

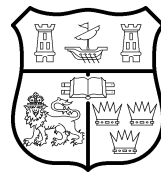
Title	A non-Bayesian preference elicitation approach for noisy decision models
Authors	Pourkhajouei, Samira
Publication date	2025
Original Citation	Pourkhajouei, S. 2025. A non-Bayesian preference elicitation approach for noisy decision models. MSc Thesis, University College Cork.
Type of publication	Masters thesis (Research)
Rights	© 2025, Samira Pourkhajouei. - https://creativecommons.org/licenses/by/4.0/
Download date	2026-10-24 23:35:43
Item downloaded from	https://hdl.handle.net/10468/17935

A Non-Bayesian Preference Elicitation Approach for Noisy Decision Models

Samira Pourkhajouei

PHD MATHEMATICS

**Thesis submitted for the degree of
Master of Science**



NATIONAL UNIVERSITY OF IRELAND, CORK

COLLEGE OF SCIENCE, ENGINEERING AND FOOD SCIENCE

SCHOOL OF COMPUTER SCIENCE & INFORMATION TECHNOLOGY

INSIGHT CENTRE FOR DATA ANALYTICS

July 2025

Head of Department: Prof. Dirk Pesch

Supervisors: Dr. Nic Wilson
Dr. Steven Prestwich

Research supported by Insight Centre for Data Analytics and Science
Foundation Ireland under Grant number 12/RC/2289-P2.

Contents

List of Tables	iii
List of Plots	iv
Abstract	vi
Acknowledgements	viii
1 Introduction	1
1.1 Multi-criteria Decision-making	1
1.2 Preference Elicitation	2
1.3 Outline and Contributions	5
1.4 Publications	8
2 Background and Overview of Preference Elicitation Methods	9
2.1 Multi-Attribute Utility Theory	10
2.1.1 Core Concepts of MAUT	10
2.1.2 Defining Alternatives and Utility Functions	13
2.1.2.1 Additive utility function	14
2.1.2.2 Weighted average utility function	14
2.2 Preferences in Artificial Intelligence	17
2.2.1 Acquisition of Preference Information	18
2.2.2 Preferences under Certainty	20
2.2.3 Pareto Dominance	21
2.2.4 AI-Driven Recommendation Systems	22
2.2.5 Comprehensive Elicitation of the Utility Function	24
2.2.6 Interactive Optimization	26
2.2.7 Bayesian Preference Elicitation	28
2.2.8 Minimizing the Error Function	31
2.3 Parameterized Preferences	33
2.3.1 Bayesian vs. Qualitative Approaches in Parameterized Preference Models	34
2.3.2 Imprecisely Specified Multi-Attribute Utility Theory and Minimax Regret Criterion	35
2.3.2.1 ISMAUT	36
2.3.2.2 MMR	37
2.3.3 Possibly Optimal Alternatives	40
2.4 Preference Learning vs. Preference Elicitation	42
2.4.1 Preference Learning Methods	43
2.4.2 Relationship to Preference Elicitation	45
2.5 Reflections and Motivation for the Proposed Approach	46
3 Non-Bayesian Preference Elicitation Under Noisy Models	47
3.1 Introduction	48
3.2 Problem Setting	50
3.3 The Core Concept of Our Approach	54
3.4 Query Selection	62
3.5 Experimental Results	65

4	Conclusions & Discussion	73
4.1	Summary of Contributions	74
4.2	Discussion and Future Research Directions	75
4.3	Limitations of the Work	76
4.4	Potential Impact on the Field	78
4.5	Final Remarks	79

List of Tables

2.1	A multi-criteria decision-making evaluation for smartphones . .	16
3.1	Experimental results w.r.t. the number of criteria p , for Method 1: Max Preference Points for Query Selection with $ A = 1000$, $\rho = 0.1$ and $ \mathcal{W}' = 4000$	67
3.2	Experimental results for Method 2: Best Split for Query Selection ; $ A = 1000$, $\rho = 0.1$ and $ \mathcal{W}' = 4000$	67
3.3	Experimental results for regret-based elicitation with the current solution query strategy; $ A = 1000$ and $\rho = 0.1$	68
3.4	Experimental results for Bayesian elicitation with the queries generated with greedy maximisation of value of information; $ A = 1000$ and $\rho = 0.1$	69
3.5	Experimental results w.r.t. the number $ \mathcal{W}' $ of user preference models, for Method 1 with $ A = 1000$, $\rho = 0.1$ and $p = 5$	71

List of Plots

3.1	Accuracy vs Number of Criteria p across all methods.	69
3.2	Iteration Time vs Number of Criteria p across all methods. . . .	70
3.3	Number of Queries vs Number of Criteria p across all methods.	70
3.4	Average preprocessing time for Method 1 over 100 experiments with $ A = 1000$ and $p = 5$	72
3.5	Accuracy varying with user noise ρ over 100 experiments with $ A = 1000$, $ \mathcal{W}' = 10000$ and $p = 4$	72

I, Samira Pourkhajouei, certify that this thesis is my own work and has not been submitted for another degree at University College Cork or elsewhere.

Samira Pourkhajouei

Abstract

Interactive preference elicitation plays a crucial role in multi-criteria decision-making (MCDM) and recommender systems, particularly when user preferences are uncertain or noisy. Traditional Bayesian approaches offer a principled way to handle such uncertainty by representing preferences as probabilistic models, updating them iteratively based on user responses. However, these methods often entail high computational costs and require prior assumptions on preference distributions, making them less feasible for real-time applications.

This thesis introduces a non-Bayesian framework for interactive preference elicitation under noisy user responses, offering a computationally efficient alternative to probabilistic approaches. Instead of relying on prior distributions, our method identifies and refines a set of plausible preference models by evaluating their consistency with observed user responses. The framework incrementally selects targeted queries that maximize information gain while remaining robust to response inconsistencies, allowing for an effective trade-off between query efficiency and recommendation accuracy.

We propose a novel query selection strategy that prioritizes alternatives based on their potential optimality across multiple preference models, reducing the number of interactions required to identify the user’s most preferred alternative. To ensure scalability, our method operates on a finite approximation of the preference space, allowing for efficient real-time decision-making.

Empirical evaluations demonstrate the effectiveness of the proposed approach, showing that it achieves high accuracy in identifying user preferences while significantly reducing computational costs compared to Bayesian and minimax regret-based methods. The results suggest that this non-Bayesian framework is well-suited for interactive decision-support systems, personalized recommendations, and real-time MCDM applications, offering a practical balance between efficiency, adaptability, and robustness to noise.

To my mother and father, with deepest love and gratitude.

Acknowledgements

I am deeply grateful to my supervisor, **Nic Wilson**, whose unwavering support, patience, and insightful guidance have been invaluable throughout this journey. His expertise, dedication, and support have not only shaped this thesis but have also greatly impacted my professional development. I feel truly privileged to have had his supervision, and I cannot thank him enough for his generosity with his time, thoughtful advice, and continuous support.

My heartfelt thanks go to my parents, Zahra and Hassan, and my siblings, Kaveh and Sarah, whose love, support, and strength have continually inspired and motivated me.

I sincerely thank Paolo Viappiani, Federico Toffano, and Guillaume Escamocher for their enriching collaborations, which contributed significantly to my scientific publications. I also thank the examiners of this thesis, Derek Bridge and Patrick Mannion, for their valuable feedback.

This research was made possible by the financial support of Insight Centre for Data Analytics and Science Foundation Ireland (Grant No. 12/RC/2289-P2) which I am truly grateful.

Finally, special thanks to Caitríona Walsh and Linda O’Sullivan for their invaluable administrative assistance.

Chapter 1

Introduction

1.1 Multi-criteria Decision-making

The rise of artificial intelligence (AI) in recent years has transformed how products, services, and content are recommended, leading to an increased reliance on intelligent systems to make decisions on behalf of users. AI-based recommendation systems, from product suggestions to streaming content, require a robust understanding of user preferences to ensure their recommendations align with individual user needs and values. However, user preferences are often complex, multidimensional, and not immediately apparent, necessitating efficient mechanisms for preference elicitation — the process of gathering information to model and understand a user's preferences [RRS11, Agg16, AT05, LWM⁺15, BZM12].

The field of multi-criteria decision-making (MCDM), or multi-criteria decision analysis (MCDA) [DFGZ19, GEF16, GSH13, IN13, ZD16, KWZ11, GEF10, BMP⁺06, NM21, Sav51, Fin72, Rai68], addresses decision-making challenges where choices must be evaluated based on several, often conflicting, criteria. This approach is crucial in domains where alternatives cannot be ranked or chosen based on a single attribute. Instead, decision-makers (DMs) are tasked with balancing multiple dimensions—such as cost, quality, efficiency, and environmental impact—that may pull the decision in different directions. MCDM provides the framework and methodologies to navigate these trade-offs, making it a foundational element in decision science, economics, engineering, and artificial intelligence.

In a typical MCDM scenario, it is rarely clear-cut which alternative is "better"

overall. Consider two alternatives, α and β , evaluated on the criteria of cost and quality: α may be lower in cost, while β may provide higher quality. In this situation, the optimal choice is not inherently obvious but rather depends on the DM's personal or organizational preferences—whether cost or quality holds more value in the decision context. The subjective nature of such trade-offs introduces complexity, as different decision-makers may prioritize these criteria differently, even within the same context. This complexity means that, in MCDM contexts, the concept of an absolute "optimal" choice is often replaced by the notion of non-dominated or Pareto-optimal alternatives. An alternative is considered Pareto optimal if it is not inferior to any other option across all criteria; it may perform worse in one criterion but excels in another, making it a viable choice depending on the DM's priorities.

However, the set of non-dominated or Pareto-optimal alternatives can often be vast and complex, particularly in cases with numerous criteria or a large decision space. This volume of choices can overwhelm decision-makers, leading to difficulties in processing and evaluating each alternative effectively. MCDM methodologies are designed to address this challenge by filtering, ranking, or sorting alternatives to streamline the decision process. By applying these methods, DMs can focus on a subset of viable options that align with their preferences, reducing cognitive load and enhancing decision-making efficiency.

1.2 Preference Elicitation

A core element of MCDM, particularly in AI-driven recommendation systems, is preference elicitation—the process of determining and refining a user's preferences to provide tailored recommendations. With the proliferation of AI in domains such as product recommendation, entertainment, and service selection, preference elicitation has become essential for systems that operate on behalf of users. By accurately understanding user preferences, these systems can deliver recommendations that better reflect individual values and priorities. Incremental preference elicitation has proven especially effective in AI applications because it allows for gradual learning of preferences rather than relying on pre-specified, fixed criteria (as in classical decision theory) [Bou13]. Incremental elicitation adapts dynamically, where the system generates targeted queries that provide maximal information based on the user's responses so far.

In incremental elicitation, the system engages the user in a series of queries de-

signed to clarify preferences, typically through pairwise comparisons between alternatives. For instance, a user may be asked to choose between two options with differing trade-offs, such as a low-cost option versus a high-quality one. Each response from the user helps refine the system’s model of their preferences, incrementally reducing the uncertainty surrounding their utility function or preference model. This iterative process continues until the system can confidently recommend an optimal or near-optimal choice or until a termination criterion is reached—whether due to low remaining uncertainty, a cap on the number of queries to prevent user fatigue, or an acceptable level of expected loss.

In recent AI-driven research on MCDM, two predominant approaches have emerged for managing the uncertainty of user preferences: Bayesian methods, which utilize probabilistic modeling and iterative updates to infer preference structures [VB20, BPS20], and non-Bayesian methods, which rely on deterministic or qualitative frameworks such as minimax regret [BP18, BGPV15] or constraint-based elicitation [BB06]. While these approaches are widely studied in AI-based decision support, other methods, including constructive preference elicitation [DTP18], robust ordinal regression [GMS14], and support vector machine-based preference learning [MW16], have also been explored in broader MCDM contexts.

Bayesian approaches to preference elicitation represent the uncertainty about the user’s preference model through a probabilistic framework, where a prior distribution over possible preference models is updated based on the user’s responses [Sav72, BPS20]. This probabilistic updating allows Bayesian methods to incorporate the possibility of errors or noise in user responses, offering robustness in cases where users may occasionally provide inconsistent or non-optimal answers. For example, Viappiani and Boutilier [VB20] propose a Bayesian approach that can adaptively choose queries to maximize information gain, handling noisy responses in multi-criteria settings. Despite these advantages, Bayesian methods are computationally demanding, especially in real-time applications, because updating the probability distribution and computing optimal recommendations after each user response is resource-intensive. Moreover, Bayesian methods require a prior assumption about the distribution of user preferences, which may not be feasible or realistic in all cases.

In light of these limitations, non-Bayesian approaches have gained attention for their potential to provide more computationally efficient solutions while

still accommodating preference uncertainty. Non-Bayesian methods avoid probabilistic updates and instead focus on feasible preference models by employing constraints on the parameter space. For instance, minimax regret methods [Bou13] are designed to reduce the DM’s regret by narrowing the parameter space as the user answers more queries. Although minimax regret offers a robust, efficient alternative, it lacks the flexibility to handle noisy responses because strict constraints can exclude feasible models, leading to suboptimal recommendations when errors in user input occur.

In this thesis, we explore a non-Bayesian, qualitative approach to preference elicitation that is designed to handle noisy responses in a multi-criteria decision-making setting. Our method seeks to identify the most plausible preference models for a user by examining the consistency of their responses, rather than assuming a prior distribution over preferences. This allows the system to incrementally filter out models that do not align with the user’s input while focusing on those that most plausibly represent the user’s underlying preferences. By focusing on the most consistent user models, the proposed approach avoids the high computational costs associated with Bayesian updates and probabilistic modeling. This qualitative approach is particularly suitable for real-time decision support and recommendation systems, where computational efficiency and responsiveness are critical.

Our method leverages incremental preference elicitation by engaging the user in a series of simple binary queries—“Do you prefer alternative α or β ?”—and translating each response into constraints on the set of possible preference models. This constraint-based approach allows the system to refine the space of feasible models without requiring exhaustive querying or complex probabilistic updates. Furthermore, we introduce a mechanism to account for potential noise in user responses, ensuring that occasional inconsistencies do not drastically reduce the accuracy of the recommendations. This focus on robustness to noisy responses ensures that the system can continue to provide high-quality recommendations even if the user’s answers are occasionally incorrect or inconsistent.

The proposed approach has been validated through experimental results demonstrating its effectiveness in identifying preferred alternatives with high accuracy and minimal computational demand. This thesis contributes a scalable, non-Bayesian framework for preference elicitation, expanding the toolkit available for AI-based decision support in MCDM contexts. By combining

computational efficiency, robustness to noise, and adaptability, the approach developed here offers a viable alternative to Bayesian methods, particularly in applications requiring real-time responsiveness and minimal user burden.

In summary, this research addresses the limitations of existing MCDM and preference elicitation techniques by introducing a qualitative, incremental approach that emphasizes efficiency and robustness. This non-Bayesian framework for preference elicitation provides an accessible and practical solution for real-time decision-making systems, offering an essential balance between accuracy, user adaptability, and computational feasibility.

1.3 Outline and Contributions

This thesis is structured to systematically build a comprehensive understanding of the proposed framework, starting with foundational concepts and culminating in experimental validation and theoretical insights. Below is an outline of the thesis, emphasizing the contents and purpose of each chapter.

Chapter 2: Background and Overview of Preference Elicitation Methods

This chapter lays the groundwork by providing a thorough exploration of the fundamental concepts underpinning preference elicitation and decision-making frameworks. We begin by introducing key ideas in Multi-Attribute Utility Theory (MAUT), such as multi-attribute utility functions, Pareto dominance, and preferential independence. These foundational principles serve as the theoretical basis for understanding how preferences are represented and evaluated in multi-criteria decision-making contexts.

The chapter then transitions to a discussion of related work, highlighting prominent approaches to preference elicitation. Special attention is given to methodologies that address uncertainty in preferences, including the minimax regret criterion. This criterion is particularly notable for its utility in decision-making under incomplete or ambiguous preference information. By minimizing the maximum possible regret associated with a decision, this approach provides a robust framework for evaluating alternatives in uncertain environments.

Further, we explore the formalization of preference relations through parameterized utility functions. This includes an analysis of how these functions can be used to represent user preferences in a structured and scalable manner. Additionally, we introduce standard optimality classes for sets of alternatives, such

as Pareto-optimal sets and possibly optimal sets, which are frequently used to narrow down choices in multi-criteria decision problems.

This chapter not only provides the theoretical context for the research presented in this thesis but also identifies gaps in existing methodologies, which the proposed framework seeks to address. By thoroughly analyzing prior work and defining the key concepts that underpin preference elicitation, this chapter sets the stage for the methodological and experimental contributions in subsequent chapters.

Chapter 3: Non-Bayesian Preference Elicitation Under Noisy Models

This chapter introduces a novel, non-Bayesian, incremental preference elicitation technique specifically designed to address challenges in scenarios involving noisy and potentially inconsistent user responses. In practical decision-making environments, users may occasionally provide inaccurate or contradictory answers to queries, either due to misunderstandings, fatigue, or inherent uncertainty in their preferences. Traditional elicitation techniques often struggle to balance robustness against such noise with efficiency in computation and query interaction.

The approach detailed in this chapter leverages a qualitative representation of uncertainty to model user preferences. Unlike Bayesian methods, which rely on probabilistic updates and prior distributions, our technique focuses on identifying and maintaining a set of the most plausible user models. These models are defined as those that align most consistently with the decision-maker's responses, providing a robust framework for managing inaccuracies without the computational overhead of Bayesian frameworks.

Our method is guided by the following key objectives:

- **Robustness to Noisy Inputs:** The approach is designed to tolerate occasional inconsistencies in user responses, ensuring that the decision-making process is not derailed by individual errors.
- **Efficiency in Query Interaction:** Minimizing the number of queries required to arrive at an optimal or near-optimal decision is a central goal, reducing the cognitive load on users and expediting the elicitation process.
- **Computational Feasibility:** Generating and analyzing queries must remain computationally efficient, making the approach suitable for real-

time or large-scale applications.

This chapter provides a detailed exposition of the proposed methodology, covering the following key aspects:

- **Query Generation:** The strategies employed to generate queries that maximize informational gain while maintaining computational simplicity. These queries are designed to refine the model incrementally by narrowing the space of plausible user preferences.
- **Stopping Criterion:** A formalized mechanism to determine when the elicitation process can be terminated. This is based on predefined thresholds of uncertainty reduction or practical limits, such as the number of queries or computational resources.
- **Experimental Results:** The performance of the proposed approach is evaluated through computational experiments. These results illustrate its robustness, efficiency, and scalability across a variety of decision-making scenarios.
- **Comparison with Bayesian Methods:** A comparative analysis highlights the strengths and trade-offs of the non-Bayesian approach relative to Bayesian frameworks. While Bayesian methods offer theoretical advantages in modeling uncertainty, their reliance on prior distributions and computational intensity can limit their applicability. In contrast, the proposed non-Bayesian method achieves comparable robustness and accuracy with reduced complexity and faster execution times.

By the end of this chapter, the reader will have a comprehensive understanding of the proposed preference elicitation method, its theoretical underpinnings, and its practical implications. The experimental validation demonstrates the approach's effectiveness and its potential to significantly advance preference modeling in decision-support systems. This contribution aligns with the broader goal of making AI-driven decision-making more accessible, efficient, and resilient to real-world uncertainties.

Chapter 4: Conclusions & Discussion

This chapter provides a concise summary of the research conducted in this thesis, emphasizing the key findings, contributions, and implications of the work. It reflects on the significant results achieved and outlines potential directions for future research to build upon the foundation established here.

1.4 Publications

Chapter 3 draws upon the research findings and methodologies presented in the following paper.

1. Samira Pourkhajouei, Federico Toffano, Paolo Viappiani, and Nic Wilson. An efficient non-bayesian approach for interactive preference elicitation under noisy preference models. In Zied Bouraoui and Srdjan Vesic, editors, *Symbolic and Quantitative Approaches to Reasoning with Uncertainty - 17th European Conference, ECSQARU 2023, Arras, France, September 19-22, 2023, Proceedings*, volume 14294 of *Lecture Notes in Computer Science*, pages 308–321. Springer, 2023. https://doi.org/10.1007/978-3-031-45608-4_23.

Aligned with this thesis, the research received the "Best Student Paper Award" at ECSQARU 2023. The award decision, made by the program chairs, was based on the high quality of the paper and the positive feedback received during the review process.

Additionally, the conference organisers invited the authors to submit an extended and revised version of the ECSQARU 2023 paper for publication in a special issue of the *International Journal of Approximate Reasoning (IJAR)*. After the peer-review process, this work led to the following published paper in IJAR:

2. Guillaume Escamocher, Samira Pourkhajouei, Federico Toffano, Paolo Viappiani, and Nic Wilson. Interactive preference elicitation under noisy preference models: An efficient non-bayesian approach. *International Journal of Approximate Reasoning*, 178:109333, 2025. <https://doi.org/10.1016/j.ijar.2024.109333>.

Chapter 2

Background and Overview of Preference Elicitation Methods

In this chapter, we lay the foundational groundwork for this thesis by exploring the essential concepts and methods related to preference elicitation in multi-criteria decision-making (MCDM). We begin by introducing multi-attribute utility theory (MAUT) and its core concepts, including additive and weighted utility functions for modeling user preferences.

We then survey traditional and modern preference elicitation techniques, including comprehensive utility function elicitation, interactive optimization, and Bayesian methods, each of which addresses preference uncertainty in different ways. Special attention is given to constraint-based approaches, such as the minimax regret criterion, which are particularly relevant to this thesis.

The chapter further examines parameterized utility models, such as ISMAUT, and explores the computation of possibly optimal alternatives and robust decision rules. We also contrast preference elicitation with preference learning, situating our proposed method within the broader landscape of AI-driven decision support systems.

This background provides the theoretical context for the non-Bayesian framework proposed in later chapters and sets the stage for the methodological contributions that follow.

2.1 Multi-Attribute Utility Theory

Multi-criteria decision-making (MCDM), also known as multi-criteria decision analysis, is a field dedicated to aiding decision-makers (DMs) in contexts where alternatives are evaluated based on various and often conflicting criteria. MCDM methods assist in organizing and synthesizing complex and diverse information, ultimately helping DMs analyze and evaluate different alternatives and their impacts on various aspects of the decision-making process. This field has been widely studied and applied across numerous academic disciplines, including finance, statistics, telecommunications, and economics (see, e.g., Zavadskas et al., [ZTK14]). Foundational works in MCDM include those by von Neumann and Morgenstern [NM21], Savage [Sav51], Fishburn [Fis79, WDF⁺08], and Raiffa [Rai68].

