex set is partitioned into three subsets: $V = \{0\} \cup P \cup C$, with 0 representing the van depot, P denoting the set of available parking slots, and C comprising the set of customers to be serviced within a planning horizon of length T . The depot has a time window $[0, T]$, and each parking slot $p \in P$ has a number of availability windows of the form $[e_p, l_p]$ during which parking reservations can be made. Each customer $c \in C$ has a demand q expressed as the number of parcels to be delivered. For each pair of vertices $i, j \in V$, set A contains two arcs: $(i, j)_d$ for driving and $(i, j)_w$ for walking (traversed by either a driver or an ADR). The travel time for walking arcs is calculated based on constant speeds v_{dri} and v_{adr} for drivers and ADRs, respectively. The driving time $\tau_{ij}(t)$ varies according to the time of day and depends on departure time t from node i [38]. Customer requests must be satisfied by a fleet of homogeneous vans, each with a capacity Q_{van} and equipped with m ADRs. ADRs have a capacity Q_{adr} and a maximum travel distance L_{adr} per full charge. Drivers, who can also perform deliveries on foot, have a capacity Q_{dri} per walking trip and a maximum daily walking distance L_{dri} . The cost structure includes daily fixed costs for vans, drivers, and ADRs, as well as variable costs per kilometer for vans. Each customer must be visited once, vans must return to the depot before the end of the day, and capacity and distance constraints must be satisfied.

The decision-making process is divided into two main components:

- *Parking Slot Selection (PSS)*, a semi-structured decision, which consists in locating k out of $|P|$ parking slots (see [39] for a review on location models). This decision involves human judgment to incorporate implicit preferences and contextual factors. LLMs can assist in generating insights to inform the decision-making process and suggest alternative locations for parking slots based on stakeholders' preferences and feedback.
- *Vehicle Routing and Scheduling (VRS)*, a structured decision, which consists in minimizing the sum of fixed and variable vehicle cost, for a given set of potential parking slots [31]. This structured component is not influenced by decision maker preferences and focuses solely on achieving the best operational efficiency.

The problem is inherently multicriteria, with objectives that include the following: $z_1 \propto k$, representing the occupation of public space; z_2 , the total cost; and z_3 , an environmental objective aimed at minimizing CO₂ and other pollutant emissions.

The qualitative measures considered by decision makers in the PSS stage can help ensure that the chosen locations for parking slots optimize the logistical aspects of last-mile delivery while also accommodating the preferences of the stakeholders (the company's management, delivery drivers, recipients, and local communities) affected by the company's operations. Preferences may include the following:

- (a) Ease of parking for delivery vehicles (e.g., avoid parking on sloping streets);
- (b) Proximity to main roads and motorways (e.g., minimize the number of intersections to be crossed);
- (c) Traffic congestion considerations;
- (d) Feedback from delivery drivers and personnel;
- (e) Ease of unloading parcels and ADRs from vehicles;
- (f) Condition of the parking area (e.g., paved or well-maintained);
- (g) Safety of the location for drivers;
- (h) Security of parked vehicles against theft or damage;
- (i) Compatibility with different vehicle sizes and types;
- (j) Flexibility to adapt to changes in delivery routes;
- (k) Significant opposition from residents due to the excessive concentration of parking slots dedicated to last-mile deliveries.

It is worth remembering that these aspects are part of the tacit knowledge of the decision makers, and cannot be easily formalized in performance measures and constraints.

Another relevant aspect is that the preferences of certain stakeholders, including company managers, representatives of drivers' unions, and local communities, may be conveyed directly to the decision support system in person. Others (such as the preferences of individual drivers, citizens, and recipients) may be reflected in formal complaint reports, social media posts, or local newspaper articles, and inputted in the system on behalf of the authors.