In this thesis, we focus on a specific branch of MCDM known as multi-attribute utility theory (MAUT) [Rai68]. It is designed to quantitatively model a decision-maker's preferences when confronted with several, often conflicting, criteria. By offering a structured framework for evaluating and comparing alternatives, MAUT transforms subjective preferences into objective numerical representations, thereby facilitating more systematic and transparent decision-making processes.

2.1.1 Core Concepts of MAUT

Understanding the foundational principles of MAUT is crucial for appreciating how this theory facilitates multi-criteria decision-making. Below, we outline the key concepts that underpin MAUT, providing a structured approach to modeling and analyzing preferences:

- Quantitative Modeling of Preferences

MAUT seeks to convert qualitative preferences into a quantitative form. This transformation involves defining a utility function that assigns a numerical value to each alternative based on its performance across various criteria. The utility function encapsulates the decision-maker's preferences, allowing for a direct comparison of alternatives on a common scale.

- Criteria and Alternatives: In any decision-making scenario, alternatives (choices or options) are assessed based on several criteria

(attributes or dimensions). For example, in choosing a car, criteria might include cost, fuel efficiency, safety, and comfort. Each criterion represents a dimension of evaluation, and each alternative is evaluated on these dimensions, producing a multi-dimensional profile for each alternative.

- Utility Function: The utility function in MAUT is a mathematical representation that aggregates the performance of an alternative across all criteria into a single utility score. This function is designed to reflect the decision-maker's relative preferences for different levels of performance on each criterion. Utility functions can take various forms, but a common approach is the additive utility function, where the overall utility is the weighted sum of the utilities for each criterion. The weights relate to the relative importance of each criterion to the decision-maker.
- Structured Decision-Making Framework

MAUT provides a structured framework for decision-making by ensuring that all relevant criteria are systematically considered. This structure helps in:

- Clarifying Preferences: By requiring explicit specification of preferences and trade-offs among criteria, MAUT helps decision-makers clarify what is most important to them. This process involves eliciting the decision-maker's preferences through various techniques, such as direct rating, pairwise comparisons, or conjoint analysis.
- Consistency and Transparency: MAUT promotes consistency in decision-making by using a formalized approach to evaluate alternatives. This consistency is crucial in scenarios where decisions need to be justified or defended. The transparency of the process is enhanced as the rationale behind each decision is clear, based on the numerical representation of preferences.
- Handling Conflicting Criteria: One of the significant strengths of MAUT is its ability to handle conflicting criteria. By quantitatively modeling how much the decision-maker values improvements in one criterion relative to sacrifices in another, MAUT provides a clear framework for resolving trade-offs. This is particularly useful in complex decision-making environments where no single alternative

is best on all criteria.

- Application of MAUT in Decision-Making

MAUT's application spans various domains, offering a robust tool for decision-makers in fields such as finance [DFGZ19], healthcare [GEF16], engineering [ZTK14], and public policy [CKM⁺22]. The method provides a structured and transparent way to assess trade-offs among multiple criteria, ensuring rational decision-making based on quantitative preference modeling [GEF16, IN13, KR93].

- Financial Decisions: In finance, MAUT can be used to evaluate investment options by considering criteria such as risk, return, liquidity, and ethical considerations. Investors can prioritize these criteria based on their preferences and use MAUT to identify the optimal investment portfolio [DFGZ19, KWZ11]. The method is particularly useful in sustainable finance, where decision-makers must weigh financial performance against social and environmental factors [Mun07].
- Healthcare: In healthcare, MAUT aids in treatment decision-making by evaluating treatment options across criteria such as effectiveness, side effects, cost, and patient quality of life. This helps patients and healthcare providers make informed decisions tailored to the patient's values and preferences [Dol10, GEF16, MK16]. Applications of MAUT in medical decision-making include prioritizing treatments for chronic diseases and designing personalized health interventions [TFL⁺09].
- Engineering: Engineers use MAUT to evaluate design alternatives, taking into account criteria such as cost, performance, reliability, and environmental impact. This structured approach ensures that all critical factors are considered in the design process [ZTK14, CKC05, MA04]. It has been widely applied in areas such as sustainable product design, transportation infrastructure planning, and energy systems optimization [KWZ11].
- Public Policy: Policymakers use MAUT to evaluate policy options, considering criteria like economic impact, social equity, environmental sustainability, and political feasibility. This helps in crafting policies that balance multiple objectives and stakeholder interests

[CKM⁺22, ZGD⁺15].

- Limitations of MAUT

While MAUT offers a robust framework for structured decision-making, it may not be appropriate in situations where criteria interact in complex ways that violate the assumption of preferential independence. For example:

- Non-Additive Interactions: In cases where the value of one criterion depends on the level of another (e.g., a product’s price might affect the perceived importance of its quality), the additive structure of utility functions in MAUT cannot accurately capture these dependencies.
- Synergistic or Antagonistic Effects: Some decision contexts involve synergies (where the combined value of criteria exceeds the sum of their individual contributions) or antagonistic effects (where the value diminishes when considered together). MAUT may struggle to represent such interactions effectively.
- Ambiguous or Imprecise Preferences: If the decision-maker cannot articulate clear preferences or if preferences change dynamically during the process, MAUT’s reliance on explicit trade-offs may pose challenges.

2.1.2 Defining Alternatives and Utility Functions

In this section, we introduce the formal framework for defining alternatives and constructing utility functions within the context of a specialized, non-Bayesian adaptation of MAUT. Unlike traditional MAUT, which often relies on probabilistic models and expected utility maximization, our approach incrementally elicits user preferences while maintaining robustness to noisy responses. By systematically outlining the structure of alternatives and detailing the mathematical formulation of utility functions, we aim to provide a clear and precise methodology for evaluating and comparing decision options. This formalization is essential for capturing the decision-maker’s preferences and facilitating informed, objective decision-making based on multiple criteria.

Let $\alpha \in A$ represent an *alternative* for the decision-maker, with A being a potentially infinite set of alternatives. In a MAUT framework, each alternative is assessed using p *criteria*, or *evaluators*, denoted as $u_i : A \rightarrow \mathbb{R}$ for all

$i \in [1, p]$. Consequently, each alternative α is associated with a vector $u(\alpha) = (u_1(\alpha), \dots, u_p(\alpha))$, referred to as the *outcome* of α , where each u_i represents the value of the alternative α with respect to criterion (or objective) i .

2.1.2.1 Additive utility function

The criteria $u_1, \dots, u_p$ used to evaluate an alternative $\alpha \in A$ are considered mutually preferentially independent if there exists an additive ordinal utility function that can be expressed as:

$$u(\alpha) = \sum_{i=1}^p x_i(u_i(\alpha)) \quad (2.1)$$

In this equation, x_i represents the single-criterion *subutility function* defined for criterion u_i [DH83, Fin72]. *Additive independence* is a significant concept in multi-criteria decision analysis, as it simplifies the utility function's structure by ensuring that the overall utility can be decomposed into the sum of the utilities for each criterion. This property allows decision-makers to assess each criterion separately, making the evaluation process more manageable and intuitive. Consequently, additive independence is frequently assumed in multi-criteria optimization problems due to its practical benefits and the ease it offers in modeling complex decision scenarios.

2.1.2.2 Weighted average utility function

The weighted average model is a prevalent approach for aggregating multi-criteria outcomes of alternatives. This model is extensively utilized in preference elicitation contexts, where it facilitates the combination of different criteria into a single, comprehensive evaluation score. It has been widely adopted in classical studies such as [Sar78, LPK02]. The model's simplicity and effectiveness have also made it a popular choice in more recent research, including works [KS14, KVVA17, BL19].

We assume that the user has a utility function $u : A \rightarrow \mathbb{R}$, which is parameterized by some parameter vector $w \in \mathcal{W}$, where $\mathcal{W}$ is the space of user preferences defined as the set of all the normalised non-negative weights vectors w ,

$$\{w \in \mathbb{R}^p : \sum_{i=1}^p w(i) = 1; w(i) \geq 0; \forall i = 1, \dots, p\} \quad (2.2)$$

We assume that the decision-maker's utility function for evaluating alternatives is a weighted average of the criteria vector, where each weight $w_i \in \mathcal{W}$ quantifies the relative importance of the criterion u_i for the decision-maker. These weights are critical as they reflect the priority or significance the decision-maker places on each criterion when evaluating alternatives. The overall value of an alternative $\alpha \in A$ is determined by aggregating its performance across all criteria, with each criterion assigned a weight that reflects its relative contribution to the decision-maker's utility function.

To be more precise, let the outcome of an alternative α be represented as $u(\alpha) = (u_1(\alpha), \dots, u_p(\alpha))$ where $u_i(\alpha)$ denotes the performance of α on the i -th criterion. The value of α is then calculated as follows:

$$u(\alpha, w) = w \cdot u(\alpha) = \sum_{i=1}^p w_i u_i(\alpha) \quad (2.3)$$

where $\cdot$ is the dot product. The assignment of weights w_i is a crucial step in the decision-making process, as these weights directly influence the ranking and selection of alternatives. The process of determining appropriate weights can involve various methods, such as direct assignment, pairwise comparisons, or more sophisticated techniques like Analytic Hierarchy Process (AHP) or conjoint analysis. These methods help ensure that the weights accurately reflect the DM's preferences and priorities [Saa80, Yue14].

As an example consider a scenario where a decision-maker is evaluating different smartphone models based on three criteria:

- Battery life (u_1): Higher is better (measured in hours).
- Performance (u_2): Higher is better (measured as a score from 0 to 10).
- price (u_3): Lower is better (measured in euros, and normalized as $-cost/100$, to reflect that lower cost increases utility).

The decision-maker assigns weights as follows, reflecting their priorities:

- $w_1 = 0.4$ for battery life,
- $w_2 = 0.4$ for performance, and
- $w_3 = 0.2$ for price (as cost is relatively less important than functionality for these particular units for the criteria).

Table 2.1: A multi-criteria decision-making evaluation for smartphones

Smartphone	Battery Life	Performance	Price
A	10	8	-10
B	12	7	-8
C	8	9	-12

We illustrate this with a concrete example. Table 2.1 presents a multi-criteria evaluation of three smartphones, assessed based on battery life, performance, and price. The utility function is computed using the weighted average model:

$$u(\alpha, w) = w_1 u_1(\alpha) + w_2 u_2(\alpha) + w_3 u_3(\alpha)$$

Then the overall value of each smartphone would be calculated as:

$$u(A, w) = (0.4 \times 10) + (0.4 \times 8) + (0.2 \times -10) = 5.5$$

$$u(B, w) = (0.4 \times 12) + (0.4 \times 7) + (0.2 \times -8) = 6.0$$

$$u(C, w) = (0.4 \times 8) + (0.4 \times 9) + (0.2 \times -12) = 4.4$$

Smartphone B is the optimal choice, with the highest utility of 6.0. This reflects a good balance of battery life, performance, and price, which aligns well with the decision-maker's priorities.

This example illustrates how the weighted average model integrates various criteria, allowing the DM to evaluate and compare alternatives systematically based on their aggregated scores. The weighted average model's simplicity and intuitiveness make it a widely adopted approach in multi-criteria decision-making. It facilitates transparent and rational decision processes, ensuring that all relevant criteria are considered according to their importance [KR93, Vin92]. This method is particularly beneficial in contexts where decisions need to be justified or communicated clearly, as the rationale behind each decision can be easily understood and explained.

Furthermore, the model's flexibility allows for adjustments in weights to reflect changing preferences or priorities over time. This adaptability is essential

in dynamic environments where the decision context may evolve, requiring periodic reassessment of criteria importance.

By quantifying the relative importance of each criterion through weights, the weighted average model empowers decision-makers to make well-informed, balanced decisions that align with their goals and values.

2.2 Preferences in Artificial Intelligence

Preferences play a significant role in various contexts within the field of artificial intelligence (AI). They can encompass multiple agents, each representing real users or persons, or involve multiple criteria over which preferences are expressed. The complexity arises when preferences of multiple agents conflict or when preferences across different criteria oppose each other. Determining how to make decisions or identify "optimal" solutions in such scenarios is a challenging task [SLB08].

In multi-agent systems, preferences from different agents often conflict, reflecting the diverse and sometimes opposing interests of the individuals involved [Woo09]. Similarly, in multi-criteria decision-making, preferences across various criteria can be contradictory, making it difficult to balance the trade-offs required to achieve a satisfactory decision [KR93].

This dissertation focuses on single-agent decision-making. However, it acknowledges that the different criteria used to evaluate alternatives can sometimes be viewed as analogous to the preferences of multiple agents. This perspective allows for the application of methods typically used in voting scenarios to handle the preferences across multiple criteria [Arr12].

Uncertainty in handling preferences in AI can stem from several sources. One major source is the lack of complete information about user preferences. Users may have implicit preferences that are not easily articulated or may change over time [RRS11]. Another source of uncertainty is the variability in the features of the alternatives themselves. For instance, the actual flight time for different flight options can vary due to factors like delays, adding an element of unpredictability to the decision-making process.

Preferences in AI are not static and can evolve based on new information or changing circumstances. This dynamic nature of preferences necessitates adaptive decision-making models that can update and refine preference representa-

tions as more information becomes available [AT05].

Furthermore, AI systems must often deal with conflicting preferences both within a single agent's criteria and across multiple agents. For example, in a recommender system, a user's preference for both low cost and high quality might conflict, requiring the system to find a balance that best satisfies both criteria. In multi-agent contexts, collaborative filtering methods [HKBR17, ERK11, RRS15, BZM12, ERR13, LHZ14] leverage the preferences of similar users to infer preferences for a target user. These systems do not seek consensus across all users, but rather use patterns in user-item interactions to make personalized recommendations. For instance, if users with similar rating histories liked a particular item, it is likely to be recommended to the target user, even if it is not broadly popular.

Incorporating preferences into AI systems also involves sophisticated techniques from fields such as optimization, machine learning, and game theory. These techniques help in formulating the problem, modeling preferences, and finding solutions that best meet the specified criteria [Mit21]. For example, preference learning algorithms can infer user preferences from observed behaviors, while optimization methods can find solutions that maximize the overall satisfaction of these inferred preferences [FH10].

Additionally, ethical considerations play a crucial role in preference handling in AI. Ensuring fairness and avoiding biases are critical, especially when the AI system's decisions significantly impact users' lives. For instance, in automated hiring systems, preferences must be managed to prevent discrimination and promote diversity [Bin18].

2.2.1 Acquisition of Preference Information

To effectively support users in decision-making, it is crucial to understand their preferences to determine which options they consider favorable. This understanding involves comprehending the user's preference relations, which order alternatives according to their desirability. Common methods for acquiring preference information include the following:

Analyzing Historical User Data: One approach is to analyze historical data that explicitly or implicitly indicates user preferences. For example, preferences over travel destinations can be extracted from location-aware information obtained from location-based social networks, allowing for personalized travel

package recommendations [BZWM15]. Similarly, analyzing browsing behavior and applying text mining techniques can lead to accurate hotel recommendations [LWM⁺15]. However, this approach requires the availability of historical user data and the user's consent to use it, which might not always be feasible. Additionally, it is important to consider the appropriate time frames in the historical data, as user preferences may change over time.

Leveraging Similar User Behavior: One can assume that, under some measure of similarity, users with similar preferences in the past are likely to share similar preferences in the future. This assumption underpins collaborative filtering, a widely used approach in recommender systems [HKBR17, ERK11, RRS15, BZM12, ERR13, LHZ14]. Rather than aiming to satisfy the preferences of a group or a majority, collaborative filtering constructs predictions for an individual user by analyzing the preferences of other users who exhibit similar behavior. These similarities can be based on past ratings, purchases, or other interactions, and are used to identify items that the target user has not yet seen but is likely to appreciate.

In contrast, content-based recommender systems [BKN07] generate recommendations using only the target user's own interaction history. These systems analyze the features or attributes of items, such as genre, keywords, or technical specifications, that the user has liked or consumed in the past, and then suggest similar items. Content-based approaches do not rely on the preferences or behaviors of other users, nor do they require a similarity metric between users. Instead, they construct a model of the user's interests based on item content, making them well-suited for scenarios where user preferences are unique or where data about others is sparse. While less affected by cold-start issues compared to collaborative filtering, content-based systems may suffer from limitations such as over-specialization, where the recommendations lack diversity and primarily suggest items very similar to those the user has already seen.

Asking the User Directly (Interactive Preference Elicitation): Directly querying the user for their preferences is a fundamental approach in interactive preference elicitation. Unlike methods that rely on historical data or user similarity, this approach actively engages users in providing explicit feedback on their preferences. While this method offers the potential for highly accurate preference models, it also requires carefully designed interaction strategies to balance user effort and the informativeness of responses.

This approach is particularly relevant in scenarios where user preferences

are uncertain or noisy, necessitating structured elicitation methods to refine decision models. Chapter 3 explores this in depth, introducing a non-Bayesian preference elicitation framework designed to efficiently gather user preferences while accounting for inconsistencies in responses.

2.2.2 Preferences under Certainty

Consider two alternatives, denoted as α and β , within a set of possible alternatives A . The expression $\alpha \succsim \beta$ indicates that a decision maker (DM) weakly prefers alternative α to alternative β , meaning α is at least as good as β . The weak preference relation $\succsim$ is assumed to be a total preorder, also known as a weak order. This weak order is characterized by two fundamental rationality properties:

Completeness: For any alternatives α and β in A , either $\alpha \succsim \beta$ or $\beta \succsim \alpha$. This means that the decision maker can always express a preference between any two alternatives or state that they are equally preferred.

Transitivity: For any three alternatives α , β , and γ in A , if $\alpha \succsim \beta$ and $\beta \succsim \gamma$, then $\alpha \succsim \gamma$. This property ensures that the decision maker can consistently rank alternatives, allowing for ties.

The weak preference relation can be decomposed into two more specific relations:

Indifference relation: $\alpha \equiv \beta$ if and only if both $\alpha \succsim \beta$ and $\beta \succsim \alpha$. This means the decision maker is indifferent between α and β . The indifference relation $\equiv$ is an equivalence relation, which satisfies reflexivity, symmetry, and transitivity.

Strict preference relation: $\alpha \succ \beta$ if and only if $\alpha \succsim \beta$ and $\beta \not\succsim \alpha$. This indicates that the decision maker strictly prefers α to β . The strict preference relation $\succ$ forms a strict order, characterized by asymmetry and transitivity.

Utility Representation: A utility representation theorem provides necessary and sufficient conditions for representing a weak preference relation $\succsim$ with a utility function. More formally, a weak preference relation over a set of alternatives $\mathcal{U} \subseteq A$ can be represented by an ordinal utility function $u : \mathcal{U} \rightarrow \mathbb{R}$. This function satisfies the following condition for any alternatives $\alpha, \beta \in \mathcal{U}$:

$$u(\alpha) \geq u(\beta) \iff \alpha \succsim \beta \quad (2.4)$$

The utility function u that represents the weak preference relation is called an ordinal utility function. These functions are unique up to a monotonic transformation [BC19], meaning that if two utility functions, u and u' , represent the same preference relation $\succsim$, then there exists a monotonic increasing function h such that $u(\cdot) = h(u'(\cdot))$. This property ensures that ordinal utility functions only establish a ranking over alternatives, without providing information on the magnitude of preference differences.

2.2.3 Pareto Dominance

Pareto dominance is a key concept in decision-making and multi-criteria evaluation. An alternative α is said to Pareto dominate another alternative β if α is at least as good as β in every criterion and strictly better in at least one criterion. This allows for eliminating Pareto-dominated alternatives from the decision process, ensuring that no better alternative is overlooked.

Formally, let α and β be two alternatives, each evaluated according to multiple criteria. Denote their outcomes as $x = (x_1, \dots, x_p)$ and $y = (y_1, \dots, y_p)$, where x_i and y_i are the evaluations of α and β on the i -th criterion, respectively. α Pareto dominates β if and only if:

$$x_i \geq y_i \quad \text{for all } i \quad (2.5)$$

and

$$x_i > y_i \quad \text{for some } i. \quad (2.6)$$

The Pareto dominance relation, denoted as $\succ_P$, is defined such that $\alpha \succ_P \beta$ if and only if α Pareto dominates β . An alternative is considered Pareto optimal if there is no other alternative that Pareto dominates it. The set of all Pareto optimal alternatives is known as the Pareto frontier or efficient set, representing the best possible choices according to the given criteria.

Example 1. Consider three apartments, α , β , and γ , with the following criteria: monthly rent (where lower is better), size, and the number of bedrooms (where higher is better). The outcomes for these alternatives are:

$$\alpha: (1400, 60, 2), \beta: (1200, 60, 2), \gamma: (1400, 70, 2)$$

In this case, apartment α is Pareto dominated by β since β has the same size and number of bedrooms but a lower monthly rent. However, both β and γ are Pareto optimal since β offers lower rent, while γ provides a larger size. Therefore, the decision maker's choice set can be reduced to β and γ , as they represent the non-dominated options.

Monotonic Utility and Pareto Optimality: A utility function $u : \mathbb{R}^p \rightarrow \mathbb{R}$ is said to be monotonic if, for any alternatives α and β , the following condition holds:

$$\alpha \succ_P \beta \implies u(\alpha) \geq u(\beta). \quad (2.7)$$

In preference elicitation, ensuring that the utility function is monotonic guarantees Pareto-optimality, aligning with the preferences of a rational decision maker. This ensures that the final recommendations do not include Pareto-dominated alternatives, which would be suboptimal.

2.2.4 AI-Driven Recommendation Systems

Traditional recommendation approaches, such as collaborative filtering [RRS15, BZM12], content-based filtering [LdGS11], and hybrid methods [Bur02], have been extensively employed to predict user preferences based on historical interactions. While these systems have proven effective, they can face challenges in early-stage recommendation scenarios. In particular, collaborative filtering methods often suffer from the cold-start problem, where recommendations are difficult to generate for new users or new items due to the lack of sufficient interaction data [DNNR18]. Content-based systems are less affected by this issue, as they can rely on item attributes to generate recommendations even with limited user history.

To overcome these limitations, modern AI-driven recommendation systems have incorporated sophisticated machine learning techniques, including deep learning, reinforcement learning, and knowledge graphs, to enhance the adaptability and intelligence of recommendation processes [ZLJ21]. These advancements have given rise to interactive and dynamic recommendation paradigms, where user preferences are actively elicited rather than passively inferred from historical behavior. One of the most significant innovations in this area is Conversational Recommender Systems (CRS) [BGMS05, BKN07, BR07,

SZ18], which utilize natural language interactions to refine recommendations dynamically [GGL18].

Unlike traditional recommender systems that rely on implicit signals such as clicks or ratings, Conversational Recommender Systems (CRS) interact directly with users to gather explicit feedback, enabling a more adaptive and personalized recommendation process [GLH⁺21]. These systems leverage natural language and dialogue management techniques to engage users in meaningful interactions, allowing them to express their preferences, clarify ambiguities, and refine their choices through real-time feedback [SZ18].

A crucial aspect of CRS is their ability to elicit preferences dynamically. Instead of relying solely on past interactions, these systems engage in conversations where users can clarify their choices, helping the system build a more precise understanding of their needs. This interactive approach enhances adaptability, as CRS do not follow rigid protocols but instead determine which queries to present based on prior responses. By selecting questions that are most likely to yield valuable information, these systems optimize both efficiency and user satisfaction [ZZZ⁺20].

Context plays a significant role in conversational recommendation. Modern CRS integrate information such as user intent, previous interactions, and domain knowledge to provide more relevant suggestions. By leveraging contextual cues, these systems can refine their recommendations, ensuring that the content or products offered align closely with the user's immediate needs [LZH⁺20]. This ability to adjust in real-time makes CRS particularly effective in domains where preferences can shift rapidly, such as travel planning, e-commerce, and entertainment platforms [ZCA⁺18].

There are several approaches to implementing CRS. Some systems rely on predefined rules, where structured scripts guide the interaction and response selection. Others employ statistical models that analyze patterns in user data to predict preferences. More advanced CRS leverage artificial intelligence techniques, such as reinforcement learning and large language models, to create highly adaptive and intelligent recommendation frameworks [SZ18]. These AI-driven systems can maintain coherent, multi-turn conversations and continuously learn from user interactions, improving personalization over time.

2.2.5 Comprehensive Elicitation of the Utility Function

The classical method for eliciting a decision-maker's (DM) preferences is rooted in the complete elicitation of their utility function, which is accomplished through a structured series of questions and responses [Fis67, Rai68]. The utility function encapsulates the DM's preferences and serves as a mathematical representation of how they value different outcomes. The goal of this method is to fully define the DM's utility function so that it can be used to guide decision-making processes and rank alternatives accordingly.

The process of utility elicitation follows a systematic approach, which can be summarized in the following key steps:

1. **Identifying the Structure of the Utility Function:** The first step is to define the foundational properties of the utility function. This includes determining the number of decision criteria, the range of possible outcomes (outcome space), and assessing preferential independence among criteria. Preferential independence refers to situations where the DM's preference for one criterion does not depend on the levels of other criteria, simplifying the construction of the utility function.
2. **Comprehensive Parameter Assessment:** Once the basic structure is established, the next step is to determine the specific parameters of the utility function. This is typically done through detailed interviews or questionnaires, where the DM is asked to evaluate different trade-offs, express their preferences, and assign values to different outcomes. The goal is to precisely quantify the DM's preferences in a way that fully specifies the utility function, allowing for clear, quantitative comparisons between options.
3. **Consistency and Sensitivity Analysis:** After the utility function has been formulated, the responses provided by the DM are checked for internal consistency. Inconsistencies may arise when the DM's answers contradict each other, which can indicate confusion or misunderstanding during the elicitation process. If inconsistencies are found, the DM is asked to review and revise their answers. Sensitivity analysis is then performed to examine how changes in inputs (such as slight variations in the DM's preferences) affect the final utility function. This ensures that the utility function is robust and accurately reflects the DM's true preferences.

Once fully defined, the DM's utility function can be employed to generate

optimal recommendations or to rank a set of alternatives that align with the DM's preferences. The utility function serves as a powerful decision-making tool, as it mathematically encapsulates how the DM values different outcomes. However, due to the complexity and effort involved in its construction, this approach is most useful when the utility function can be reused in the long term or applied to a variety of recommendation problems.

Despite its theoretical appeal, this process can be challenging in practice. Research with real users has shown that fully eliciting a utility function can be a cognitively demanding and error-prone task [TK74, PC08]. Humans often struggle with making consistent judgments, particularly when faced with complex decision problems involving trade-offs between multiple criteria. This process can also become prohibitively expensive in terms of the number of questions required to fully specify the utility function, especially in combinatorial domains where the number of potential alternatives grows exponentially with the number of criteria involved.

For instance, in multi-criteria decision-making scenarios, where many factors are considered simultaneously, the number of possible alternatives can increase rapidly. As a result, learning the DM's preferences across every point in the utility function may be impractical. In these cases, alternative approaches such as Bayesian methods for preference elicitation can be employed. Bayesian techniques aim to model the uncertainty in the DM's utility function by representing it as a probability distribution over possible parameters of the value function. Instead of attempting to specify the utility function in full, Bayesian methods incrementally update the model based on observed preferences, thus reducing the cognitive and computational burden on the DM.

In summary, while the complete elicitation of a utility function offers a rigorous and precise way to capture a DM's preferences, it is often challenging to implement in practice, particularly in complex decision environments. Therefore, it may be more appropriate in situations where the utility function can be reused or where decision support needs are long-term. In other contexts, more flexible and scalable methods, such as Bayesian approaches, may offer a more practical solution for preference elicitation.

2.2.6 Interactive Optimization

Interactive optimization is a well-established approach commonly used in multi-criteria decision-making. It enables the decision-maker (DM) to explore Pareto-optimal alternatives through iterative interactions, without needing to evaluate every possible option upfront (see, e.g., [SS78, Kor05, BLL20, VLBB24]). This method allows the DM to gradually refine their preferences by interacting with the system, which uses the information provided to guide the search for optimal solutions.

In interactive optimization, it is typical to adopt a parameterized form of the DM's utility function. This parameterization reflects different preference scenarios, and the constraints applied to these parameters evolve based on the feedback gathered from the DM during the interaction. As the DM provides additional preference information, the system adjusts these parameters, narrowing down the search space for optimal alternatives. Common utility functions employed in this context include linear scalarizing functions, such as the weighted average method (discussed in sub-section 2.1.2.2), and Chebyshev scalarizing functions, which focus on minimizing the distance from an ideal utility vector [Wie80].

Methods that progressively interact with the DM to reduce uncertainty about the utility function are also known as *incremental methods* [BP15, BG15, LB11b]. These methods aim to recommend alternatives without requiring a fully specified utility function from the outset. Instead, they rely on minimizing uncertainty or using decision criteria like the minimax regret criterion (see Section 3.2) to guide the selection of optimal alternatives. In some cases, these incremental methods can guarantee optimal recommendations even without a complete utility function.

Typical Interactive Optimization Process: A typical interactive optimization approach can be broken down into the following steps:

1. **Computation:** The system generates a set of solutions that belong to the Pareto frontier, which represents the set of non-dominated alternatives where no improvement in one criterion can be made without sacrificing performance in another.
2. **Interaction:** The system presents a subset of these Pareto-optimal solutions to the DM, prompting the DM to provide new preference infor-

mation, such as preferences between alternatives or adjustments to their stated criteria.

3. Termination: The process continues iteratively until either the DM decides to stop, or a pre-defined stopping criterion is met, such as a sufficient level of preference clarity or a maximum number of iterations.

Preference Elicitation Techniques: Different strategies have been developed to generate queries for the DM during the interactive process [SR91]. One classic approach involves asking the DM to compare alternatives directly, specifying their preferences between two or more options [ZW76, SC83]. The responses help refine the preference model by reducing uncertainty about the DM's utility function. Another common technique involves asking the DM to define *aspiration levels* or make *interval judgments*.

Aspiration levels serve as reference points that reflect the DM's desired outcomes for specific criteria. For example, the DM might specify that they want a certain minimum performance in one criterion while being more flexible in others [Wie80, LSZ92]. Interval judgments require the DM to make trade-offs between different criteria, expressing preferences in relative terms, such as stating that one criterion is at least twice as important as another [SH92, KM97].

Advantages and Challenges of interactive optimization: Interactive optimization has several advantages. One key benefit is that it does not require the system to explore or consider the entire set of possible alternatives. The information learned through interactions can be reused for new sets of alternatives that may not have been part of the initial elicitation process, making this method especially useful for complex, large-scale decision problems. It is also highly effective in cases where the alternatives are represented by a combinatorial problem, providing a structured way to explore the Pareto frontier without overwhelming the DM with choices.

However, one major drawback of interactive optimization is that it often requires numerous interactions with the DM to identify a solution that fully satisfies their preferences. As the number of queries increases, so does the risk of introducing inconsistencies in the DM's responses, particularly when their true preferences are not well-defined or when the problem is highly complex. The accumulation of inconsistent information can lead to inaccurate

recommendations or misrepresentation of the DM's actual preferences.

To address this challenge, the literature has proposed several methods for managing inconsistent preference data. Common solutions include approaches that use Bayesian models to represent uncertainty in the DM's preferences, allowing the system to handle inconsistencies probabilistically. Alternatively, some methods focus on minimizing an error function that measures deviations between the collected preference data and an idealized model of the DM's utility function. Another option is to start the process with a predefined utility function that is incrementally updated as the DM provides additional feedback, as seen in methods like [DTP18].

Recent research has also explored the use of machine learning techniques for preference elicitation. For instance, [MW16] demonstrated the use of support vector machines to estimate the DM's utility function based on their responses, while [DTP18] developed a method that starts with a pre-configured utility function and refines it iteratively based on the DM's input.

Interactive Optimization Interfaces: Implementing interactive optimization systems requires designing intuitive user interfaces that allow the DM to easily provide feedback and view the results of their input (see, e.g., [Kli92, Bly02, TMHI13]). A well-designed interface can significantly improve the decision-making process by making it easier for the DM to understand the consequences of their preferences and decisions. Furthermore, if inconsistencies in the DM's responses are detected, the interface can notify the DM directly and request clarification, reducing the likelihood of error propagation.

In conclusion, while interactive optimization offers a powerful framework for multi-criteria decision-making, especially in complex or large-scale problems, it requires careful management of DM interactions and the risk of inconsistency. Continued advancements in preference elicitation techniques and user interface design are critical for improving the effectiveness and user experience of interactive optimization systems.

2.2.7 Bayesian Preference Elicitation

Bayesian preference elicitation is a powerful approach that addresses the uncertainty inherent in modeling a decision-maker's (DM) utility function. In this framework, the uncertainty is represented through a prior probability

distribution, which can be estimated using data from other users with similar preferences or historical data. This prior serves as an initial belief about the DM's preferences when no direct input from the DM is available. Based on this prior, the system can recommend alternatives by choosing the one that maximizes the expected utility. This strategy is reasonable when no other specific information about the DM's preferences is yet known.

However, in scenarios where interaction with the DM is possible, the system can improve the accuracy of the utility function through a process of incremental elicitation. This involves asking the DM targeted questions to refine the utility model over time. Each interaction is used to update the belief about the DM's preferences, reducing uncertainty and making future recommendations more aligned with the DM's true utility function (see, e.g., [CKP00, BB06]).

Value of Information in Bayesian Elicitation: A key concept in Bayesian preference elicitation is the *value of information*, which is used to assess the importance of a query. The value of information measures how much a specific query can increase the expected utility of future recommendations. In other words, a query is deemed valuable if the DM's response to it significantly enhances the understanding of their utility function. This means that queries with higher values of information are prioritized, as they are more likely to yield useful insights into the DM's preferences.

Despite its theoretical appeal, calculating the exact value of information for a given query can be computationally prohibitive. This is because it would require considering all possible future queries and responses, which creates an enormous computational burden, particularly in complex decision-making environments (see, e.g., [Bou02]). To address this, a more practical approach often used is a *myopic strategy*. In this strategy, the value of information is calculated based on the immediate impact of a DM's response to a single query, rather than considering all possible future interactions. The expected utility is compared before and after the DM's answer, simplifying the computational demands while still guiding the system to meaningful interactions (see, e.g., [VB10]).

Advantages of Bayesian Preference Elicitation: Bayesian preference elicitation has several distinct advantages, particularly in the context of interactive decision-making. First, it allows for personalized and highly relevant recommendations with a limited number of interactions. Rather than requiring

the DM to provide extensive preference information upfront, the system can learn incrementally, tailoring its recommendations as it gains more insight. This makes it efficient and user-friendly, as it minimizes the cognitive load on the DM while still converging toward an accurate utility model (see, e.g., [GS10, VB10]).

Another strength of the Bayesian approach is its robustness in handling inconsistent or incomplete preference information. Unlike some other elicitation methods that discard or override previous inputs when a conflict arises, Bayesian models treat new information as a refinement of the probability distribution over the DM's preference space. This means that even conflicting responses can be incorporated without excluding possible preference models, providing greater flexibility and resilience in the face of ambiguity or noise in the DM's inputs.

Challenges and Practical Considerations: While the Bayesian framework offers many theoretical benefits, it also presents practical challenges, particularly when it comes to selecting the optimal queries for the DM. Even with the use of myopic strategies, the process of query selection can still be computationally intensive due to the need to continuously update and compute expected utility. To mitigate these challenges, approximate methods are often employed. For example, Monte Carlo techniques can be used to approximate the value of information by simulating different possible outcomes and computing the expected changes in utility (see, e.g., [Via12, VLHB20]). These techniques allow for a more feasible implementation of Bayesian preference elicitation in real-world applications, striking a balance between precision and computational efficiency.

Despite these advances, another challenge lies in establishing a reliable prior probability distribution, especially in cases where little or no relevant data is available. In such situations, it may be difficult to accurately represent the DM's initial preferences, leading to a slower convergence towards an optimal utility model. Moreover, the prior may not always be well-suited for all DMs, particularly when their preferences diverge significantly from the norm or when they are dealing with highly unique decision problems.

Bayesian preference elicitation is a flexible and powerful method for modeling and refining a DM's preferences through iterative interactions. It balances the need for personalized, accurate recommendations with the practical constraints of limited interaction time and computational complexity. By focusing on

the value of information, Bayesian approaches prioritize the most informative queries, allowing systems to rapidly hone in on the DM's true utility function with minimal effort on the part of the user. However, the challenges of query selection and prior distribution estimation mean that practical implementations often rely on approximate methods to ensure efficiency and usability. Despite these challenges, Bayesian preference elicitation remains a widely adopted approach for building robust, interactive decision-support systems capable of handling uncertain and evolving user preferences.

2.2.8 Minimizing the Error Function

In preference elicitation, a common approach involves estimating a decision-maker's (DM) utility function by minimizing an error function based on the DM's input preferences. Various methods in the literature focus on this approach, aiming to fine-tune the parameters of a utility function to best match the given preference information. One notable class of methods following this strategy is known as *Utilité Additive* (UTA) (see, e.g., [SGM⁺05, SGMP18]), where the objective is to construct an additive utility function from a set of input preferences provided by the DM.

Typically, these input preferences could be expressed in various forms, such as a weak ordering of a subset of alternatives or specific information about the relative importance of certain criteria. The goal is to estimate the parameters of a utility function by solving a constrained optimization problem that minimizes the discrepancy (or error) between the observed preferences and the predicted utility values. This is often referred to as minimizing a utility error (see, e.g., [MR05]).

UTA Method and Its Considerations: The UTA method is particularly useful for handling cases where the DM's preferences can be represented in an additive form, where the utility of each alternative is the sum of the utilities across different criteria. By minimizing the error function, the system fine-tunes the utility function so that it aligns with the DM's preferences. However, a challenge with this approach is that the more preference data provided by the DM, the greater the likelihood of inconsistencies arising. These inconsistencies can make it difficult to obtain an accurate utility model unless proper inconsistency management techniques are employed. Without a mechanism to handle inconsistent inputs, the model's robustness may be compromised, leading to

suboptimal recommendations.

Another limitation of this approach is the somewhat arbitrary nature of the error function used for optimization. Common choices include quadratic error (see, e.g., [MR05]) or variance minimization (see, e.g., [Koj07]). The choice of error function can significantly affect the results, yet there is often no clear rationale for selecting one over another, which introduces potential bias in the utility estimation process.

Support Vector Machine (SVM) for Preference Learning: A more modern variation of utility function estimation leverages machine learning techniques, such as the use of support vector machines (SVMs) to model preferences. In the work of [MW16], SVM is applied to estimate a weighted sum utility function, where the weights represent the relative importance of different criteria. SVM is typically used for binary classification, and in this context, it helps in defining a hyperplane in a multi-dimensional space ($\mathbb{R}^\ell$) that separates two classes of preferences, those indicating that one alternative is preferred over another.

In practice, this method involves presenting the DM with binary preference queries such as “Is alternative a better than alternative b?” Each response is mapped to a point in a multi-dimensional space, and the SVM algorithm determines the hyperplane that best divides the positive responses (yes) from the negative ones (no). Once the hyperplane is learned, it can be used to rank a set of alternatives according to the inferred utility function, making it a powerful method for preference learning in an interactive environment.

Robust Ordinal Regression: A more general and flexible approach to utility function estimation is *robust ordinal regression*, which extends the principles of UTA by considering all sets of parameters that are consistent with the DM’s preference information (see, e.g., [GSFM10, GMS14]). Rather than focusing on a single utility function, robust ordinal regression seeks to account for multiple possible utility functions that fit the input preferences, creating a more comprehensive representation of the DM’s uncertainty.

This method generates a set of linear constraints that capture the conditions under which the preference information holds. The outcome is not a precise utility function but rather a family of utility functions, any of which could plausibly reflect the DM’s true preferences. As a result, robust ordinal regression can offer a more flexible and reliable model, particularly when the DM’s

preferences are complex or uncertain. However, a downside is that this approach may require a larger amount of input data to achieve a meaningful approximation of the DM's utility function. The greater the uncertainty in the utility function's structure, the more data is necessary to refine the model and reduce ambiguity.

Advantages and Challenges: The main advantage of these approaches, including UTA and robust ordinal regression, lies in their ability to work with limited initial preference information and iteratively refine the utility model. By minimizing an error function, the system can progressively align its utility estimates with the DM's actual preferences. These methods are particularly useful in decision-making scenarios where the utility function is unknown or difficult to specify upfront.

However, as with any optimization-based approach, there are trade-offs. The quality of the final utility function depends heavily on the choice of the error function and the amount of input data. Too little data may lead to an inaccurate or incomplete utility model, while too much data could introduce inconsistencies that degrade the model's performance. Additionally, the computational complexity of some of these methods, especially those involving SVMs or robust ordinal regression, can pose challenges in real-world applications where decision-making needs to be efficient and scalable.

In conclusion, minimizing an error function is a central approach in preference elicitation, offering a structured way to estimate utility functions based on the DM's preferences. While various methods like UTA, SVM, and robust ordinal regression provide powerful tools for refining utility models, they must be carefully managed to avoid inconsistencies and ensure robustness in practical decision-making settings.

2.3 Parameterized Preferences

In decision-making contexts, user preferences are often uncertain, particularly in multi-criteria or multi-objective problems. This uncertainty stems from the challenge of accurately defining a decision-maker's (DM's) utility function, which is a task prone to error. As a result, modern preference elicitation systems rely on multiple preference models, each parameterized by a set of scenarios, denoted as $\mathcal{U}$. Each scenario $w \in \mathcal{U}$ represents a specific configuration of

the parameters underlying the utility function, yielding a corresponding utility function u_w that can be used to compare alternatives.

Consider a scenario where the set of available alternatives is denoted by A . If the true scenario $w \in \mathcal{U}$ were known, representing the DM's actual preferences, then it would be possible to select the best alternative from A based on the utility function u_w . More precisely, each utility function u_w associated with a scenario $w \in \mathcal{U}$ induces a weak preference relation $\succsim_w$ over the set of alternatives. Thus, a scenario w can be seen as a model of the DM's preferences that aligns with the available preference information. However, due to the uncertainty in the DM's preferences, we often deal with a non-singleton set $\mathcal{U}$ of possible scenarios, each representing a different user preference model.

The main objective of a parameterized preference elicitation system is to reduce the uncertainty regarding the DM's preferences by narrowing down the set $\mathcal{U}$. This set contains the potential parameters that define the utility function. Common utility functions utilized in such elicitation systems include the weighted sum function, the ordered weighted average (OWA) function (e.g., [Yag88]), and the Choquet integral (e.g., [GPS10]).

In the context of social choice, utility functions are often referred to as social choice functions or rank aggregation functions (e.g., [LB11a]). Here, the preferences of individual voters are typically represented as rankings of alternatives, and the goal is to aggregate these preferences into a consensus alternative or ranking. Popular social choice functions include the Plurality Rule and the Borda Count (e.g., [FSST17, LS23]).

Parameterizing preference models provides a way to systematically explore different scenarios, each capturing a potential configuration of the DM's utility function. Preference elicitation techniques aim to progressively reduce the number of feasible scenarios until the system can make a well-informed recommendation. These methods often enable the system to recommend an optimal alternative, even when the DM's utility function is not fully known.

2.3.1 Bayesian vs. Qualitative Approaches in Parameterized Preference Models

There is a significant distinction in the literature regarding how uncertainty in preference models is treated. On one hand, Bayesian approaches assume a probabilistic model over the parameters of the utility function. These methods,

such as those in [CKP00], [Bou02], and [VB10], offer a structured way to handle uncertainty by updating the probability distributions based on new information. A common approach in this framework is to maximize the expected value of information (EVOI), either through a myopic or sequential process. Bayesian approaches are advantageous due to their robustness to inconsistent preference inputs, although they often come with significant computational costs, particularly for complex combinatorial alternatives.

On the other hand, qualitative approaches rely on logical, often rule-based, representations of uncertainty. This line of research includes methods such as those in [BPPS06], [VB09], [BB12], and [MRW13]. These approaches are typically computationally efficient but may suffer from sensitivity to inconsistencies in user responses. An example of this is the minimax regret (MMR) criterion, which ranks alternatives by minimizing the worst-case regret across scenarios. MMR, explored throughout this thesis, is particularly useful when the utility function is only partially known.

2.3.2 Imprecisely Specified Multi-Attribute Utility Theory and Minimax Regret Criterion

One of the earliest methods developed to address uncertainty in user preferences is Imprecisely Specified Multi-Attribute Utility Theory (ISMAUT), which operates based on the weighted sum utility function. ISMAUT allows for a range of potential utility functions to be considered, narrowing down the possible scenarios through incremental preference elicitation.