4.2. Implementation

To demonstrate that LLMs can be successfully integrated with optimization in semi-structured decision making, we developed a prototype implementation using ChatGPT-4o [40], a version of OpenAI's language model that was released in May 2024, along with a heuristic for the park-and-multi-loop vehicle routing and scheduling problem with autonomous delivery robots [31] and LangChain [41], a powerful framework that facilitates seamless interaction between LLMs and other applications. The prototype architecture comprises several layers, including the following:

- The *User Interface Layer*, which provides a user-friendly dashboard for interacting with the system. Users can initiate optimization tasks, visualize the results of the analysis performed by the LLM, and provide high-level directions in natural language.
- *Orchestration Middleware*, based on the LangChain Python library, which acts as the intermediary between the LLM and the Java application implementing the optimization algorithm. It coordinates the data flow, ensuring that insights from the LLM are appropriately applied to the optimization model.
- The *LLM Integration Layer*, which provides access to the LLM for processing natural language prompts and generating responses through API.
- The *Optimization Layer*, which implements the VRS algorithm presented in [31]. It exposes RESTful endpoints for initiating optimization tasks, receiving weights w_2 and w_3 from the LLM, and retrieving optimization results.

The interaction between stakeholders, the LLM, and the optimization algorithm is illustrated in Figure 4.

In this case study, the role of the LLM was twofold: (1) to serve as a constructive heuristic for the location selection component, embedding stakeholder preferences, and (2) to adjust the parameters of a multi-objective metaheuristic [31] for optimizing vehicle routing and scheduling.

In an initial phase, the LLM was given a prompt-before-prompt describing the decision in terms of entities and relationships among them. We also outlined the PSS constructive heuristic to be implemented, as well as the role of weights w_2 and w_3 in the parameterized heuristic [31], which are used to control the trade-off between objectives z_2 and z_3 .

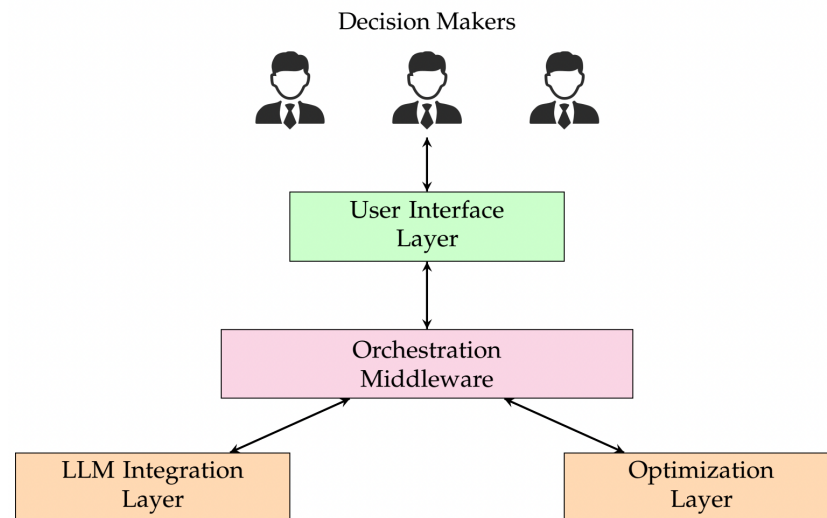


Figure 4. Architecture of the DSS.

As a PSS constructive heuristic, we implemented a simple procedure. We asked the LLM to divide the service territory in a 3×3 grid with given values of latitude and longitude at the borders. For a given k , the LLM was required to select the parking slots according to two criteria: assuring a balanced customer/parking slot ratio for each sub-zone (*parking slot density* criterion), and maximizing the proximity of the selected parking slots to customers (*parking slot average distance from customer* criterion).

After examining the solution to the VRS problem for a certain set of parking slots, the decision makers can ask—on the basis of their preferences—to increase or reduce a certain objective, the replacement of a specific parking slot with a valid alternative, etc. The LLM reviews the requests and possibly modifies $z_1 = k$, identifies a new pool of parking slots, and sets different weights for z_2 and z_3 . Then, the optimization module generates a new solution to the VRS problem, and so on.