The minimax regret criterion (MMR) provides a framework for decision-making under utility uncertainty. When a DM's utility function is not precisely known, MMR offers a method for evaluating and ranking alternatives by minimizing the worst regret one might experience if the true utility function were known after making a decision. This criterion is well-suited for situations where the DM is uncertain about their exact preferences but desires to avoid significantly worse outcomes.

In summary, parameterized preference models provide a structured way to manage the inherent uncertainty in decision-making processes. By iteratively narrowing down the set of possible preference models through either Bayesian or qualitative approaches, these models allow for more informed and robust recommendations. In this thesis, we will delve deeper into the specifics of

ISMAUT and MMR as foundational methods for handling such uncertainty.

2.3.2.1 ISMAUT

The Imprecisely Specified Multi-Attribute Utility Theory (ISMAUT), introduced by [ISD84], represents one of the earliest methodologies for dealing with uncertain or incomplete information regarding a decision maker's (DM) preferences. This approach was designed for additive decision models, where the overall utility of an alternative is the sum of weighted local utility functions for each criterion. The key idea behind ISMAUT is to use linear inequalities to represent the DM's preferences and reduce the set of alternatives by eliminating those that are dominated by others. Related works, such as those by Hazen [Haz86] and White [WSD84], have expanded on this concept, exploring similar challenges in decision-making under uncertainty.

ISMAUT addresses situations where a decision maker's utility function is parameterized using a set of weights and evaluators, reflecting their preferences across multiple criteria. It focuses on narrowing down both the set of possible utility functions and the set of alternatives. This process is done incrementally, based on the DM's feedback and preferences, which are formalized through linear constraints.

Model Structure The ISMAUT approach considers a finite set of alternatives denoted by A , and an additive utility function parameterized by a weight vector $w = (w_1, \dots, w_p)$ and evaluators $u = (u_1, \dots, u_p)$. The DM's utility function for an alternative $\alpha \in A$ is defined as:

$$u_{w,u}(\alpha) = \sum_{i=1}^p w_i u_i(\alpha) = w \cdot u(\alpha) \quad (2.8)$$

Here, w represents the trade-offs between different criteria and is constrained to be a point within the unit $(p-1)$ -simplex, i.e., $w \in \{w \in \mathbb{R}^p : w_i \geq 0, \sum_{i=1}^p w_i = 1\}$. The function $u : A \rightarrow \mathbb{R}^p$ maps each alternative to a p -dimensional vector of evaluations.

The goal is to refine the feasible set of parameters by gathering preference information from the DM, which is used to eliminate less favorable alternatives. This information can be classified into three main categories:

Weight Constraints: The DM can specify relationships or bounds between the weights of different criteria (e.g., $w_{\text{cost}} \geq w_{\text{time}}$ or $w_{\text{time}} \in [0, 0.5]$), thus reducing

the feasible space of the weight vector $\mathcal{W}$.

Evaluator Constraints: These constraints relate to the values of specific criteria. For instance, the DM can express preferences like $u_{\text{cost}}(\text{cheap}) \geq u_{\text{cost}}(\text{expensive})$ or define bounds on the utility values, such as $u_{\text{cost}}(\text{cheap}) \in [0, 0.2]$.

Pairwise Comparisons: If the DM prefers one alternative over another, this preference can be translated into a linear inequality, $w \cdot u(\alpha) \geq w \cdot u(\beta)$. The set $\Lambda = \{(\alpha, \beta) \in A \times A : \alpha \succ \beta\}$ captures these comparisons, which further restrict the set of possible parameters.

Thus, the aim is to refine the feasible set of utility functions $\mathcal{U}$ and consequently reduce the set of undominated alternatives. As more preference information is obtained from the DM, the size of the feasible parameter set $\mathcal{U}$ decreases, which in turn reduces the set of undominated alternatives. Several studies, such as [ILC01] and [GK03], have built upon ISMAUT by simplifying the parameterization of the utility function, focusing solely on the weight vector w . These approaches typically assume that the single-criterion evaluators are fixed, and the DM's preferences are elicited through a series of binary comparisons between alternatives. The response to each query is translated into a linear constraint, further narrowing the set of feasible weight vectors.

2.3.2.2 MMR

The *Minimax Regret* criterion, as introduced by Savage [Sav72] and further explored in decision theory and robust optimization [KY13, Ger98], is a fundamental approach used to tackle decision-making problems under uncertainty. More recently, its application has expanded to fields such as artificial intelligence (AI), where it serves to assess alternatives for decision-making scenarios, including recommendations and predictions. In these cases, uncertainty typically revolves around unknown or partially known parameters within a decision model [SH01, Bou02, BPPS06, BB06]. The criterion aims to minimize the worst-case potential loss, given the uncertainty about the true utility function parameters, represented by a set of feasible scenarios $\mathcal{U}$. This makes the minimax regret principle a suitable and robust method for evaluating alternatives in decision-support systems.

Within AI, the minimax regret approach is extensively used in the elicitation of preferences, such as determining multi-attribute utilities [WB03, BB12,

BP15] or resolving preference-related problems in ranking and voting contexts [LB11b, BDPV16, BPV17]. One of the key benefits of this method is its capacity to reduce cognitive load on decision-makers by posing simple, targeted preference queries, rather than requiring them to define complex utility function parameters upfront. The effectiveness of this approach has been demonstrated through various studies, including those involving real-world users [BB10].

For simplicity, we assume a set of parameters $\mathcal{U} \subset \mathbb{R}^p$, which is closed and bounded. For any scenario $w \in \mathcal{U}$, a continuous utility function $u_w(\alpha)$ is defined for each alternative $\alpha \in A$. However, the definitions that follow can be extended to more general sets of parameters $\mathcal{U}$ and utility functions u_w , by replacing $\max_{w \in \mathcal{U}}$ with $\sup_{w \in \mathcal{U}}$.

Pairwise Max Regret (PMR): The Pairwise Max Regret (PMR) of an alternative α relative to another alternative β within the set $\mathcal{W} \subseteq \mathcal{U}$ is defined as:

$$PMR_{\mathcal{W}}(\alpha, \beta) = \max_{w \in \mathcal{W}} (u_w(\beta) - u_w(\alpha)). \quad (2.9)$$

The $PMR_{\mathcal{W}}$ value represents the worst-case loss, across all feasible utility functions u_w given by $\mathcal{W}$, that results from choosing α over β . If $PMR_{\mathcal{W}}(\alpha, \beta) \leq 0$, this implies that $u_w(\alpha) \geq u_w(\beta)$ for all $w \in \mathcal{W}$, meaning α is at least as good as β given the available preference information.

Max Regret (MR): The Max Regret (MR) of an alternative α relative to a set of alternatives $A \subseteq A$ is given by:

$$MR_{\mathcal{W}}(\alpha, A) = \max_{\beta \in A} PMR_{\mathcal{W}}(\alpha, \beta) \quad (2.10)$$

This value represents the worst-case loss for recommending α instead of any other alternative in A , across all possible scenarios $\mathcal{W}$. It provides an upper bound on the loss in utility associated with selecting α over any other option in the set of alternatives.

Key Properties of Max Regret. Two well-known properties of max regret are stated below, with proofs included for completeness.

Lemma 2.3.1. *Let $A \subseteq \mathcal{A}$ be a finite set of alternatives and $\alpha \in A$. For $\mathcal{W} \subseteq \mathcal{U}$, $MR_{\mathcal{W}}(\alpha, A) \geq 0$.*

Proof. Since $\alpha \in A$ and the regret is maximized over all alternatives $\beta \in A$, setting $\beta = \alpha$ gives $u_w(\alpha) = u_w(\beta) = 0$ for any $w \in \mathcal{W}$. Therefore:

$$\begin{aligned} MR_{\mathcal{W}}(\alpha, A) &= \max_{\beta \in A} PMR_{\mathcal{W}}(\alpha, \beta) \\ &= \max_{\beta \in A} \max_{w \in \mathcal{W}} (u_w(\beta) - u_w(\alpha)) \\ &\geq \max_{\beta \in \{\alpha\}} \max_{w \in \mathcal{W}} (u_w(\beta) - u_w(\alpha)) = 0 \end{aligned} \tag{2.11}$$

□

Lemma 2.3.2. *Let $A \subseteq \mathcal{A}$ and $\alpha \in A$. For $\mathcal{W} \subseteq \mathcal{U}$, $MR_{\mathcal{W}}(\alpha, A) = 0$ if and only if $u_w(\alpha) \geq u_w(\beta)$ for all $w \in \mathcal{W}$ and $\beta \in A$.*

Proof. Following Lemma 2.3.1, $MR_{\mathcal{W}}(\alpha, A) \geq 0$. Therefore, $MR_{\mathcal{W}}(\alpha, A) = 0$ only when the inequality $u_w(\alpha) \geq u_w(\beta)$ holds for every scenario $w \in \mathcal{W}$ and every alternative $\beta \in A$. □

Minimax Regret (MMR): The Minimax Regret (MMR) for a finite set of alternatives A is defined as:

$$MMR_{\mathcal{W}}(A) = \min_{\alpha \in A} MR_{\mathcal{W}}(\alpha, A) \tag{2.12}$$

The $MMR_{\mathcal{W}}(A)$ represents the smallest possible maximum regret one can achieve by selecting the best alternative from A under the worst-case scenario. Selecting the alternative that minimizes the $MMR_{\mathcal{W}}$ offers the most robust choice under uncertainty, as it provides the least amount of regret across all possible scenarios in $\mathcal{W}$.

When the utility function u_w is linear, with respect to the parameters in $\mathcal{W} \subseteq \mathbb{R}^p$, the maximization of regret occurs at one of the extreme points of $\mathcal{W}$ [KVVA17]. This insight allows for efficient calculation of the MMR . Specifically, MMR can be computed also using linear programming, leveraging Equation 2.9 for PMR . This makes it a practical choice in complex decision-making environments.

2.3.3 Possibly Optimal Alternatives

In decision-making contexts, particularly those involving multiple criteria, an alternative is considered optimal based on how well it satisfies a decision-maker's preferences, which are often represented by a weight vector. Specifically, an alternative α is deemed *optimal* within a set of alternatives A with respect to a given weight vector w if, and only if, it yields a utility that is greater than or equal to that of any other alternative in the set. This condition is mathematically expressed as $u(\alpha, w) \geq u(\beta, w)$ for every $\beta \in A$.

However, in real-world scenarios, the exact weight vector w representing the decision-maker's preferences may not be known with certainty. In such cases, it becomes useful to consider the notion of *possibly optimal* alternatives, a concept widely applied in multiple criteria decision-making and relevant literature concerning decision problems with uncertain or incomplete preferences.

Definition 2.3.3. *An alternative $\alpha \in A$ is said to be possibly optimal in the set A , with respect to a set of possible weight vectors $\mathcal{W}$, if and only if there exists at least one weight vector $w \in \mathcal{W}$ such that α is optimal under w . Formally, this means that $u(\alpha, w) \geq u(\beta, w)$ for all other alternatives $\beta \in A$. In other words, under at least one possible configuration of the decision-maker's preferences represented by a weight vector $w \in \mathcal{W}$, the alternative α emerges as the best choice in the set A .*

We denote the set of all possibly optimal alternatives in A with respect to the set of weight vectors $\mathcal{W}$ as $PO(A, \mathcal{W})$. This set includes every alternative for which there exists some $w \in \mathcal{W}$ that makes it optimal.

The concept of possibly optimal alternatives is essential for decision-making under uncertainty, as it allows us to identify all alternatives that could be the best choice under some plausible preference scenario. This is particularly valuable when the decision-maker's exact preferences are unknown or when they are subject to change. Rather than focusing solely on a fixed preference structure, we broaden the search to consider a range of potential preferences, thereby ensuring that no viable option is prematurely disregarded.

Computing Possibly Optimal Alternatives. With linear utility functions the process of determining whether an alternative α belongs to the set of possibly optimal alternatives $PO(A, \mathcal{W})$ can be framed as a linear programming problem. This is because verifying whether $\alpha \in PO(A, \mathcal{W})$ involves determining whether there exists a weight vector $w \in \mathcal{W}$ such that the utility of α is at least as large

as the utility of any other alternative in A , given that particular w . Formally, this means evaluating the feasibility of the system of linear constraints:

$$u(\alpha, w) \geq u(\beta, w) \text{ for all } \beta \in A \setminus \{\alpha\} \text{ with } w \in \mathcal{W}$$

where w is constrained to lie within the set $\mathcal{W}$ of possible weight vectors.

By solving this system using linear programming techniques, we can determine whether there exists a feasible weight vector that makes α optimal, confirming its membership in $\text{PO}(A, \mathcal{W})$. This approach assumes that $\mathcal{W}$ is defined by a set of linear inequalities, meaning that verifying whether $\alpha \in \text{PO}(A, \mathcal{W})$ amounts to checking the feasibility of a linear constraint system. Consequently, this problem can be efficiently solved using an LP solver. As demonstrated in prior work, such as [TW22], this computational approach allows for the efficient identification of possibly optimal alternatives, even in cases where the set of alternatives and weight vectors are large or complex.

Importance of Focusing on Possibly Optimal Alternatives. The identification of possibly optimal alternatives plays a critical role in decision-making procedures. When faced with uncertainty regarding the true preference vector, the decision-maker's most preferred alternative must be optimal for some weight vector within the set $\mathcal{W}$ that represents the range of possible preferences. Therefore, by narrowing our attention to the set $\text{PO}(A, \mathcal{W})$, we can focus exclusively on alternatives that are possibly optimal, given some plausible preference scenario.

This approach has several key advantages: **Reduction in Complexity:** By filtering out alternatives that are not possibly optimal, we reduce the size of the decision problem. This pre-processing step simplifies the subsequent analysis by reducing the number of alternatives under consideration. However, in cases where the interaction terminates early, there is a risk that a minimax regret alternative may not be included in $\text{PO}(A, \mathcal{W})$, which could affect the final recommendation. Thus, ensuring sufficient elicitation is crucial for maintaining decision quality. This can be particularly useful in large decision spaces where evaluating every alternative would be computationally expensive.

Robust Decision-Making: Since the true preferences may not be fully known, focusing on possibly optimal alternatives ensures that no viable choice is overlooked. This leads to more robust decision-making, where the decision-maker retains flexibility in the face of uncertainty regarding their preferences.

Efficiency: Identifying possibly optimal alternatives early in the process can streamline decision support systems. Instead of evaluating all possible alternatives under every possible weight configuration, which may be computationally infeasible, decision-makers can concentrate on a smaller, manageable subset of candidates.

In practical applications, the pre-processing step of filtering out alternatives that are not in $PO(A, \mathcal{W})$ is often a crucial first step. By eliminating these alternatives upfront, the complexity of the decision-making process is significantly reduced. Since the decision-maker's most preferred option must lie within the set of possibly optimal alternatives, examining alternatives outside of $PO(A, \mathcal{W})$ becomes unnecessary. This allows decision analysts to focus their computational resources and analytical efforts on a subset of alternatives that are more likely to align with the decision-maker's true preferences.

Thus, the possibly optimal alternatives serve as a focal point in the decision-making process, ensuring that only the most relevant alternatives are considered in further analysis, which ultimately leads to more efficient and effective decision-making.

2.4 Preference Learning vs. Preference Elicitation

Understanding and modeling user preferences is essential for decision-support systems, including recommender systems and multi-criteria decision-making frameworks. Two primary approaches exist for capturing and refining these preferences:

- Preference Learning, which infers user preferences from observed behaviors, historical interactions, or implicit feedback using machine learning techniques.
- Preference Elicitation, which actively queries users to obtain explicit preference information, often through structured interactions that refine a decision model incrementally.

While both approaches aim to approximate a user's true preferences, they differ in methodology and application. Preference elicitation is often used in multi-criteria decision analysis, where direct user input is needed to construct a reliable preference model. Methods such as Bayesian preference elicitation 2.2.7 and minimax regret-based elicitation 2.3.2.2 aim to reduce

uncertainty about preferences through interactive questioning.

In contrast, preference learning is widely used in recommender systems, where explicit preference data is often unavailable, and patterns must be inferred from indirect user interactions [FH10]. More recent developments, such as Direct Preference Optimization (DPO) and Reinforcement Learning from AI Feedback (RLAIF), leverage large-scale AI models to refine preferences using implicit learning techniques [RSM⁺23, LPM⁺24].

2.4.1 Preference Learning Methods

Preference learning methods are widely used in AI-driven recommendation systems 2.2.4 to predict and model user preferences without direct elicitation. Several well-established techniques include:

1. *Collaborative Filtering* ([HKBR17, SKKR01, ERK11, RRS15, BZM12]): This method relies on the principle that users who have exhibited similar behaviors in the past will likely share similar preferences in the future. Collaborative filtering is divided into two main subtypes:
 - Item-based collaborative filtering: This approach analyzes the relationships between items based on user interaction patterns. If two items are frequently rated or purchased together by users with similar preferences, the system recommends one item to users who have shown interest in the other.
 - User-based collaborative filtering: This method identifies users with similar preferences by analyzing their past interactions. The system then recommends items that have been rated highly by similar users, assuming that like-minded individuals will share common interests.

Collaborative filtering is particularly effective in situations where users have little individual preference history, but there is a large pool of user data to draw from. However, it may suffer from the "cold start" problem, where recommendations for new users or new items are difficult due to limited initial data.

2. *Knowledge-Based Filtering* ([Bur02, MPBEE08, UFL⁺24]): Unlike content-based and collaborative filtering, which rely on historical data, knowledge-based filtering incorporates explicit knowledge about the user, items, and the relationships between them to generate recommendations. This ap-

proach is particularly useful in domains where user preferences are complex or change rapidly, such as when purchasing high-involvement products like cars or real estate. In these cases, the system requires more specific input from the user, such as functional requirements or constraints, which are then used to filter and prioritize items. For example, a user looking to purchase a car might specify that they prefer electric vehicles with a certain range and price, and the system would suggest models that meet those criteria. Knowledge-based systems are more flexible and can better accommodate users with specific needs or those who are new to the system without requiring extensive historical data.

3. *Content-Based Filtering* ([BKN07, LdGS11]): In content-based filtering, recommendations are generated by analyzing the features or attributes of items (such as products, movies, or articles) and comparing them to the user's preferences. These preferences are often inferred from the user's history of interactions, such as items they have previously liked or consumed. The system creates a profile based on these past preferences and suggests items that share similar attributes. For example, if a user frequently watches action movies, the system is likely to recommend other action-packed films. Content-based systems excel in scenarios where user preferences are well-defined and rely heavily on detailed item descriptions to make accurate predictions.
4. *Machine Learning for Preference Learning* ([FH10]): Recent advancements in preference learning leverage machine learning models to refine recommendations and decision-support systems:
 - **Pairwise Preference Learning:** Uses user comparisons (e.g., "Do you prefer A or B?") to train ranking models, commonly applied in decision-support systems.
 - **Ranking SVMs and Neural Networks:** Learn preference functions from ranking data, improving recommendation quality [FH10].
 - **Reinforcement Learning for Preferences:** Dynamically adapts recommendations by interacting with users over time, balancing exploration (learning new preferences) and exploitation (leveraging known preferences).
 - **Direct Preference Optimization (DPO):** A recent technique that refines AI models by directly learning from human preference data

without requiring reinforcement learning or separate reward models [RSM⁺23].

- Reinforcement Learning from AI Feedback (RLAIF): A method for scaling preference learning by training models using AI-generated feedback rather than human annotations [LPM⁺24].

These approaches allow for more efficient and scalable preference learning, particularly in AI-driven decision-making environments.

Each of these approaches has its strengths and limitations, making them suitable for different types of applications and user scenarios. Content-based methods are ideal for scenarios where the user's preferences are stable and well-understood, whereas collaborative filtering benefits from large, diverse user populations to discover new preferences. Knowledge-based systems, on the other hand, offer greater flexibility in handling complex, constraint-based decisions, especially where user preferences may not align neatly with historical patterns.

2.4.2 Relationship to Preference Elicitation

While preference learning passively infers preferences from user behavior, preference elicitation actively engages users to refine their preference models. The choice between the two depends on the application domain:

- In recommender systems, preference learning is more common because direct user input is often impractical.
- In decision analysis and multi-criteria decision-making, preference elicitation is preferred to ensure decisions align with user objectives.

The Bayesian and minimax regret elicitation methods represent structured frameworks where preference models are refined through direct user interactions. While these approaches provide strong theoretical guarantees, they may require more user effort than preference learning techniques.

2.5 Reflections and Motivation for the Proposed Approach

This chapter has outlined the foundational concepts and techniques in preference elicitation and decision-making, with a focus on multi-attribute utility theory, interactive and Bayesian methods, the minimax regret criterion, and the distinction between preference elicitation and preference learning. These methods address different aspects of handling preference uncertainty and have shaped the development of decision-support systems.

However, many existing approaches, particularly those based on Bayesian inference or strict constraint formulations, face limitations in terms of computational cost or sensitivity to noisy input. These challenges motivate the development of more efficient, noise-resilient methods for real-time preference elicitation.

In the next chapter, we build on this foundation and introduce a novel non-Bayesian preference elicitation framework specifically designed to address these limitations. This approach aims to balance efficiency, robustness, and scalability in multi-criteria decision-making contexts under noisy user feedback.

Chapter 3

Non-Bayesian Preference Elicitation Under Noisy Models

This chapter presents the main component of this thesis, a novel approach to interactive preference elicitation that can robustly manage noisy user inputs. Developing models capable of handling such noisy input preferences is essential in advancing artificial intelligence for interactive decision-making. Although Bayesian models provide a robust way to manage uncertainty in user preferences, they come with significant computational cost. The need for frequent model updates, optimal recommendation calculations, and query selection generates high processing demands, which restrict the practical use of Bayesian techniques in real-time applications. Furthermore, Bayesian approaches require an assumption of prior distributions over potential user preference models.

This work deals with multi-criteria decision-making problems by adopting a more qualitative approach to manage preference uncertainty. Instead of relying on Bayesian priors, we focus on identifying the most plausible user preference models and aim to generate a query strategy capable of finding an optimal alternative across these models. We propose a non-Bayesian, algorithm-driven method for recommendation and interactive elicitation, examining a broad spectrum of possible user models and evaluating their alignment with observed input preferences. This approach facilitates efficient query generation, offering real-time applicability with minimal computational cost. Experimental results indicate the effectiveness of our method, demonstrating high accuracy in identifying the user's preferred choice.

3.1 Introduction

Over the past decade, artificial intelligence technologies have made significant strides in areas like product recommendations, entertainment selection, and service suggestions [Bly02, Bou02, CGNS13, CKP00]. Consequently, AI-driven recommendation systems, which operate on behalf of users, must incorporate mechanisms capable of accurately capturing and assessing user preferences [KR93, SH01, Bou02].

Incremental elicitation methods work by querying users iteratively to gather new insights into their preferences. As users respond, the system gradually reduces uncertainty around their underlying preference model, often represented by a parameterized utility function. Unlike static protocols, these methods generate queries adaptively. At each step, the system can select the most informative query based on existing user data, adapting the process dynamically to refine the user model progressively.

The interaction continues until an optimal recommendation is identified, or a stopping condition is met, whether due to minimized information loss, reaching a threshold of time or cognitive load, or user fatigue. When the elicitation process ends prematurely, the system should still provide a recommendation grounded in the partial preference data gathered, approximating the user's preferences as accurately as possible [BPPS06, VB20].

To handle uncertainty in user preferences, many interactive preference elicitation methods rely on systematically reducing the preference space through iterative user interaction. These methods are widely studied in decision analysis and artificial intelligence, particularly in applications such as multi-criteria decision-making, multi-objective optimization, and recommender systems [BPPS06, VB20, Bou13]. One common class of methods is based on constraint-based elicitation, where user responses progressively restrict the set of feasible preference models until a decision can be made with high confidence.

For example, methods rooted in minimax regret rely on iteratively refining the set of possible utility parameters by treating user feedback as constraints. This approach simplifies model updates and allows rapid adaptation [WB03, BPPS06]. However, deterministic regret-based approaches assume that all user responses are precise and consistent, which is often unrealistic. Errors or inconsistencies in user responses can lead to biased parameter estimations, potentially excluding the true preference model and resulting in suboptimal

recommendations [BPPS06, BP18].

To address response uncertainty and inconsistency, probabilistic preference models provide an alternative by representing the decision maker’s preferences as probability distributions. These models explicitly account for noise in user responses, ensuring robustness to errors and inconsistencies. Bayesian methods, in particular, leverage prior distributions over the unknown parameters and refine them iteratively using Bayes’ theorem as new information is collected [CKP00, Bou02, GP04, GS10].

One foundational Bayesian approach in preference elicitation was introduced by Chajewska et al. [CKP00], which employed Bayesian updating to refine a probability distribution over a decision maker’s preferences based on their responses to pairwise gamble queries. This model was particularly useful for decision-making under uncertainty, as it tolerated inconsistent responses over time by maintaining a probabilistic belief over the true utility function.

In recent years, Bayesian models have been extended to multi-objective decision-making and combinatorial optimization, where the decision space is significantly larger. For example, Sauré and Vielma introduced an ellipsoidal credibility region approach, which uses a multivariate normal distribution to model uncertainty and refines the preference space in a Bayesian manner [SV19]. This approach improves error tolerance while ensuring that the model remains computationally tractable. Similarly, Vendrov et al. developed a Bayesian query selection strategy based on the Expected Value of Information (EVOI), which selects queries that maximize the expected reduction in uncertainty about the decision maker’s preferences [VLHB20].

Within incremental preference elicitation, Bayesian methods have demonstrated strong performance in handling noisy responses. One significant contribution in this field is the work of Bourdache, Perny, and Spanjaard, who proposed a Bayesian approach for multi-objective combinatorial optimization [?]. Their method assumes that preferences are modeled by an aggregation function with uncertain parameters, which are refined incrementally using Bayesian updates. This approach balances query informativeness with cognitive efficiency, optimizing the elicitation process by using mixed-integer linear programming (MILP) to compute expected regret minimization. By leveraging Bayesian inference, their model is able to handle inconsistencies in responses, ensuring robustness in decision-making.

Similarly, Guo and Sanner developed a Bayesian model for pairwise comparison queries, demonstrating that this technique effectively reduces the cognitive burden on users while maintaining elicitation accuracy [GS10]. Their framework uses logistic noise models to account for inconsistent answers, ensuring that small errors in preference judgments do not drastically alter the inferred utility function.

Despite their effectiveness, Bayesian preference elicitation methods face two primary challenges:

- Computational complexity: Updating the posterior distribution and selecting optimal queries require non-trivial inference procedures, which can be computationally expensive, especially in real-time applications.
- The choice of a prior distribution: Bayesian methods necessitate a prior over preference parameters, which may not always be straightforward to define in practice. In some cases, uninformative priors can be used, but these may slow down convergence rates, requiring more queries to achieve an accurate preference model [Bis16, TW87].

While adaptive prior selection and approximate inference techniques (such as Markov Chain Monte Carlo (MCMC) or variational inference [GCS⁺13, HBWP12, Mur12]) can partially mitigate these issues, further research is needed to develop scalable Bayesian preference elicitation models that balance computational efficiency with robust handling of user uncertainty.

This chapter presents an alternative, non-Bayesian incremental elicitation method designed to manage noisy responses efficiently. Instead of probabilistic assumptions, this approach employs a qualitative uncertainty model, concentrating on the subset of preference models most consistent with user feedback. This enables a robust elicitation process resilient to occasional user errors, while remaining efficient in both query count and computation time.

3.2 Problem Setting

Consider a system tasked with recommending the best option to a user from a finite set A of possible alternatives. Each alternative $\alpha \in A$ is characterized by a vector $(\alpha(1), \alpha(2), \dots, \alpha(p))$, where each component $\alpha(i)$ reflects the value of the alternative α concerning a specific criterion i . For simplicity, we assume without loss of generality that higher values indicate better performance for

each criterion, allowing for a straightforward comparison across options.

Utility Function We assume that the decision maker has a utility function $u : (\mathcal{W}, \mathcal{A}) \rightarrow \mathbb{R}$, which is parameterized by a vector $w \in \mathcal{W}$. Here, $\mathcal{W}$ represents the space of potential user preferences, defined as the set of all normalized non-negative weight vectors w such that $\sum_{i=1}^p w(i) = 1$ and $w(i) \geq 0$ for all $i = 1, \dots, p$. The utility function uses these weights to evaluate each alternative based on the weighted average of its criteria values, representing the decision maker's preferences.

Given a specific weight vector $w \in \mathcal{W}$, the utility of an alternative α is then calculated as $u(\alpha, w) = \sum_{i=1}^p w(i)\alpha(i)$. Let w^* denote the decision maker's true preference weights, which are typically unknown to the system. The preference relation $\alpha \succeq \beta$ indicates that the decision-maker prefers α over β , meaning that, for the true preferences w^* , $\alpha \succeq \beta$ if and only if $u(\alpha, w^*) \geq u(\beta, w^*)$. The goal is to identify the alternative most preferred by the decision maker, specifically the alternative that maximizes the utility function, i.e., $\arg \max_{\alpha \in \mathcal{A}} u(\alpha, w^*)$, without explicitly displaying all options or knowing the user's true preference weights w^* .

Example: Consider a scenario where a user needs to choose a rental property from a list of available options:

$$\mathcal{A} = \{\alpha = (12.7, 5, 3), \beta = (13, 3, 2), \gamma = (10.5, 3, 2)\}.$$

Here, each vector $(\alpha(1), \alpha(2), \alpha(3))$ represents a property's monthly rent, distance from the city center, and number of bedrooms, respectively. The criteria are scaled so that higher values indicate better performance. Suppose the user's preferences can be modeled by a weighted average with the weight vector $w^* = (0.7, 0.1, 0.2)$, indicating a strong preference for lower rent (high weight on the first criterion) with moderate consideration for proximity and room count.

The utility function $u(\alpha_i, w^*)$ outputs a real number representing the user's score for each property. Given this model, the most preferred property will be the one with the highest utility score. Calculating this:

$$u(\alpha, w^*) = 12.7 \times 0.7 + 5 \times 0.1 + 3 \times 0.2 = 9.99,$$

$$u(\beta, w^*) = 13 \times 0.7 + 3 \times 0.1 + 2 \times 0.2 = 9.8,$$

$$u(\gamma, w^*) = 10.5 \times 0.7 + 3 \times 0.1 + 2 \times 0.2 = 8.05$$

Thus, the system would identify α as the user's most preferred property. While this example assumes known user preferences, in real applications, these preferences are not directly accessible. Instead, interactive preference elicitation methods can be used to approximate w^* by engaging the user in a series of comparisons. This process aims to progressively reveal the user's optimal choice without requiring full disclosure of all weights or alternatives upfront.

Possibly Optimal Alternatives: An alternative α is considered optimal within a set of options A , given a specific weight vector w , if and only if its utility $u(\alpha, w)$ is at least as high as that of any other alternative β in A . Mathematically, this condition is satisfied if $u(\alpha, w) \geq u(\beta, w)$ for all $\beta \in A$. In other words, for a particular weight configuration w , α is the optimal alternative if it yields the maximum utility in the set.

The concept of possibly optimal alternatives was introduced in Chapter 2.3.3, where we defined $PO(A, \mathcal{W})$ and discussed its significance in decision-making under preference uncertainty. To briefly summarize, we define $PO(A, \mathcal{W})$ as the set of all possibly optimal alternatives in A , where each element of this set is optimal for at least one weight vector within $\mathcal{W}$. Here we focus only on the alternatives in $PO(A, \mathcal{W})$ because the decision-maker's most preferred alternative must be optimal for the true preference w^* . Thus, we do not need to consider alternatives $\beta \notin PO(A, \mathcal{W})$ and these alternatives can be filtered out as pre-processing.

Query-Based Preference Elicitation: Our approach relies on using binary queries, a method of gathering preference information by asking the decision maker to compare two alternatives directly. Specifically, each query takes the form of a pair (α, β) , where $\alpha, \beta \in A$ and $\alpha \neq \beta$. The question presented to the decision-maker is simply: "Do you prefer option α or option β ?"

This query model operates under the assumption that the decision maker's preference between two options reflects their underlying weight vector w^* . Given a query (α, β) , the decision-maker will prefer alternative α if and only if $w^* \cdot (\alpha - \beta) \geq 0$, i.e., the dot product (2.3) between w^* and the difference of

the utilities of α and β . Conversely, if $w^* \cdot (\alpha - \beta) < 0$, then the decision-maker prefers β . This direct binary comparison enables the system to iteratively refine its understanding of the user's preferences.

Many incremental preference elicitation approaches (e.g., [ZW76, SC83, VB20, TVW21]) employ this style of query to progressively limit the space of feasible weight vectors. Each answer effectively introduces a new constraint on the possible values of the decision-maker's weights, narrowing down the set of viable preference models. However, adding these constraints in a rigid manner can lead to potential inaccuracies; if the decision-maker provides an inconsistent or erroneous response, the true preference vector w^* could be inadvertently excluded from the feasible set.

This approach to querying offers a practical and efficient means of gathering preference information in real-time; this is particularly useful when large sets of alternatives or criteria make exhaustive preference surveys impractical. However, managing the impact of noise or errors in responses is critical in maintaining the integrity of the elicitation process. Thus, adaptive and probabilistic extensions to the binary query model—those that account for potential inaccuracies—are areas of ongoing development and are essential for making these systems robust in real-world applications.

User Model: Our framework adopts a straightforward user model defined by two key parameters: the true preference vector $w^* \in \mathcal{W}$ and a noise parameter ρ , where $0 \leq \rho < 1$ (for example, $\rho = 0.1$). The preference vector w^* represents the user's underlying weights across criteria, which determine their ideal preference order, while the noise parameter ρ reflects the likelihood of response inaccuracies or inconsistencies.

In each query (α, β) , the decision-maker's response aligns with their true preference w^* with probability $1 - \rho$. Specifically, the user will choose α over β if and only if $w^* \cdot (\alpha - \beta) \geq 0$ and, similarly, will prefer β if $w^* \cdot (\alpha - \beta) < 0$. However, with probability ρ , the user may respond incorrectly, introducing noise by selecting β over α even when $w^* \cdot (\alpha - \beta) \geq 0$ (or vice versa).

Crucially, neither the true preference vector w^* nor the noise parameter ρ is directly accessible to the system. The system can only infer information about these through the user's responses to each query. This uncertainty in responses, captured by the parameter ρ , allows the model to account for occasional decision-maker inconsistencies, which are common in real-world applications

due to factors like cognitive fatigue.

This model underpins the need for an elicitation approach that is resilient to noise, enabling robust inference of preferences even when responses are not perfectly reliable. By incorporating a probabilistic component into user modeling, this framework supports iterative refinement of preference estimations, leveraging both consistent and inconsistent responses to converge on a representation of the user’s true preference structure.

Simplifying Assumption. To streamline our approach, we assume that for each preference vector w , there exists a unique alternative α_w in the set A that maximizes the utility $u(\alpha, w)$. For the case where A consists of random, real-valued vectors, this assumption is generally valid and helps simplify the mathematical notation and the overall design of the algorithms. This assumption allows us to focus on a single optimal alternative per preference vector without needing to handle ties or indeterminate optima. However, it is important to note that all methods presented here could be adapted to cases where multiple optimal alternatives exist for a given preference vector, ensuring flexibility in application.

3.3 The Core Concept of Our Approach

Our approach is grounded in the idea of identifying a single optimal alternative under noisy conditions and uncertain user responses. If, hypothetically, there were a unique possibly optimal alternative α within the set A —meaning that $\text{PO}(A, \mathcal{W}) = \alpha$, then α should be recommended, as it would be the top choice across all possible user preference models represented by vectors $w \in \mathcal{W}$. This would imply that α is also the optimal alternative for the user’s unknown true preference vector w^* .

Similarly, if we could be certain that the user consistently provided accurate answers to all queries, and $\mathcal{U}$ represented the set of all preference vectors compatible with these answers, and if $\text{PO}(A, \mathcal{U}) = \alpha$, then we would confidently recommend α .

In practice, however, user responses are rarely flawless, as inconsistencies or errors may arise due to various factors. In this uncertain setting, we adjust our approach by focusing on the set of plausible preference vectors, denoted as $\mathcal{U}'$. This set, $\mathcal{U}'$, represents the most likely candidates for the true preference vector

w^* , even in the presence of occasional incorrect responses. Unlike Bayesian methods, we do not rely on any prior distribution over $\mathcal{W}$. Instead, we define plausibility based on the similarity of user responses to those we would expect if w were indeed w^* , assuming entirely accurate answers.

To evaluate plausibility, we consider each query (α, β) with a response favoring α . We assess whether $w \cdot (\alpha - \beta) \geq 0$, which would indicate that α is preferred to β under the weight vector w . The more often these conditions hold across queries, the more likely it is that w approximates the user's actual preference order. This notion of plausibility is formalized through a function, $mistakes(\cdot)$, which counts inconsistencies between observed user responses and what we would expect based on each candidate vector w .

Our method thereby refines the set of plausible preference vectors by using the observed consistency between user responses and each vector's predicted preference order, allowing the system to make recommendations even in the presence of noise and uncertainty. This approach balances the need for accurate preference elicitation with the practical limitations of noisy user feedback, optimizing both the efficiency and robustness of the recommendation process.

The Function $mistakes(w)$: To identify the preference vectors in $\mathcal{W}$ that most closely align with the decision-maker's observed preferences, we use a function $mistakes(w)$ to count the number of inconsistencies (or "mistakes") each vector $w \in \mathcal{W}$ would produce if it were assumed to be the true preference vector, w^* . Specifically, $mistakes(w)$ represents the number of times that the inequality $w \cdot (\alpha - \beta) \geq 0$ is not satisfied, given the user's responses to each query (α, β) when the answer indicates a preference for α .

In practical terms, each query result is checked for consistency against the predicted preference order of the vector w . For instance, if the user indicates a preference for α over β in a query (α, β) , but $w \cdot (\alpha - \beta) < 0$, we increase $mistakes(w)$ by one. This process is repeated for each query to build a cumulative count of inconsistencies for the vector w .

However, because users may respond incorrectly due to noise, uncertainty, or cognitive biases, we assume a probability ρ of making a mistake per query. That is:

- With probability $1 - \rho$, the user answers correctly (selects the better option according to their true utility function).

- With probability ρ , the user answers incorrectly (chooses the worse option).

If the system asks K queries, and each query has an independent probability ρ of being answered incorrectly, then the total number of mistakes made by the user follows a binomial distribution:

$$\text{mistakes}(w^*) \sim \text{Binomial}(K, \rho) = B(K, \rho)$$

Note: A binomial distribution models the number of successes (or failures) in a sequence of independent trials. In this case:

- Each query is a trial.
- The user has a probability ρ of making a mistake in each trial.
- The total number of mistakes is the sum of independent Bernoulli trials, which follows a Binomial distribution.

Thus, the expected number of mistakes (mean of the binomial distribution) is:

$$\mathbb{E}[\text{mistakes}(w^*)] = \rho K$$

This means that, on average, the user will make ρK mistakes across K queries.

Since we do not have direct knowledge of $\text{mistakes}(w^*)$ (as w^* is unknown), we can instead focus on identifying the set $\mathcal{W}_{\min}^k$ of plausible preference vectors. This set includes those vectors w for which $\text{mistakes}(w)$ is within a specified threshold k of the minimum number of mistakes observed across all preference vectors in $\mathcal{W}$. By focusing on this subset, we capture the most likely candidates for the true preference vector, thereby refining our model even in the presence of response noise.

Finite Approximation $\mathcal{W}'$ of $\mathcal{W}$: To make computation more feasible, we approximate the continuous space of preference vectors $\mathcal{W}$ with a finite set of sampled points, denoted $\mathcal{W}'$. By reducing $\mathcal{W}$ to a manageable subset, we can perform calculations more efficiently while still capturing a representative sample of potential user preferences.

In our experiments, we generate a finite subset $\mathcal{W}'$ of the preference space $\mathcal{W}$ by randomly sampling weight vectors from a uniform distribution over

the probability simplex. This ensures that all potential user preference models are equally likely to be considered, preventing bias in the elicitation process. The probability simplex constraint $\sum w_i = 1, w_i \geq 0$ ensures that the sampled weights remain valid for multi-criteria decision-making. To maintain computational efficiency and consistency throughout the iterative elicitation process, we keep the same fixed set $\mathcal{W}'$ rather than dynamically resampling at each step. While this approach enhances query efficiency and simplifies preference comparisons, it also introduces a potential limitation: if the true user preference lies outside the sampled set, the system may approximate rather than precisely capture the user's actual utility function.

The Set $\mathcal{W}_{\min}^k$ of k -Plausible Preference Vectors To identify the most plausible preference vectors based on the user's responses, we define a subset $\mathcal{W}_{\min}^k$, which contains the vectors in $\mathcal{W}'$ that most closely align with the decision-maker's observed preferences. First, we determine the minimum number of inconsistencies, denoted by μ , by finding the smallest value of $mistakes(w)$ across all vectors $w \in \mathcal{W}'$. For a given parameter $k \geq 0$, the set $\mathcal{W}_{\min}^k$ includes all vectors $w \in \mathcal{W}'$ for which $mistakes(w) \leq \mu + k$. This subset captures the preference vectors that are within a threshold k of the minimal inconsistency, thus representing the most plausible user models. We use $\mathcal{W}_{\min}^k$ to guide query generation, focusing on comparisons that refine our understanding of user preferences.

It is important to note that $\mathcal{W}_{\min}^k$ depends on the specific finite set $\mathcal{W}'$, which is randomly chosen. Although our notation does not explicitly reflect this dependence, it is an important factor influencing the composition of $\mathcal{W}_{\min}^k$.

We define that two preference vectors w and w' differ on a query (α, β) if one of the following holds: (a) $w \cdot \alpha \geq w \cdot \beta$ and $w' \cdot \alpha < w' \cdot \beta$, or (b) $w' \cdot \alpha \geq w' \cdot \beta$ and $w \cdot \alpha < w \cdot \beta$. This condition helps illustrate the degree to which different preference vectors align with or diverge from the true user preference w^* in response to specific queries.

This framework provides insights into the structure of $\mathcal{W}_{\min}^k$ as we collect more responses through successive queries, gradually refining the set to include only the most consistent preference models. This narrowing of plausible models becomes essential in efficiently converging on the user's true preferences within a finite and computationally practical space.

Proposition 1. *Let $w \in \mathcal{W}$. Consider a sequence of queries, and let K_w be the*

number of queries in the sequence in which w and w^* differ. Let the random variable $\mathbf{X}_w^{w^*}$ be equal to $\text{mistakes}(w) - \text{mistakes}(w^*)$. Then $\mathbf{X}_w^{w^*} \sim K_w - 2B(K_w, \rho)$, where $B(K_w, \rho)$ is a binomial distribution with K_w experiments and probability of success ρ . The expected value $E[\mathbf{X}_w^{w^*}]$ of $\mathbf{X}_w^{w^*}$ is equal to $K_w(1 - 2\rho)$, and the standard deviation of $\mathbf{X}_w^{w^*}$ equals $2\sqrt{K_w\rho(1 - \rho)}$.

Proof: Let us label the queries in the sequence on which w and w^* differ as (α_i, β_i) for $i = 1, \dots, K_w$, and let the Boolean random variable $\mathbf{e}_i$ be such that $\mathbf{e}_i = 1$ if and only if the user answers query (α_i, β_i) incorrectly. The variables $\mathbf{e}_i$ for $i = 1, \dots, K_w$ are independent with $\Pr(\mathbf{e}_i = 1) = \rho$. If $\mathbf{e}_i = 1$ then $\text{mistakes}(w^*)$ is incremented and $\text{mistakes}(w)$ is unchanged, so $\text{mistakes}(w) - \text{mistakes}(w^*)$ is decremented; and if $\mathbf{e}_i = 0$ then $\text{mistakes}(w)$ is incremented and so $\text{mistakes}(w) - \text{mistakes}(w^*)$ is incremented. So, in both cases, $\mathbf{X}_w^{w^*}$ changes by $1 - 2\mathbf{e}_i$. (For the other queries, on which w and w^* do not differ, $\text{mistakes}(w) - \text{mistakes}(w^*)$ is unchanged.) Thus, $\mathbf{X}_w^{w^*} = \text{mistakes}(w) - \text{mistakes}(w^*)$ is equal to $\sum_{i=1}^{K_w} (1 - 2\mathbf{e}_i)$, i.e., $K_w - 2\sum_{i=1}^{K_w} \mathbf{e}_i$. Therefore, $\mathbf{X}_w^{w^*} \sim K_w - 2B(K_w, \rho)$, because $\sum_{i=1}^{K_w} \mathbf{e}_i$ has the binomial distribution $B(K_w, \rho)$.