4.3. Experimentation

To test the DSS prototype, a real-world case scenario was built with 500 pharmacies to be served daily in a metropolitan area in Rome (Italy), delimited by a latitude between 41.8538 and 41.8814, and a longitude between 12.4397 and 12.5070. A preliminary analysis allowed 325 potential parking slots and one van depot to be identified (see Figure 5).

The value of the new approach can be measured with respect to several aspects, including (a) accuracy, (b) consistency and stability, (c) time saved and reduction in the cognitive effort of the decision makers, and (d) perception of flexibility experienced by end users. We conducted a test in a simulated environment, in which we were able to evaluate only (a) and (b), leaving an assessment of (c) and (d) to a future on-the-field study with real end users.

To verify whether the LLM correctly captures the DMs' qualitative preferences and integrates them into the optimization algorithm effectively (criterion (a)), we generated (based on the logistics experience of the authors) ten fictitious stakeholder requests for eight of the most common categories of preferences listed in Section 4.1, which resulted into eighty statements. Specifically, we formulated a range of requests commonly raised by various stakeholders—such as drivers, citizens, and customers—when deploying decision support systems in city logistics design projects. The preferences were expressed in Italian and articulated with the utmost clarity and linguistic precision.

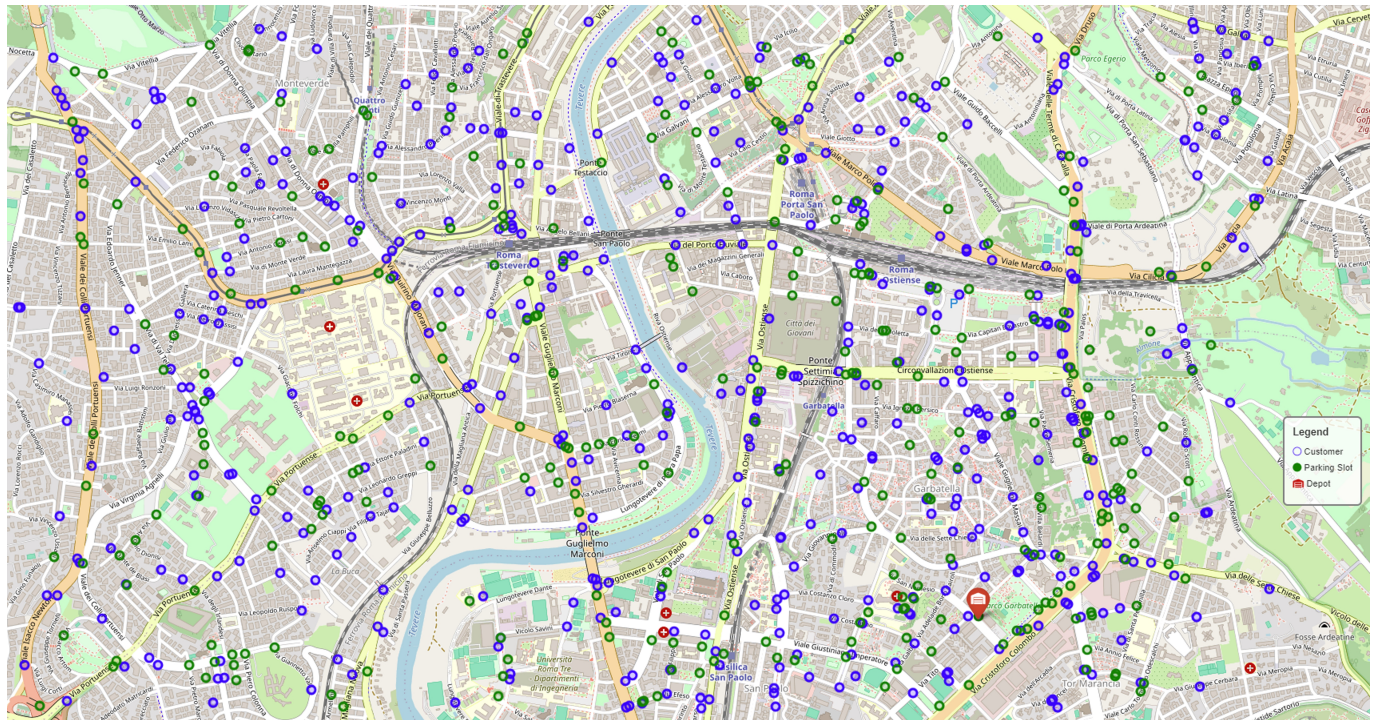


Figure 5. Spatial distribution of customers and parking slots.

To assess the consistency and stability of the DSS (criterion (b)), an LLM (other than the one used in the prototype) was used to randomly modify each request 5 times with decreasing levels of clarity and coherence (the last incorporating dialectal elements). The success rate was then evaluated subjectively by the authors of the study and is reported in Table 1. The results show that, with perfectly articulated requests (degradation level 0) and slightly less clear ones (degradation level 1), accuracy is at its peak (98.8%). As the degradation level increases, accuracy declines slightly but steadily. When informal phrasing and regional vocabulary are introduced (degradation level 5), accuracy falls more sharply to 91.3%. As a conclusion, the DSS performs very well under optimal and near-optimal conditions but experiences noticeable difficulty as the quality of input degrades, particularly at the highest degradation level.

Table 1. Accuracy across different degradation levels of requests.

Degradation Levels	Accuracy (%)
0 (perfectly articulated requests)	98.8
1	98.8
2	97.5
3	96.3
4	95.0
5 (informal phrasing, regional vocabulary)	91.3

As a side note, in the phase of analyzing the solution of problems $\mathcal{P}^{(h)}$, the LLM presented to the decision maker a report including graphical summaries like those reported in Figure 6 (describing the spatial distribution of customers and parking slots) and Figure 7 (describing parking slot densities and parking slot average distances from customers). Based on this information, the users decided whether to accept the current solution $\mathbf{x}^{(h)}$ or request the inclusion of additional preferences into the optimization model.

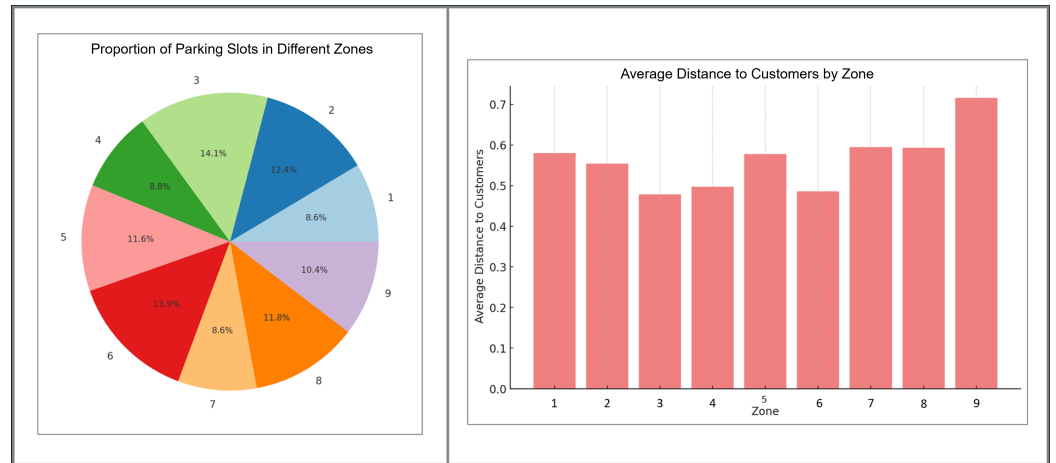


Figure 6. Parking slot densities and parking slot average distances from customers in zones 1, . . . , 9 computed by the LLM and presented to the decision makers for evaluation.