The expected value of $\sum_{i=1}^{K_w} \mathbf{e}_i$ is $K_w\rho$, and so $E[\mathbf{X}_w^{w^*}] = K_w(1 - 2\rho)$. The variance of $\mathbf{e}_i$ is equal to $E[(\mathbf{e}_i)^2] - (E[\mathbf{e}_i])^2 = \rho - \rho^2$, and so the variance of $\mathbf{X}_w^{w^*}$, which equals the variance of $2\sum_{i=1}^{K_w} \mathbf{e}_i$, is $4K_w\rho(1 - \rho)$; hence the standard deviation of $\mathbf{X}_w^{w^*}$ is equal to $2\sqrt{K_w\rho(1 - \rho)}$. $\square$

Using the notation of Proposition 1 we have the following consequence.

Corollary 1. Assume $w^* \in \mathcal{W}'$.

- (i) If $\mathbf{X}_w^{w^*} > \kappa$ then $w \notin \mathcal{W}_{min}^\kappa$.
- (ii) If $w \in \mathcal{W}_{min}^\kappa$ and $K_w(1 - 2\rho) > \kappa$ then the random variable $\mathbf{X}_w^{w^*}$ is at least $\frac{K_w(1-2\rho)-\kappa}{2\sqrt{K_w\rho(1-\rho)}}$ standard deviations less than its mean.

Proof: (i) Assume $\mathbf{X}_w^{w^*} > \kappa$. Since $w^* \in \mathcal{W}'$ we have $\text{mistakes}(w^*) \geq \mu$. Then $\text{mistakes}(w) - \mu \geq \text{mistakes}(w) - \text{mistakes}(w^*) = \mathbf{X}_w^{w^*} > \kappa$, so $\text{mistakes}(w) - \mu > \kappa$, and thus, $w \notin \mathcal{W}_{min}^\kappa$.

(ii) By part (i), $w \in \mathcal{W}_{min}^\kappa$ implies $\mathbf{X}_w^{w^*} \leq \kappa$. By Proposition 1, the random variable has mean $K_w(1 - 2\rho)$ and standard deviation of $2\sqrt{K_w\rho(1 - \rho)}$, and $\frac{K_w(1-2\rho)-\mathbf{X}_w^{w^*}}{2\sqrt{K_w\rho(1-\rho)}} \geq \frac{K_w(1-2\rho)-\kappa}{2\sqrt{K_w\rho(1-\rho)}}$, so $\mathbf{X}_w^{w^*}$ is at least $\frac{K_w(1-2\rho)-\kappa}{2\sqrt{K_w\rho(1-\rho)}}$ standard deviations less than its mean. $\square$

Corollary 1 implies that the w (in $\mathcal{W}'$) with high K_w will tend to not be in $\mathcal{W}_{min}^\kappa$. Indeed, if K_w is high then $K_w(1 - 2\rho)$, the expected value of $\mathbf{X}_w^{w^*}$, is high too. By Corollary 1(i), this means that w is not in $\mathcal{W}_{min}^\kappa$.

In particular, if w^* was in $\mathcal{W}'$ and κ was small (such as $\kappa \in \{0, 1, 2\}$), then it is very unlikely for w to be in $\mathcal{W}_{min}^\kappa$ unless K_w is fairly small. This is because if w were in $\mathcal{W}_{min}^\kappa$ then, by Corollary 1(ii), we would have the approximately normally distributed random variable $\mathbf{X}_w^{w^*}$ being at least $\frac{K_w(1-2\rho)-\kappa}{2\sqrt{K_w\rho(1-\rho)}}$ standard deviations less than its mean. For instance, with $\kappa = 2$ and $\rho = 0.1$ and $K_w = 10$ (respectively, $K_w = 15$), $\mathbf{X}_w^{w^*}$ will be more than 3 (respectively, 4.3) standard deviations from its mean.

The following result, which is quite similar in its proof to Proposition 1, is useful for the case when $w^* \notin \mathcal{W}'$.

Proposition 2. *Let $v, w \in \mathcal{W}$. Consider a sequence of queries, and, for $u \in \{v, w\}$ let K_u be the number of queries in the sequence in which u and w^* differ; let K_w^v be the number of queries in the sequence in which w differs from both w^* and v (and so v and w^* agree), and define K_v^w analogously. Then $K_w - K_v = K_w^v - K_v^w$. Let the random variable $\mathbf{Z}_w^v$ be equal to $\text{mistakes}(w) - \text{mistakes}(v)$. Then $\mathbf{Z}_w^v = \mathbf{X}_w^{w^*} - \mathbf{X}_v^{w^*}$, and $\mathbf{Z}_w^v \sim (K_w - K_v) - 2B(K_w^v, \rho) + 2B(K_v^w, \rho)$, with the two binomial distributions being independent. The expected value $E[\mathbf{Z}_w^v]$ of $\mathbf{Z}_w^v$ is equal to $(K_w - K_v)(1 - 2\rho)$, and the standard deviation of $\mathbf{Z}_w^v$ equals $2\sqrt{(K_w^v + K_v^w)\rho(1 - \rho)} \leq 2\sqrt{(K_w + K_v)\rho(1 - \rho)}$.*

Proof: K_w can be written as $K_w^v + L$, where L is the number of queries in which w and v both disagree with w^* . Similarly, $K_v = K_v^w + L$, so $K_w - K_v = K_w^v - K_v^w$ and $K_w \geq K_w^v \geq 0$ and $K_v \geq K_v^w \geq 0$.

Let us label the queries in the sequence on which w differs from both v and w^* as (α_i, β_i) for $i = 1, \dots, K_w^v$, and let the Boolean random variable $\mathbf{e}_i$ be such that $\mathbf{e}_i = 1$ if and only if the user answers query (α_i, β_i) incorrectly.

Similarly, let us label the queries in the sequence on which v differs from both w and w^* as (α'_j, β'_j) for $j = 1, \dots, K_v^w$, and let the Boolean random variable $\mathbf{e}'_j$ be such that $\mathbf{e}'_j = 1$ if and only if the user answers query (α'_j, β'_j) incorrectly.

If v differs from both w and w^* on a query then w agrees with w^* ; hence, the variables $\mathbf{e}_i$ for $i = 1, \dots, K_w^v$ and the variables $\mathbf{e}'_j$ for $j = 1, \dots, K_v^w$, do not overlap, and so are all mutually independent, with $\Pr(\mathbf{e}_i = 1) = \Pr(\mathbf{e}'_j = 1) = \rho$. Also, $\mathbf{Z}_w^v$ is equal to $\sum_{i=1}^{K_w^v} (1 - 2\mathbf{e}_i) - \sum_{j=1}^{K_v^w} (1 - 2\mathbf{e}'_j) =$

$K_w^v - K_v^w - 2 \sum_{i=1}^{K_w^v} \mathbf{e}_i + 2 \sum_{j=1}^{K_v^w} \mathbf{e}'_j$, i.e., $K_w - K_v - 2 \sum_{i=1}^{K_w^v} \mathbf{e}_i + 2 \sum_{j=1}^{K_v^w} \mathbf{e}'_j$. Therefore, $\mathbf{Z}_w^v \sim K_w - K_v - 2B(K_w^v, \rho) + 2B(K_v^w, \rho)$, involving independent binomial distributions, i.e., binomial distributions on independent variables.

The expected value of each $\mathbf{e}_i$ and $\mathbf{e}'_j$ are both ρ , and their variances are both $\rho(1 - \rho)$. Hence, the expected value of $\mathbf{Z}_w^v$ is $K_w^v - 2K_w^v\rho - K_v^w + 2K_v^w\rho = (K_w^v - K_v^w)(1 - 2\rho) = (K_w - K_v)(1 - 2\rho)$. And the variance of $\mathbf{Z}_w^v$ is $4\rho(1 - \rho)K_w^v + 4\rho(1 - \rho)K_v^w = 4\rho(1 - \rho)(K_w^v + K_v^w) \leq 4\rho(1 - \rho)(K_w + K_v)$, and so the standard deviation of $\mathbf{Z}_w^v$ is at most $2\sqrt{(K_w + K_v)\rho(1 - \rho)}$. $\square$

Even if $w^* \notin \mathcal{W}'$, the expected value of $\mathbf{Z}_w^v = \mathbf{X}_w^{w^*} - \mathbf{X}_v^{w^*} = \text{mistakes}(w) - \text{mistakes}(v)$ is equal to $(K_w - K_v)(1 - 2\rho)$ (for $w, v \in \mathcal{W}'$), by Proposition 2; also, the standard deviation of $\mathbf{Z}_w^v$ increases only proportionally with $2\sqrt{K_w^v + K_v^w} \leq 2\sqrt{K_w + K_v}$. Random variable $\mathbf{Z}_w^v$ is approximately normal, as a difference between two independent distributions that are approximately normal. Thus, if K_w is substantially larger than K_v then $\mathbf{Z}_w^v$ will very likely be greater than κ , and so w will then not be in $\mathcal{W}_{\min}^\kappa$. Thus, $\mathcal{W}_{\min}^\kappa$ will tend to contain the $w \in \mathcal{W}'$ with smallest K_w , i.e., that differ on fewest queries with the true model w^* .

More generally, the weights vectors w that order the alternatives in A most similarly to how w^* orders them, will tend to be in $\mathcal{W}_{\min}^\kappa$, increasingly so as we ask more queries. Therefore, the most plausible user models w , i.e., those most likely to be the true user preference model w^* (or close to it), are those with smaller values of $\text{mistakes}(w)$, which is why $\mathcal{W}_{\min}^\kappa$ may be considered as consisting of the most plausible user preference models.

As the iterative query process progresses, preference vectors w that order the alternatives in A in a way that closely resembles the ordering given by the true preference vector w^* will increasingly populate the set $\mathcal{W}_{\min}^k$. Consequently, the most plausible user models—those that are likely to approximate or match w^* —tend to have lower values of $\text{mistakes}(w)$. This makes $\mathcal{W}_{\min}^k$ a subset representing the most plausible user preference models, effectively narrowing down the feasible options as more queries are answered and inconsistencies are minimized.

Stopping Criterion: Our algorithm leverages the refined set $\mathcal{W}_{\min}^k$, specifically with values of k ranging from 0 up to a chosen threshold k , which we set at $k = 2$ in our experimental framework. The algorithm's stopping criterion is met when all models within $\mathcal{W}_{\min}^k$ unanimously agree on the most preferred

alternative α . In formal terms, this condition is expressed as $\text{PO}(\mathcal{A}, \mathcal{W}_{\min}^k) = \alpha$. Achieving this consensus indicates that the model has converged on an alternative that is optimal across the most plausible user preference models.

To maximize efficiency, our query strategies are designed to satisfy this stopping criterion as swiftly as possible. Specifically, we focus on queries involving pairs of alternatives (α_v, α_w) where $v, w \in \mathcal{W}_{\min}^k$. This approach is both practical and targeted: such queries are always available if the stopping criterion has not yet been met. Moreover, each query is likely, with probability $1 - \rho$, to increase the *mistakes* count for any vector w that differs from the true preference vector w^* in its ordering of the pair of selected alternatives.

If we continue to pose queries that emphasize inconsistencies in non-optimal models, we incrementally exclude those vectors from $\mathcal{W}_{\min}^k$. Over time, and with high probability, these iterations will refine $\mathcal{W}_{\min}^k$ until only the most accurate representations of the user's true preferences remain, ultimately satisfying the stopping criterion and enabling a reliable recommendation.

The query selection, invoked through $\text{Select_query}(\mathcal{A}, \mathcal{W}_{\min}^k)$ at line 3, uses one of the methods detailed in Section 3.4. Following each query, the decision maker's response is used to update the $\text{mistakes}(w)$ function for each vector $w \in \mathcal{W}'$, refining the assessment of each vector's plausibility. The set $\mathcal{W}_{\min}^k$ is then recalculated for $k \in 0, 1, 2$ to reflect the updated preference structure based on the latest response.

This cycle of querying, updating, and refining continues until the set of possibly optimal alternatives, $\text{PO}(\mathcal{A}, \mathcal{W}_{\min}^2)$, is reduced to a single element, denoted α . At this point, the set $\text{PO}(\mathcal{A}, \mathcal{W}_{\min}^1)$ and $\text{PO}(\mathcal{A}, \mathcal{W}_{\min}^0)$ must also be the singleton set $\{\alpha\}$, signifying that all plausible user models in $\mathcal{W}_{\min}^k$ agree on the preferred choice. This alternative α is then recommended as the system's estimate of the decision-maker's most favored option.

The intuition behind this iterative exclusion process is that the loop effectively filters out weight vectors in $\mathcal{W}_{\min}^k$ for $k \in 0, 1, 2$ that display preference orders markedly different from the user's true preferences, w^* . As the iterations progress, the vectors remaining in $\mathcal{W}_{\min}^k$ are increasingly likely to align closely with the decision-maker's genuine preference structure, making the recommendation of α a robust reflection of the user's optimal choice.

Algorithm 1

```

1: procedure RECOMMEND_ALTERNATIVE( $A, \mathcal{W}'$ )
2:   repeat
3:      $q \leftarrow \text{Select\_query}(A, \mathcal{W}_{\min}^k \text{ for } k \in \{0, 1, 2\})$ 
4:     Ask query  $q$  to the decision-maker
5:     Update  $\mathcal{W}_{\min}^k$  for  $k \in \{0, 1, 2\}$ 
6:   until  $|\text{PO}(A, \mathcal{W}_{\min}^2)| = 1$ 
7:   return The unique alternative in  $\text{PO}(A, \mathcal{W}_{\min}^2)$ 

```

3.4 Query Selection

In this section, we introduce two methods for selecting queries that aim to effectively narrow down the user's preferences.

Method 1: Max Preference Points for Query Selection. The first method, Max Preference Points, selects a query by identifying a pair of alternatives (α, β) that are optimal across the maximum number of weight vectors $w \in \mathcal{W}_{\min}^k$. This approach prioritizes alternatives that are favored by the greatest number of plausible user models, starting with lower values of k to focus on more certain preferences first.

To formalize this, let $\text{Opt}^k(\alpha)$ represent the count of preference points $w \in \mathcal{W}_{\min}^k$ for which α is the most preferred alternative according to the utility function $u(\alpha, w)$. We select the query (α, β) according to the following rules:

Primary Selection: If the set $\text{PO}(A, \mathcal{W}_{\min}^0)$ contains more than one alternative, select a query (α, β) such that both α and β are in $\text{PO}(A, \mathcal{W}_{\min}^0)$. Additionally, ensure that $\text{Opt}^0(\alpha) \geq \text{Opt}^0(\gamma)$ and $\text{Opt}^0(\beta) \geq \text{Opt}^0(\gamma)$ for every other alternative $\gamma \in \text{PO}(A, \mathcal{W}_{\min}^0) \setminus \{\alpha, \beta\}$.

Secondary Selection: If $\text{PO}(A, \mathcal{W}_{\min}^0)$ has reduced to a single element α_0 but $\text{PO}(A, \mathcal{W}_{\min}^1)$ still contains multiple alternatives, select a query (α_0, β) where β belongs to $\text{PO}(A, \mathcal{W}_{\min}^1)$, is distinct from α_0 , and satisfies $\text{Opt}^1(\beta) \geq \text{Opt}^1(\gamma)$ for all other $\gamma \in \text{PO}(A, \mathcal{W}_{\min}^1) \setminus \{\alpha_0, \beta\}$.

Tertiary Selection: If both $\text{PO}(A, \mathcal{W}_{\min}^0)$ and $\text{PO}(A, \mathcal{W}_{\min}^1)$ have converged to the same single element α_0 , but $\text{PO}(A, \mathcal{W}_{\min}^2)$ contains multiple alternatives, select a query (α_0, β) such that $\beta \in \text{PO}(A, \mathcal{W}_{\min}^2)$, $\beta \neq \alpha_0$, and $\text{Opt}^2(\beta) \geq \text{Opt}^2(\gamma)$ for all $\gamma \in \text{PO}(A, \mathcal{W}_{\min}^2) \setminus \{\alpha_0, \beta\}$.

By following these steps, Method 1 systematically selects queries that reduce the ambiguity in preference ordering, homing in on the user's most likely preference model based on current data.

Method 2: Best Split for Query Selection. In addition to Method 1, we also consider a second method for selecting queries, designed to further refine preference elicitation and address any remaining uncertainties in the user's preference model.

In the second query selection approach, referred to as the Best Split method, we select a query (α, β) where both α and β are elements of $\text{PO}(\mathbf{A}, \mathcal{W}_{\min}^k)$, aiming to divide the preference points in $\mathcal{W}_{\min}^k$ into two balanced groups. Specifically, we look for a query that splits $\mathcal{W}_{\min}^k$ as evenly as possible into the sets $w : (\alpha - \beta) \cdot w \geq 0$ and $w : (\alpha - \beta) \cdot w < 0$. This balanced division helps maximize the information gained from each query by equally distinguishing between vectors that favor either α or β .

To implement this approach, we first generate a candidate set of potential queries Q , as follows:

Initial Set of Potential Queries: If the set $\text{PO}(\mathbf{A}, \mathcal{W}_{\min}^0)$ contains multiple alternatives, we set $k = 0$ and define Q as $(\alpha, \beta) : \alpha, \beta \in \text{PO}(\mathbf{A}, \mathcal{W}_{\min}^0), \alpha \neq \beta$, representing all pairs of alternatives within $\text{PO}(\mathbf{A}, \mathcal{W}_{\min}^0)$.

Secondary Set of Potential Queries: If $\text{PO}(\mathbf{A}, \mathcal{W}_{\min}^0)$ has reduced to a single alternative, denoted α_0 , we then proceed with the following checks:

If $\text{PO}(\mathbf{A}, \mathcal{W}_{\min}^1)$ contains multiple alternatives, we set $k = 1$ and define Q as $(\alpha_0, \beta) : \beta \in \text{PO}(\mathbf{A}, \mathcal{W}_{\min}^1), \beta \neq \alpha_0$, including pairs composed of α_0 and each alternative in $\text{PO}(\mathbf{A}, \mathcal{W}_{\min}^1)$. If $\text{PO}(\mathbf{A}, \mathcal{W}_{\min}^1)$ also reduces to a single alternative, then we increase k to 2 and define Q as $(\alpha_0, \beta) : \beta \in \text{PO}(\mathbf{A}, \mathcal{W}_{\min}^2), \beta \neq \alpha_0$, where each query pairs α_0 with alternatives in $\text{PO}(\mathbf{A}, \mathcal{W}_{\min}^2)$. After constructing Q , we select the query that maximizes the informational value by computing a score using Algorithm 2, with Q and $\mathcal{W}_{\min}^k$ as inputs. The value of k is determined by the process described above for generating Q . In Algorithm 2, for each query $q = (\alpha, \beta)$, we define $\lambda_q = \alpha - \beta$, representing the vector difference between the two alternatives.

This Best Split approach enhances the efficiency of the query selection process by focusing on maximally informative splits, which enables the algorithm to home in on the user's true preference model with fewer queries.

Algorithm 2

```

1: procedure SELECT_QUERY( $Q, \mathcal{W}_{min}^k$ )
2:    $query\_score \leftarrow \infty$ 
3:   for  $q$  in  $queries$  do
4:      $min\_score \leftarrow 0$ 
5:      $max\_score \leftarrow 0$ 
6:     for  $w$  in  $\mathcal{W}_{min}^k$  do
7:        $res = \lambda_q \cdot w$ 
8:       if  $res > 0$  then
9:          $min\_score \leftarrow min\_score + 1$ 
10:      if  $res < 0$  then
11:         $max\_score \leftarrow max\_score + 1$ 
12:       $maxmin\_score \leftarrow \max(min\_score, max\_score)$ 
13:      if  $maxmin\_score < query\_score$  then
14:         $query \leftarrow q$ 
15:       $query\_score \leftarrow maxmin\_score$ 
return  $query$ 

```

Using both Methods 1 and 2, if the size of the set $PO(A, \mathcal{W}_{min}^0)$ is greater than one, we select a query in the form $(\alpha v, \alpha w)$, where $v, w \in \mathcal{W}_{min}^0$. This choice aims to reduce the size of $\mathcal{W}_{min}^0$ and consequently decrease $|PO(A, \mathcal{W}_{min}^0)|$ in the next iteration. Through this process, each query incrementally filters out preference vectors, either reducing $PO(A, \mathcal{W}_{min}^0)$ or leaving it unchanged.

This iterative reduction of $\mathcal{W}_{min}^0$ continues until $PO(A, \mathcal{W}_{min}^0)$ is reduced to a single alternative α_0 . At this point, the algorithm proceeds by selecting a query of the form (α_0, β) , where β is an alternative within $\mathcal{W}_{min}^k$ for some k .

However, if the user's response indicates a preference for β over α_0 , this preference will influence $\mathcal{W}_{min}^0$, and it's possible that $|PO(A, \mathcal{W}_{min}^0)|$ could once again exceed one in the following stage. In such cases, the algorithm will repeat the cycle: issuing queries that refine $PO(A, \mathcal{W}_{min}^0)$ until it once again converges to a singleton, and so on, allowing for dynamic adaptation based on each new response.

This approach provides flexibility, adapting to unexpected user responses by re-expanding $PO(A, \mathcal{W}_{min}^0)$ when necessary, ensuring that the elicitation process accurately captures the user's preference even if initial reductions don't fully resolve ambiguities.

Challenges with Finite Approximation $\mathcal{W}'$ of $\mathcal{W}$

The finite approximation of the preference space, represented by the subset $\mathcal{W}' \subset \mathcal{W}$, introduces certain challenges that can impact the accuracy of our recommendations.

The first challenge is that our approach can only recommend an alternative that is optimal for at least one preference vector within the finite set $\mathcal{W}'$. In other words, it can only choose an element from $\text{PO}(\mathbf{A}, \mathcal{W}')$. If the user's actual optimal choice, denoted as α_{w^*} , is not part of $\text{PO}(\mathbf{A}, \mathcal{W}')$, then the algorithm will be unable to recommend the correct alternative, as it lies outside the feasible options defined by $\mathcal{W}'$.