Parking Slot Density				Parking Slot Average Distance From Customers					
Zone	Customers	Parking Slots	Customer/Parking Slot Ratio	Parking Slot	LAT	LONG	Zone	Avg Distance to Customers	
1	43	32	1.3	1	501	41.8782	12.5052	8	0.8916
2	62	31	2.0	2	502	41.8556	12.4486	0	0.6442
3	71	61	1.2	3	503	41.8631	12.4888	5	0.7182
4	44	22	2.0	4	504	41.8755	12.4878	7	0.5895
5	58	39	1.5	5	505	41.8701	12.4634	7	0.9788
6	70	53	1.3	6	506	41.8643	12.4476	5	0.5903
7	43	26	1.7	7	507	41.8769	12.4639	7	0.6057
8	59	37	1.6	8	508	41.9593	12.4805	8	0.7889
9	52	24	2.2	9	509	41.8587	12.4481	0	0.6198
				10	510	41.8645	12.4991	5	0.7103

Figure 7. Graphical summary of parking slot densities and parking slot average distances from customers computed by the LLM and presented to the decision makers for evaluation. In particular, the table on the left reports, for each zone, the number of customers, the number of parking slots, and the number of customers per parking slot. The table on the right reports, for each parking slot, the latitude and longitude as well as the zone and the average distance from the customers of the associated zone.

5. Discussion and Conclusions

In this paper, we presented a novel approach designed to assist organizations in semi-structured decision-making processes. Our method aims to significantly reduce the cognitive effort required to analyze solutions, evaluate alternatives, and make informed decisions.

A key advantage of our framework is its potential to serve as a foundation for a new generation of decision support systems. Unlike conventional systems that rely on rigid and restrictive interfaces, our approach enables users to express their preferences in a more natural, intuitive, and flexible manner, enhancing usability and accessibility.

In previous contributions, LLMs have been used to assist decision makers by generating recommendations among a *restricted* number of alternatives, based, e.g., on the latest research, clinical guidelines, and medical literature. On the contrary, our approach is well suited for combinatorial or continuous decision problems where feasible solutions are either infinite or exceedingly numerous and must also be evaluated using quantitative criteria.

We have demonstrated the effectiveness of our framework through a case study focused on designing a last-mile delivery system within a smart city setting. The results demonstrate excellent performance when operating under optimal and near-optimal conditions, maintaining a high degree of accuracy. However, as the quality of the input deteriorates, the system’s performance begins to decline. This decline becomes particularly pronounced at the highest level of degradation, where informal phrasing and the use of regional vocabulary introduce significant challenges. These results highlight the system’s

strong reliability in well-structured scenarios while also underscoring the importance of improving its robustness to handle less standardized or noisier inputs effectively.

Looking ahead, we acknowledge several limitations of this study that warrant attention in future research efforts. First, we aim to implement and test our system in real-world contexts to assess its practical effectiveness and adaptability. Additionally, we intend to explore the applicability of our approach across various decision-making domains, such as marketing and finance, to evaluate its versatility and broader impact.

Moreover, we plan to extend our framework to incorporate model-based optimization, where the role of the LLM would expand to include dynamically modifying a baseline optimization model. This could involve adjusting weights, adding or removing constraints, and other modifications based on new preferences articulated in natural language by decision makers.

A further open area of investigation includes determining whether fine-tuning the LLM would enhance its capability for this advanced functionality.

Addressing these questions will provide valuable insights into optimizing the DSS for diverse and complex decision-making scenarios.

Author Contributions: Conceptualization, G.G. and G.S.; methodology, G.G. and G.S.; software, G.S.; validation, G.G., G.S. and G.E.; writing—review and editing, G.G., G.S. and G.E. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been partly supported by Ministero dell’Università e della Ricerca (MUR) of Italy, research program “Sustainable Mobility Center” (Centro Nazionale per la Mobilità Sostenibile-CN MOST), Spoke 7, CN00000023

Data Availability Statement: Data are available from authors upon request.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

- Simon, H. *The New Science of Management Decision*; Harper & Row: New York, NY, USA, 1960.
- Gorry, G.A.; Morton, M.S.S. A framework for management information systems. *MIT Sloan Manag. Rev.* **1989**, *30*, 49.
- Mintzberg, H. *The Nature of Managerial Work*; Harper & Row: New York, NY, USA, 1973.
- Kida, T.E.; Moreno, K.K.; Smith, J.F. The influence of affect on managers’ capital-budgeting decisions. *Contemp. Account. Res.* **2001**, *18*, 477–494.
- Balducci, B.; Marinova, D. Unstructured data in marketing. *J. Acad. Mark. Sci.* **2018**, *46*, 557–590. [[CrossRef](#)]
- Nickell, S.J. *The Investment Decisions of Firms*; CUP Archive: Cambridge, UK, 1978.
- Korniotis, G.M.; Kumar, A. Do older investors make better investment decisions? *Rev. Econ. Stat.* **2011**, *93*, 244–265. [[CrossRef](#)]
- Kaliardos, W.N. *Semi-Structured Decision Processes: A Conceptual Framework for Understanding Human-Automation Systems*. Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 1999.
- Harskamp, R.E.; De Clercq, L. Performance of ChatGPT as an AI-assisted decision support tool in medicine: A proof-of-concept study for interpreting symptoms and management of common cardiac conditions (AMSTELHEART-2). *Acta Cardiol.* **2024**, *79*, 358–366. [[CrossRef](#)] [[PubMed](#)]
- Zhang, Z.; Xie, H.; Yu, Z. LegalBERT: Pre-trained Bidirectional Encoder Representations from Transformers Model for Legal Text Mining. *arXiv* **2019**, arXiv:1908.03602.
- Araci, D. FinBERT: A Pretrained Language Model for Financial Communications. In Proceedings of the 2019 Financial Technology Conference (FinTech 2019), Bergen, Norway, 2 December 2019.
- INFORMS. The Institute for Operations Research and the Management Sciences (INFORMS). 2024. Available online: <https://www.informs.org/> (accessed on 10 December 2024).
- Meta. Advancing AI Research and Innovation. 2024. Available online: <https://ai.meta.com/> (accessed on 10 December 2024).
- INFORMS and Meta AI. The Natural Language for Optimization (NL4Opt) Competition. 2023. Available online: <https://nl4opt.github.io/> (accessed on 10 November 2024).
- Ahmed, T.; Choudhury, S. LM4OPT: Unveiling the potential of Large Language Models in formulating mathematical optimization problems. *INFOR Inf. Syst. Oper. Res.* **2024**, *62*, 559–572. [[CrossRef](#)]
- Ramamonjison, R.; Yu, T.; Li, R.; Li, H.; Carenini, G.; Ghaddar, B.; He, S.; Mostajabdaveh, M.; Banitalebi-Dehkordi, A.; Zhou, Z.; et al. NL4opt competition: Formulating optimization problems based on their natural language descriptions. In Proceedings of the NeurIPS 2022 Competition Track, PMLR, New Orleans, LA, USA, 28 November–9 December 2023; pp. 189–203.