A second challenge arises even if α^* is included in $\text{PO}(\mathbf{A}, \mathcal{W}')$ as α_v for some vector $v \in \mathcal{W}'$. In this case, while α^* is the top choice according to both w^* and v , the preference rankings associated with v may still diverge significantly from those of the user's true preference model w^* . For instance, consider a scenario with $\mathbf{A} = \{\alpha = (1, 0, 0), \beta = (0.5, 0.3, 0.2), \gamma = (0.5, 0.2, 0.3)\}$, where $w^* = (0.5, 0.5, 0)$ and $v = (0.5, 0, 0.5)$. Here, both $\alpha_{w^*} = \alpha$ and $\alpha_v = \alpha$, meaning both vectors rate α as optimal. However, they differ in ranking β and γ : $w^* \cdot (\beta - \gamma) \geq 0$ whereas $v \cdot (\beta - \gamma) < 0$.

If the decision maker correctly answers the query (β, γ) in line with w^* , this would increase the $\text{mistakes}(v)$ count by one. Such discrepancies can lead to a situation where $\text{mistakes}(v)$ becomes notably larger than $\text{mistakes}(w^*)$ and even exceeds $\text{mistakes}(w)$ for some other $w \in \mathcal{W}'$ where $\alpha_v \neq \alpha^*$. Consequently, v may be excluded from $\mathcal{W}_{\min}^2$, potentially leading to $\text{PO}(\mathbf{A}, \mathcal{W}_{\min}^2) = \alpha_w$, excluding α^* and resulting in an incorrect recommendation by the system.

Fortunately, this situation is relatively uncommon, especially when $\mathcal{W}'$ contains a large number of preference points. With a more extensive set, there is likely to be a vector in $\mathcal{W}'$ that closely approximates w^* , thereby maintaining similar ordering preferences and mitigating this issue. Despite these limitations, as we will demonstrate, the algorithm still achieves a high level of accuracy, effectively identifying the user's most preferred alternative in the vast majority of cases.

3.5 Experimental Results

In this section, we discuss the results obtained from experimental tests of our approach on randomly generated decision-making problems.

For each test case, a random problem is created using a randomly generated set A of possibly optimal utility vectors. In these experiments, we simulate a decision-maker with a utility function $u(\alpha, w^*)$, aiming to identify the alternative that the decision-maker would most prefer, i.e., we seek the alternative $\alpha_{w^*} \in A$ that maximizes $u(\alpha, w^*)$ for any $\alpha \in A$.

To simulate the decision-making process, we generate a random weights vector w^* for each test case to represent the decision-maker's preferences. In an ideal noise-free scenario, the simulated decision-maker's response to a comparison query between two alternatives (α, β) would select α if $w^* \cdot \alpha \geq w^* \cdot \beta$, and β otherwise. However, to introduce the element of noisy user response, we factor in a fixed probability ρ (e.g., $\rho = 0.1$) of receiving an incorrect answer from the simulated decision-maker.

Let $\text{PO}(A, \mathcal{W})$ denote the set of potentially optimal alternatives in A with respect to $\mathcal{W}$. To generate random sets consisting of n alternatives, we begin with an empty set, $A = \emptyset$ and proceed as follows.

1. Generate a random vector $\alpha = (\alpha_1, \dots, \alpha_p)$ with $\alpha_i \geq 0$ and $\sum \alpha_i = 1$.
2. $A = A \cup \{\alpha\}$ if $\text{PO}(A \cup \{\alpha\}, \mathcal{W}) = A \cup \{\alpha\}$.
3. Repeat until $|A| = n$.

Given a random user with weights vector w , the most preferred alternative is defined as $\alpha_w = \arg \max_{\alpha \in A} u(\alpha, w)$. However, as highlighted in Section 3.4, since our approach utilizes only a finite subset $\mathcal{W}' \subseteq \mathcal{W}$ of weights vectors, it is possible that we may not be able to recommend $\alpha^* (= \alpha_{w^*})$ since there may not be $w \in \mathcal{W}'$ such that $\alpha^* = \arg \max_{\alpha \in A} u(\alpha, w)$.

In Table 3.1, we present the average number of queries, the average iteration time, and the accuracy achieved during the experiments. Accuracy is defined as the proportion of experiments in which the recommended alternative matched the correct one, i.e., the optimal alternative α_{w^*} in A according to the true, but unknown, user model w^* . The reported results are based on the average of 100 experiments conducted with random sets $\mathcal{W}'$ of 4000 weight vectors and input sets A containing 1000 randomly generated potentially optimal alternatives. For consistency, the same set A of alternatives was used across all experiments, along with 100 different randomly generated user models w^* . The accuracy was consistently high, and in all cases, there was at least one $w \in \mathcal{W}'$ such that $\alpha_w = \alpha_{w^*}$.

Table 3.1: Experimental results w.r.t. the number of criteria p , for **Method 1: Max Preference Points for Query Selection** with $|A| = 1000$, $\rho = 0.1$ and $|\mathcal{W}'| = 4000$.

p	Queries	Time[s]	Accuracy
3	10.66	0.038	1.00
4	22.16	0.038	1.00
5	30.44	0.038	0.97
6	35.88	0.038	1.00

Iteration Time Analysis. For **Method 1: Max Preference Points for Query Selection**, shown in Table 3.1, iteration time appears to be largely independent of the number of criteria. This is because the most computationally expensive operation in Method 1 involves updating the values of $mistakes(w)$ for each $w \in \mathcal{W}$, which depends primarily on the size of $\mathcal{W}$ rather than the number of criteria. The primary computational load comes from calculating $|\mathcal{W}'|$ dot products, which remains relatively stable regardless of the number of criteria. For instance, when $p = 5$, the average computation time per iteration was as follows: 0.34ms for query selection, 36.2ms for updating $mistakes(w)$, and 1.05ms for computing the three sets $\mathcal{W}_{min}^k$, $k = 0, 1, 2$.

In contrast, the iteration time for **Method 2: Best Split for Query Selection**, presented in Table 3.2, is more significantly impacted by the number of criteria. The computational cost of query evaluation in this method depends on the size of the subsets $\mathcal{W}_{min}^k$, which tends to increase as the number of criteria grows. For example, when $p = 5$, the average iteration times for Method 2 were as follows: 2.53s to compute a query, 35.6ms to update $mistakes(w)$ for each $w \in \mathcal{W}$, and 1ms to compute the sets $\mathcal{W}_{min}^k$.

Table 3.2: Experimental results for **Method 2: Best Split for Query Selection**; $|A| = 1000$, $\rho = 0.1$ and $|\mathcal{W}'| = 4000$.

p	Queries	Time[s]	Accuracy
3	11.55	0.289	1.00
4	17.55	0.508	0.99
5	25.84	2.574	1.00
6	28.42	4.789	1.00

The results for Method 2, presented in Table 3.2, demonstrate similar trends to those of the first method. However, Method 2 typically requires fewer queries on average to converge, albeit at the cost of significantly longer computation times per iteration compared to the first method.

In general, the number of queries required increases with the number of criteria. This is expected because identifying a subset $\mathcal{W}_{min}^2$ of $\mathcal{W}'$, where all preference points correspond to the same optimal alternative, becomes more challenging as the dimensionality grows. Additionally, for preference points corresponding to the same optimal alternative, there may be variations in preference orders over the remaining alternatives (see Section 3.3). This phenomenon becomes more pronounced as the number of dimensions increases, necessitating more queries to meet the stopping criterion.

For comparison, we present the simulation results obtained using state-of-the-art elicitation methods applied to the same datasets. Table 3.3 summarizes the performance of interactive elicitation approach based on the minimax regret criterion, with queries generated using existing solution strategies (these results were obtained by our collaborator Paolo Viappiani. Similarly for Table 3.4 results). As expected, regret-based elicitation demonstrates lower accuracy, primarily due to its inability to effectively handle noise in user responses.

Table 3.3: Experimental results for regret-based elicitation with the current solution query strategy; $|A| = 1000$ and $\rho = 0.1$.

p	Queries	Time[s]	Accuracy
3	4.58	4.12	0.60
4	7.82	3.41	0.34
5	11.93	3.66	0.34
6	16.69	3.69	0.28

In addition to the first comparison method, table 3.4 presents results for the Bayesian elicitation method described in [VB20]. In this approach, queries are selected to maximize the Expected Utility of Selection (EUS), which serves as a proxy for the myopic value of information, using a greedy maximization strategy. Bayesian updates are carried out using Monte Carlo methods with 50,000 particles. The elicitation process terminates when the expected loss falls

below 0.001, at which point the alternative with the highest expected utility is recommended.

As indicated in the table 3.4, the Bayesian method achieves high accuracy. However, this accuracy comes at the cost of significantly longer computation times. While increasing the number of particles could yield even higher accuracy, it would also lead to further increases in computational effort.

Table 3.4: Experimental results for Bayesian elicitation with the queries generated with greedy maximisation of value of information; $|A| = 1000$ and $\rho = 0.1$.

p	Queries	Time[s]	Accuracy
3	12.05	2.62	0.97
4	17.93	3.68	0.99
5	26.46	5.07	0.97
6	35.86	5.50	1.00

Comparative Performance Visualization

To enable a clearer comparison of all tested algorithms, below we present visual summaries of the experimental results in Tables 3.1 – 3.4.

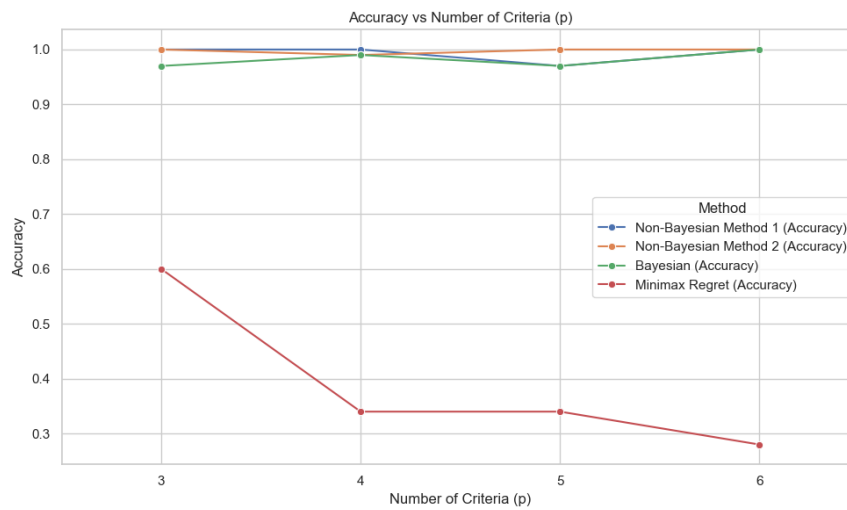


Figure 3.1: Accuracy vs Number of Criteria p across all methods.

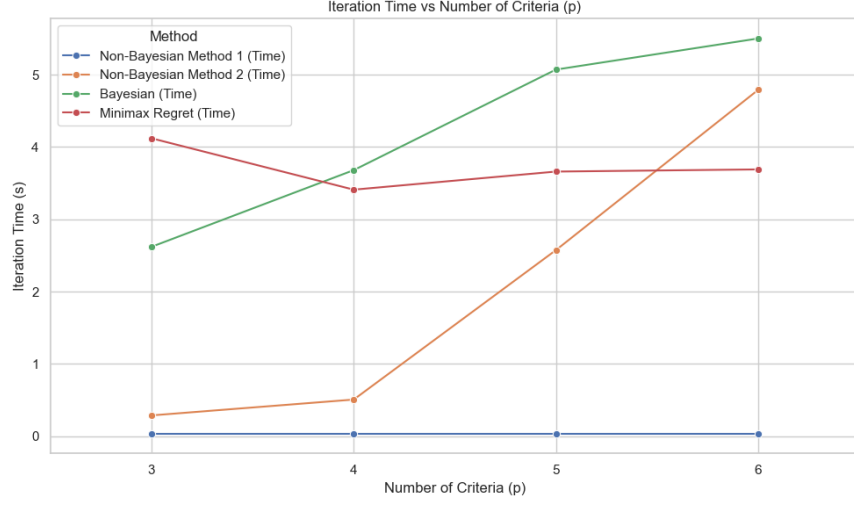


Figure 3.2: Iteration Time vs Number of Criteria p across all methods.

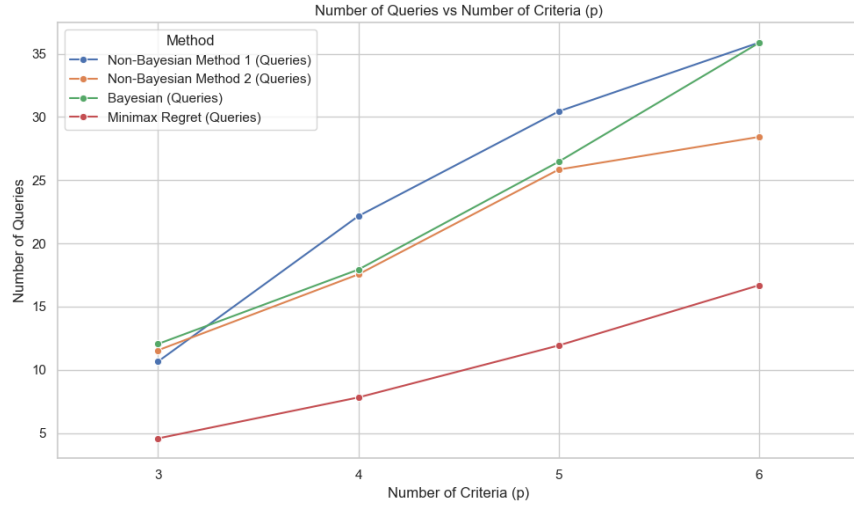


Figure 3.3: Number of Queries vs Number of Criteria p across all methods.

Figures 3.1, 3.2, and 3.3 highlight the comparative performance of the proposed non-Bayesian methods (Method 1 and 2) against the Bayesian and minimax regret-based approaches. While the Bayesian method achieves similar accuracy, its computational cost is significantly higher. The non-Bayesian approach maintains strong accuracy with minimal execution time, especially Method 1. The minimax regret method, although fast in terms of query count, underperforms in accuracy under noisy conditions.

We now return to our proposed approach, specifically focusing on Method 1 for query generation. Table 3.5 presents the performance of our method with respect to the size of $\mathcal{W}'$. As expected, accuracy improves as $|\mathcal{W}'|$ increases.

However, this comes with a corresponding rise in computation time and, to a lesser degree, an increase in the average number of queries required. The experimental results below refer to Method 1, which is the fastest and thus probably the most suitable for real-time applications.

Table 3.5: Experimental results w.r.t. the number $|\mathcal{W}'|$ of user preference models, for Method 1 with $|A| = 1000$, $\rho = 0.1$ and $p = 5$.

$ \mathcal{W}' $	Queries	Time[s]	Accuracy
2000	27.28	0.019	0.97
4000	30.44	0.038	0.97
6000	32.71	0.057	1.00
8000	33.12	0.075	0.99
10000	34.50	0.096	1.00

We also evaluated the performance of our approach with varying numbers of alternatives, $|A| \in \{200, 400, 600, 800, 1000\}$, while keeping $|\mathcal{W}'| = 4000$, $p = 4$ and $\rho = 0.1$. The results showed no significant differences in terms of accuracy or execution time across the tested values of $|A|$. The average number of queries was affected slightly more, ranging between 20.67 and 22.16, with fewer queries required for smaller $|A|$.

For an efficient implementation of Method 1, it is beneficial to precompute the best alternative α_w for each $w \in \mathcal{W}'$. This preprocessing step significantly accelerates the calculation of $\text{Opt}^k(\alpha)$ at each iteration. The complexity of this operation is proportional to $|A| * |\mathcal{W}'|$, but it only needs to be performed once for a given pair A and $\mathcal{W}'$. This allows the preprocessing to be done offline, with the results reusable for any decision-maker.

Figure 3.4 illustrates the average preprocessing time (in seconds) as a function of the cardinality $|\mathcal{W}'|$, based on 100 experiments with $|A| = 1000$ and $p = 5$. The preprocessing time is largely unaffected by the number of criteria, as confirmed by additional experiments with $p \in \{3, 4, 5, 6\}$ where the maximum observed difference was only 0.5 seconds on average.

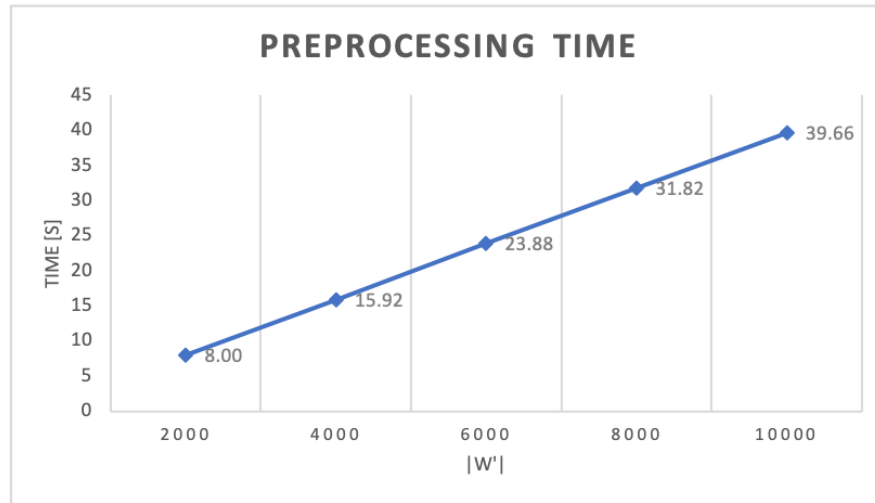


Figure 3.4: Average preprocessing time for Method 1 over 100 experiments with $|A| = 1000$ and $p = 5$.

Figure 3.5 shows how accuracy varies with respect to user noise ρ , with $|A| = 1000$, $|W'| = 10000$ and $p = 4$. As expected, accuracy decreases as the level of user noise ρ increases. However, the results demonstrate that our model maintains good performance even under noisier responses. In this experiment, the average query time was 0.093 seconds. The observed drop in accuracy for higher values of ρ aligns with our analysis in Proposition 1. To sustain very high accuracy under increased noise levels, it would be necessary to adjust the parameter κ in our algorithm (currently set to 2), thereby making the stopping condition more stringent and reducing the impact of noisy responses on the final recommendation.

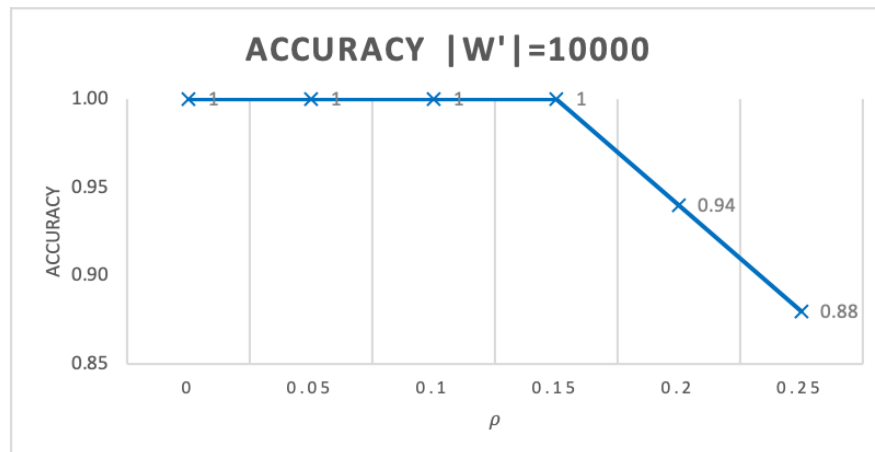


Figure 3.5: Accuracy varying with user noise ρ over 100 experiments with $|A| = 1000$, $|W'| = 10000$ and $p = 4$.

Chapter 4

Conclusions & Discussion

This thesis has introduced a non-Bayesian approach to interactive preference elicitation, designed to function efficiently in decision-making environments where user responses may be noisy or unreliable. Unlike traditional Bayesian methods that require explicit probability distributions over user preferences, this approach instead maintains and refines a set of plausible preference models, determining optimal alternatives based on their alignment with observed user responses. By generating targeted comparison queries, the model incrementally gathers preference information, ensuring both computational efficiency and robustness.

A fundamental component of this approach is its adaptive stopping criterion. The interaction process concludes and an alternative α is recommended when α is optimal across all the most plausible preference models. In this framework, plausibility is defined based on how closely the preference models align with actual user responses. To achieve computational feasibility, this work focuses on a finite approximation of the space of preference models, denoted as $\mathcal{W}'$, which enables efficient real-time decision-making.

Empirical validation of this method demonstrates its suitability for real-time applications, achieving strong accuracy while requiring only a reasonable number of user interactions. A comparison with the Bayesian method proposed by Viappiani and Boutilier [VB20] shows that the non-Bayesian approach significantly improves computational speed while achieving comparable or slightly better accuracy, making it an effective alternative where efficiency is a key concern.

Additionally, the stopping criterion used in this work—which ensures that

the maximum regret of the recommended alternative is zero across all most plausible preference models—presents an interesting opportunity for further refinement. Future work could explore a relaxed criterion, where the minimum max regret falls below a small threshold ϵ . This adaptation could enhance applicability in complex decision scenarios, reducing the number of interactions required while still maintaining high recommendation quality.

4.1 Summary of Contributions

This research makes several important contributions to interactive preference elicitation and decision-support systems, particularly within the realm of artificial intelligence and multi-criteria decision-making.

- **Development of a Non-Bayesian Framework for Preference Elicitation:** Unlike Bayesian approaches, which assume a prior probability distribution over user preferences, this model takes a qualitative approach, focusing on the most plausible preference models rather than maintaining probability distributions.
- **Efficient and Scalable Decision-Making:** By avoiding computationally expensive Bayesian updates, this framework offers a computationally lightweight alternative that scales effectively, making it viable for real-time applications.
- **Adaptive Query Selection for Noisy Responses:** The proposed query selection mechanism dynamically chooses the most informative comparisons, ensuring that user responses contribute meaningfully to refining the preference model while remaining robust to noisy or inconsistent feedback.
- **Empirical Evaluation Against Bayesian Methods:** Experimental results demonstrate that the non-Bayesian approach performs comparably or better than Bayesian models, with a significant advantage in computational efficiency. This highlights its potential as a practical alternative for interactive AI-driven decision systems.
- **Potential Applications Beyond Single-Agent Decision Making:** While this thesis focuses on single-user decision-making, the framework is adaptable to multi-agent settings, interactive recommender systems, and combinatorial decision-making problems.