17. Mostajabdaveh, M.; Yu, T.T.; Ramamonjison, R.; Carenini, G.; Zhou, Z.; Zhang, Y. Optimization modeling and verification from problem specifications using a multi-agent multi-stage LLM framework. *INFOR Inf. Syst. Oper. Res.* **2024**, *62*, 599–617. [[CrossRef](#)]
18. Astorga, N.; Liu, T.; Xiao, Y.; van der Schaar, M. Autoformulation of Mathematical Optimization Models Using LLMs. *arXiv* **2024**, arXiv:2411.01679.
19. AhmadiTeshnizi, A.; Gao, W.; Udell, M. Optimus: Optimization modeling using mip solvers and large language models. *arXiv* **2023**, arXiv:2310.06116.
20. Jiang, C.; Shu, X.; Qian, H.; Lu, X.; Zhou, J.; Zhou, A.; Yu, Y. LLMOPT: Learning to Define and Solve General Optimization Problems from Scratch. *arXiv* **2024**, arXiv:2410.13213.
21. AhmadiTeshnizi, A.; Gao, W.; Brunborg, H.; Talaei, S.; Udell, M. OptiMUS-0.3: Using Large Language Models to Model and Solve Optimization Problems at Scale. *arXiv* **2024**, arXiv:2407.19633.
22. Power, D.J.; Sharda, R. Model-driven decision support systems: Concepts and research directions. *Decis. Support Syst.* **2007**, *43*, 1044–1061. [[CrossRef](#)]
23. Du, D.; Pardalos, P.M. *Handbook of Combinatorial Optimization*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 1998; Volume 4.
24. Ruszczyński, A. *Nonlinear Optimization*; Princeton University Press: Princeton, NJ, USA, 2011.
25. Raiaan, M.A.K.; Mukta, M.S.H.; Fatema, K.; Fahad, N.M.; Sakib, S.; Mim, M.M.J.; Ahmad, J.; Ali, M.E.; Azam, S. A review on large Language Models: Architectures, applications, taxonomies, open issues and challenges. *IEEE Access* **2024**, *12*, 26839–26874. [[CrossRef](#)]
26. Russell, S.; Norvig, P. *Artificial Intelligence: A Modern Approach*, 3rd ed.; Prentice-Hall: Hoboken, NJ, USA, 2010.
27. Fourer, R.; Gay, D.M.; Kernighan, B.W. A Modeling Language for Mathematical Programming. *Manag. Sci.* **1990**, *36*, 519–554. [[CrossRef](#)]
28. Rosenthal, R.E. *GAMS—A User’s Guide*; GAMS Development Corporation: Washington, DC, USA, 2017.
29. Van Hentenryck, P. *The OPL Optimization Programming Language*; MIT Press: Cambridge, UK, 1999.
30. Batini, C.; Ceri, S.; Navathe, S. *Conceptual Database Design—An Entity-Relationship Approach*; The Benjamin/Cummings Publishing Company, Inc.: Redwood City, CA, USA, 1992.
31. Adamo, T.; Ghiani, G.; Guerriero, E.; Manni, E.; Solazzo, G. Park-and-multi-loop with autonomous delivery robots in last-mile logistics. In *Technical Report*; Transportation Research: East Liberty, OH, USA, 2024.
32. Ghiani, G.; Laporte, G.; Musmanno, R. *Introduction to Logistics Systems Management: With Microsoft Excel and Python Examples*; John Wiley & Sons: Hoboken, NJ, USA, 2022.
33. Hossain, M. Autonomous delivery robots: A literature review. *IEEE Eng. Manag. Rev.* **2023**, *51*, 77–89. [[CrossRef](#)]
34. Starship. Starship’s Self-Driving Robot. 2020. Available online: <http://www.starship.xyz> (accessed on 30 June 2023).
35. Chen, C.; Demir, E.; Huang, Y.; Qiu, R. The adoption of self-driving delivery robots in last mile logistics. *Transp. Res. Part E Logist. Transp. Rev.* **2021**, *146*, 102214. [[CrossRef](#)] [[PubMed](#)]
36. Geronimo, A. Aramex Tests Drone and Robot Delivery Service in Dubai. 2023. Available online: <https://www.itp.net/emergent-tech/aramex-tests-drone-and-robot-delivery-service-in-dubai> (accessed on 30 June 2023).
37. FedEx. Meet Roxo, the FedEx SameDay Bot. 2020. Available online: <https://www.fedex.com/en-cw/about/sustainability/our-approach/roxo-delivery-robot.html> (accessed on 30 June 2023).
38. Adamo, T.; Gendreau, M.; Ghiani, G.; Guerriero, E. A review of recent advances in time-dependent vehicle routing. *Eur. J. Oper. Res.* **2024**, *319*, 1–15. [[CrossRef](#)]
39. Farahani, R.Z.; SteadieSeifi, M.; Asgari, N. Multiple criteria facility location problems: A survey. *Appl. Math. Model.* **2010**, *34*, 1689–1709. [[CrossRef](#)]
40. OpenAI. GPT-4. Available online: <https://openai.com/index/gpt-4/> (accessed on 10 December 2024).
41. LangChain. LangChain: Build, Run, and Manage LLM Applications. 2024. Available online: <https://www.langchain.com> (accessed on 24 September 2024).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.