4.2 Discussion and Future Research Directions

Although this research introduces a computationally efficient and scalable alternative to Bayesian preference elicitation, several promising directions for future work remain.

- **Extending to Multi-Agent and Collaborative Preference Models:** The current framework focuses on eliciting preferences from a single user, but in real-world decision-making, preferences often involve multiple stakeholders. Future research could explore how this non-Bayesian approach can be adapted to multi-agent decision-making, incorporating methods for aggregating, reconciling, or negotiating preferences across multiple users.
- **Integration with Conversational Recommender Systems:** As interactive AI systems evolve, Conversational Recommender Systems provide an exciting opportunity for integrating non-Bayesian preference elicitation. By incorporating natural language processing and reinforcement learning, future iterations of this model could enable real-time, dialogue-based preference refinement, where users express trade-offs and evolving preferences through natural conversation.
- **Hybrid Models Combining Bayesian and Non-Bayesian Approaches:** While this thesis demonstrates the advantages of a non-Bayesian framework, certain decision-making scenarios might still benefit from elements of Bayesian reasoning. A hybrid approach could combine Bayesian probability updates with non-Bayesian plausibility filtering, selectively applying Bayesian inference where it enhances accuracy without adding excessive computational cost.
- **Scaling to Combinatorial Decision Problems:** The current model efficiently handles finite alternative sets, but many decision-making problems involve combinatorial complexity, where the number of alternatives grows exponentially. Future work could explore heuristic optimization techniques, constraint satisfaction methods, or reinforcement learning to extend the scalability of non-Bayesian preference elicitation in these settings.
- **Refining the Stopping Criterion for Broader Applications:** The current stopping criterion, based on ensuring that the maximum regret of the

selected alternative is zero, could be modified for greater flexibility. Introducing a threshold-based stopping criterion (ϵ -max regret) could reduce the number of interactions required while still maintaining a high confidence in the recommended alternative.

4.3 Limitations of the Work

Although the methods proposed in this thesis demonstrate promising performance under controlled conditions, several limitations should be acknowledged.

- **Simulated evaluations only:** A key limitation of this thesis is that the evaluation of the proposed preference elicitation methods has been conducted exclusively in simulated environments using randomly generated decision-making problems. While this approach ensures experimental control and reproducibility, it does not fully account for the complexities of real-world decision-making, particularly when human cognitive factors are involved. Human users may exhibit preference inconsistencies, fatigue during extended elicitation, or find certain types of queries more intuitive or cognitively demanding than others.

To address this gap, future research should incorporate empirical evaluations involving real users. Such studies would not only test the algorithm's performance under realistic conditions but also provide insights into usability, interpretability, and human preference behavior that cannot be captured in simulation alone.

One promising direction is to design user studies modeled after the work of Zintgraf et al. [ZRL⁺18], who compared multiple preference elicitation strategies (pairwise comparisons, full rankings, and clustering) through both simulation and human-subject experiments. Their study used a job-offer selection scenario and additionally tested the methods in a real-world traffic planning application in collaboration with the City of Amsterdam. Their results indicated that full rankings often led to better utility approximations and were generally preferred by users, despite requiring more effort.

Adopting similar experimental frameworks would allow a more comprehensive evaluation of the methods proposed in this thesis. For instance,

controlled user studies could be designed around application domains highlighted in Chapter 2, such as recommender systems, healthcare decision aids, or financial planning tools. These domains naturally involve trade-offs between multiple objectives and require efficient preference elicitation from users who may not be familiar with formal decision models.

Such studies should include both quantitative metrics, e.g. the number of queries needed to converge on a high-utility recommendation, time per query, and final utility achieved, and qualitative assessments, such as user satisfaction, perceived cognitive load, and willingness to continue the interaction. Additional methodologies like think-aloud protocols or post-task interviews could uncover users' mental models and reveal potential barriers to adoption.

Moreover, testing with real users can help evaluate the practical implications of key modeling assumptions, such as the linearity and monotonicity of utility functions. In situations where these assumptions fail to hold, user feedback could inform the design of adaptive or hybrid elicitation strategies that better match real-world preferences.

- **Assumption of linear utility functions:** The current framework assumes that user preferences can be expressed as a linear weighted sum of attribute values. While this simplifies regret computation and supports tractable reasoning, it may not generalize well to contexts with non-linear trade-offs or more expressive preference models. Extending the approach to support non-linear utility functions such as polynomials, piecewise linear models, or GAI models would expand its applicability, though it would likely require modifications to the regret and plausibility inference mechanisms.
- **Sample-based utility representation:** The algorithm operates by sampling a set of plausible utility vectors to approximate the space of possible user preferences. While effective in low- to moderate-dimensional problems, this introduces approximation error and may lead to suboptimal decisions in higher dimensions or when the true utility lies in sparsely sampled regions.
- **User feedback modeled in limited form:** While simulated user responses include random noise via a fixed error rate ρ , this approach

assumes uniform, memoryless inaccuracies. In real-world applications, user errors may depend on factors such as cognitive load, query difficulty, or interface design, and may follow systematic patterns (e.g., satisficing, anchoring). Additionally, the current framework uses crisp updates to eliminate utility vectors, without modeling uncertainty or belief over user feedback. Future work could extend the framework to handle richer noise models, adaptive querying, and error-tolerant preference learning mechanisms.

4.4 Potential Impact on the Field

The proposed plausibility-based preference elicitation framework offers a lightweight and interpretable alternative to existing probabilistic models. By avoiding full Bayesian inference and instead leveraging sampled utility representations with regret and plausibility reasoning, the approach may be especially suitable for systems where transparency, efficiency, and responsiveness are key.

Potential areas of impact include:

- Interactive decision-support systems, where rapid, intelligible feedback is essential to maintain user engagement.
- Resource-constrained environments, where full probabilistic modeling may be infeasible or unwarranted.
- Embedded AI applications, such as adaptive user interfaces or intelligent assistants, that require fast and robust preference estimation.

In addition to technical contributions, this work helps advance the broader research discourse around preference elicitation by demonstrating that non-Bayesian, sample-based methods can achieve competitive performance. This paves the way for hybrid models that combine plausibility reasoning with machine learning techniques to balance tractability and expressiveness.

Furthermore, the approach is compatible with multiple query types, including pairwise comparisons and partial rankings, making it adaptable to diverse application settings. It also lends itself well to integration with user-centric evaluation frameworks, enhancing its relevance for human-AI collaborative systems in domains such as recommender systems, health technology, and multi-objective planning.

4.5 Final Remarks

This research contributes to the advancement of preference elicitation techniques by proposing a computationally efficient, non-Bayesian approach that overcomes many of the limitations of probabilistic models. By shifting from probability-based reasoning to plausibility-based refinement, this work offers a scalable, adaptive, and robust alternative for real-time decision-making applications.

The experimental findings demonstrate that this method achieves comparable or superior accuracy to Bayesian approaches while significantly improving computational speed, making it highly relevant for AI-driven recommendation systems, interactive decision-support tools, and multi-criteria optimization.

Moving forward, integrating this framework into Conversational AI, multi-agent systems, and large-scale decision environments represents a promising avenue for enhancing AI-driven personalization and user-centric decision-making. By further refining the balance between efficiency, accuracy, and adaptability, this work lays the foundation for future developments in interactive AI-driven preference elicitation.

Ultimately, this research highlights the importance of adaptive, user-driven AI models, positioning non-Bayesian preference elicitation as a practical and scalable alternative for intelligent decision-support systems.

References

- [Agg16] Charu C. Aggarwal. *Recommender Systems: The Textbook*. Springer International Publishing, 2016.
- [Arr12] Kenneth J. Arrow. *Social Choice and Individual Values*. Yale University Press, 2012.
- [AT05] Gediminas Adomavicius and Alexander Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Trans. Knowl. Data Eng.*, 17(6):734–749, 2005.
- [BB06] Darius Braziunas and Craig Boutilier. Preference elicitation and generalized additive utility. In *Proceedings, The Twenty-First National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference, July 16-20, 2006, Boston, Massachusetts, USA*, pages 1573–1576. AAAI Press, 2006.
- [BB10] Darius Braziunas and Craig Boutilier. Assessing regret-based preference elicitation with the UTPREF recommendation system. In David C. Parkes, Chrysanthos Dellarocas, and Moshe Tennenholtz, editors, *Proceedings 11th ACM Conference on Electronic Commerce (EC-2010), Cambridge, Massachusetts, USA, June 7-11, 2010*, pages 219–228. ACM, 2010.
- [BB12] Darius Braziunas and Craig Boutilier. Minimax regret based elicitation of generalized additive utilities. *CoRR*, abs/1206.5255, 2012.
- [BC19] Silvia Bacci and Bruno Chiandotto. *Introduction to Statistical Decision Theory: Utility Theory and Causal Analysis*. CRC Press, 2019.

- [BDPV16] Nawal Benabbou, Serena Di Sabatino Di Diodoro, Patrice Perny, and Paolo Viappiani. Incremental preference elicitation in multi-attribute domains for choice and ranking with the borda count. In Steven Schockaert and Pierre Senellart, editors, *Scalable Uncertainty Management - 10th International Conference, SUM 2016, Nice, France, September 21-23, 2016, Proceedings*, volume 9858 of *Lecture Notes in Computer Science*, pages 81–95. Springer, 2016.
- [BG15] Tim Baarslag and Enrico H. Gerding. Optimal incremental preference elicitation during negotiation. In Qiang Yang and Michael J. Wooldridge, editors, *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, pages 3–9. AAAI Press, 2015.
- [BGMS05] Derek G. Bridge, Mehmet H. Göker, Lorraine McGinty, and Barry Smyth. Case-based recommender systems. *Knowl. Eng. Rev.*, 20(3):315–320, 2005.
- [BGPV15] Nawal Benabbou, Christophe Gonzales, Patrice Perny, and Paolo Viappiani. Minimax regret approaches for preference elicitation with rank-dependent aggregators. *Eur. J. Deci. Proc.*, 3(1):29–64, 2015.
- [Bin18] Reuben Binns. Fairness in machine learning: Lessons from political philosophy. In Sorelle A. Friedler and Christo Wilson, editors, *Conference on Fairness, Accountability and Transparency, FAT 2018, 23-24 February 2018, New York, NY, USA*, volume 81 of *Proceedings of Machine Learning Research*, pages 149–159. PMLR, 2018.
- [Bis16] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Information Science and Statistics. Springer New York, 2016.
- [BKN07] Peter Brusilovsky, Alfred Kobsa, and Wolfgang Nejdl. *The Adaptive Web: Methods and Strategies of Web Personalization*. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2007.
- [BL19] Nawal Benabbou and Thibaut Lust. An interactive polyhedral approach for multi-objective combinatorial optimization with incomplete preference information. In Nahla Ben Amor, Benjamin Quost, and Martin Theobald, editors, *Scalable Uncertainty Management - 13th International Conference, SUM 2019, Compiègne, France, De-*

- ember 16-18, 2019, *Proceedings*, volume 11940 of *Lecture Notes in Computer Science*, pages 221–235. Springer, 2019.
- [BLL20] Nawal Benabbou, Cassandre Leroy, and Thibaut Lust. An interactive regret-based genetic algorithm for solving multi-objective combinatorial optimization problems. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 2335–2342. AAAI Press, 2020.
- [Bly02] Jim Blythe. Visual exploration and incremental utility elicitation. In Rina Dechter, Michael J. Kearns, and Richard S. Sutton, editors, *Proceedings of the Eighteenth National Conference on Artificial Intelligence and Fourteenth Conference on Innovative Applications of Artificial Intelligence, July 28 - August 1, 2002, Edmonton, Alberta, Canada*, pages 526–532. AAAI Press / The MIT Press, 2002.
- [BMP⁺06] Denis Bouyssou, Thierry Marchant, Marc Pirlot, Alexis Tsoukias, and Philippe Vincke. *Evaluation and Decision Models with Multiple Criteria: Stepping stones for the analyst*. International Series in Operations Research & Management Science. Springer US, 2006.
- [Bou02] Craig Boutilier. A POMDP formulation of preference elicitation problems. In Rina Dechter, Michael J. Kearns, and Richard S. Sutton, editors, *Proceedings of the Eighteenth National Conference on Artificial Intelligence and Fourteenth Conference on Innovative Applications of Artificial Intelligence, July 28 - August 1, 2002, Edmonton, Alberta, Canada*, pages 239–246. AAAI Press / The MIT Press, 2002.
- [Bou13] Craig Boutilier. Computational decision support regret-based models for optimization and preference elicitation. In *Comparative Decision Making*. Oxford University Press, 2013.
- [BP15] Nawal Benabbou and Patrice Perny. Incremental weight elicitation for multiobjective state space search. In Blai Bonet and Sven Koenig, editors, *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA*, pages 1093–1099. AAAI Press, 2015.

- [BP18] Nawal Benabbou and Patrice Perny. Interactive resolution of multiobjective combinatorial optimization problems by incremental elicitation of criteria weights. *Eur. J. Deci. Proc.*, 6(3):283–319, 2018.
- [BPPS06] Craig Boutilier, Relu Patrascu, Pascal Poupart, and Dale Schuurmans. Constraint-based optimization and utility elicitation using the minimax decision criterion. *Artif. Intell.*, 170(8-9):686–713, 2006.
- [BPS20] Nadjet Bourdache, Patrice Perny, and Olivier Spanjaard. Bayesian preference elicitation for multiobjective combinatorial optimization. *CoRR*, abs/2007.14778, 2020.
- [BPV17] Nawal Benabbou, Patrice Perny, and Paolo Viappiani. Incremental elicitation of choquet capacities for multicriteria choice, ranking and sorting problems. *Artif. Intell.*, 246:152–180, 2017.
- [BR07] Derek G. Bridge and Francesco Ricci. Supporting product selection with query editing recommendations. In Joseph A. Konstan, John Riedl, and Barry Smyth, editors, *Proceedings of the 2007 ACM Conference on Recommender Systems, RecSys 2007, Minneapolis, MN, USA, October 19-20, 2007*, pages 65–72. ACM, 2007.
- [Bur02] Robin D. Burke. Hybrid recommender systems: survey and experiments. *User Model. User Adapt. Interact.*, 12(4):331–370, 2002.
- [BZM12] Craig Boutilier, Richard S. Zemel, and Benjamin M. Marlin. Active collaborative filtering. *CoRR*, abs/1212.2442, 2012.
- [BZWM15] Jie Bao, Yu Zheng, David Wilkie, and Mohamed F. Mokbel. Recommendations in location-based social networks: A survey. *GeoInformatica*, 19(3):525–565, 2015.
- [CGNS13] Urszula Chajewska, Lise Getoor, Joseph Norman, and Yuval Shahar. Utility elicitation as a classification problem. *CoRR*, abs/1301.7367, 2013.
- [CKC05] Jongwoo Chae, Jungkyu Kwon, and Mokdong Chung. Computer security management model using MAUT and SNMP. In Osvaldo Gervasi, Marina L. Gavrilova, Vipin Kumar, Antonio Lagana, Heow Pueh Lee, Youngsong Mun, David Taniar, and Chih

- Jeng Kenneth Tan, editors, *Computational Science and Its Applications - ICCSA 2005, International Conference, Singapore, May 9-12, 2005, Proceedings, Part II*, volume 3481 of *Lecture Notes in Computer Science*, pages 147–156. Springer, 2005.
- [CKM⁺22] Marco Cinelli, Milosz Kadzinski, Grzegorz Miebs, Michael Gonzalez, and Roman Slowinski. Recommending multiple criteria decision analysis methods with a new taxonomy-based decision support system. *Eur. J. Oper. Res.*, 302(2):633–651, 2022.
- [CKP00] Urszula Chajewska, Daphne Koller, and Ronald Parr. Making rational decisions using adaptive utility elicitation. In Henry A. Kautz and Bruce W. Porter, editors, *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence, July 30 - August 3, 2000, Austin, Texas, USA*, pages 363–369. AAAI Press / The MIT Press, 2000.
- [DFGZ19] Michalis Doumpos, José Rui Figueira, Salvatore Greco, and Constantin Zopounidis. *New Perspectives in Multiple Criteria Decision Making: Innovative Applications and Case Studies*. Multiple Criteria Decision Making. Springer, 2019.
- [DH83] Gerard Debreu and Werner Hildenbrand. *Topological Methods in Cardinal Utility Theory*, page 120–132. Econometric Society Monographs. Cambridge University Press, 1983.
- [DNNR18] Amra Delic, Julia Neidhardt, Thuy Ngoc Nguyen, and Francesco Ricci. An observational user study for group recommender systems in the tourism domain. *J. Inf. Technol. Tour.*, 19(1-4):87–116, 2018.
- [Dol10] James G. Dolan. Multi-criteria clinical decision support: A primer on the use of multiple criteria decision making methods to promote evidence-based, patient-centered healthcare. *Patient*, 3(4):229–248, 2010.
- [DTP18] Paolo Dragone, Stefano Teso, and Andrea Passerini. Constructive preference elicitation over hybrid combinatorial spaces. In Sheila A. McIlraith and Kilian Q. Weinberger, editors, *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence*

- (IAAI-18), and the 8th AAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018, pages 2943–2950. AAAI Press, 2018.
- [ERK11] Michael D. Ekstrand, John Riedl, and Joseph A. Konstan. Collaborative filtering recommender systems. *Found. Trends Hum. Comput. Interact.*, 4(2):175–243, 2011.
- [ERR13] Mehdi Elahi, Francesco Ricci, and Neil Rubens. Active learning strategies for rating elicitation in collaborative filtering: A system-wide perspective. *ACM Trans. Intell. Syst. Technol.*, 5(1):13:1–13:33, 2013.
- [FH10] Johannes Fürnkranz and Eyke Hüllermeier. *Preference Learning*. Springer Berlin Heidelberg, 2010.
- [Fin72] Terrence L. Fine. Review of ‘utility theory for decision making’ (fishburn, peter c.; 1970). *IEEE Trans. Inf. Theory*, 18(5):696–697, 1972.
- [Fis67] Peter C. Fishburn. Methods of estimating additive utilities. *Manag. Sci.*, 13(7):435–453, 1967.
- [Fis79] Peter C. Fishburn. *Utility Theory for Decision Making*. Publications in operations research. Krieger, 1979.
- [FSST17] Piotr Faliszewski, Piotr Skowron, Arkadii Slinko, and Nimrod Talmon. Multiwinner voting: A new challenge for social choice theory. *Trends Comput. Soc. Choice*, 74:27–47, 2017.
- [GCS⁺13] Andrew Gelman, John B. Carlin, Hal S. Stern, David B. Dunson, Aki Vehtari, and Donald B. Rubin. *Bayesian Data Analysis, Third Edition*. Chapman & Hall/CRC Texts in Statistical Science. Taylor & Francis, 2013.
- [GEF10] Salvatore Greco, Matthias Ehrgott, and José Rui Figueira. *Trends in Multiple Criteria Decision Analysis*. International Series in Operations Research & Management Science. Springer US, 2010.
- [GEF16] Salvatore Greco, Matthias Ehrgott, and José Rui Figueira. *Multiple Criteria Decision Analysis: State of the Art Surveys*. International Series in Operations Research & Management Science. Springer New York, 2016.

- [Ger98] Alex E. Gerodimos. Robust discrete optimization and its applications. *J. Oper. Res. Soc.*, 49(12):1303–1304, 1998.
- [GGL18] Jianfeng Gao, Michel Galley, and Lihong Li. Neural approaches to conversational AI. In Kevyn Collins-Thompson, Qiaozhu Mei, Brian D. Davison, Yiqun Liu, and Emine Yilmaz, editors, *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08-12, 2018*, pages 1371–1374. ACM, 2018.
- [GK03] Soumyadip Ghosh and Jayant Kalagnanam. Polyhedral sampling for multiattribute preference elicitation. In Daniel A. Menascé and Noam Nisan, editors, *Proceedings 4th ACM Conference on Electronic Commerce (EC-2003), San Diego, California, USA, June 9-12, 2003*, pages 256–257. ACM, 2003.
- [GLH⁺21] Chongming Gao, Wenqiang Lei, Xiangnan He, Maarten de Rijke, and Tat-Seng Chua. Advances and challenges in conversational recommender systems: A survey. *AI Open*, 2:100–126, 2021.
- [GMS14] Salvatore Greco, Vincent Mousseau, and Roman Slowinski. Robust ordinal regression for value functions handling interacting criteria. *Eur. J. Oper. Res.*, 239(3):711–730, 2014.
- [GP04] Christophe Gonzales and Patrice Perny. GAI networks for utility elicitation. In Didier Dubois, Christopher A. Welty, and Mary-Anne Williams, editors, *Principles of Knowledge Representation and Reasoning: Proceedings of the Ninth International Conference (KR2004), Whistler, Canada, June 2-5, 2004*, pages 224–234. AAAI Press, 2004.
- [GPS10] Lucie Galand, Patrice Perny, and Olivier Spanjaard. Choquet-based optimisation in multiobjective shortest path and spanning tree problems. *Eur. J. Oper. Res.*, 204(2):303–315, 2010.
- [GS10] Shengbo Guo and Scott Sanner. Real-time multiattribute bayesian preference elicitation with pairwise comparison queries. In Yee Whye Teh and D. Mike Titterton, editors, *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2010, Chia Laguna Resort, Sardinia, Italy, May*

- 13-15, 2010, volume 9 of *JMLR Proceedings*, pages 289–296. JMLR.org, 2010.
- [GSFM10] Salvatore Greco, Roman Slowinski, José Rui Figueira, and Vincent Mousseau. Robust ordinal regression. In Matthias Ehrgott, José Rui Figueira, and Salvatore Greco, editors, *Trends in Multiple Criteria Decision Analysis*, pages 241–283. Springer, 2010.
- [GSH13] Tomas Gal, Theodor J. Stewart, and Thomas Hanne. *Multicriteria Decision Making: Advances in MCDM Models, Algorithms, Theory, and Applications*. International Series in Operations Research & Management Science. Springer US, 2013.
- [Haz86] Gordon B. Hazen. Partial information, dominance, and potential optimality in multiattribute utility theory. *Oper. Res.*, 34(2):296–310, 1986.
- [HBWP12] Matthew D. Hoffman, David M. Blei, Chong Wang, and John W. Paisley. Stochastic variational inference. *CoRR*, abs/1206.7051, 2012.
- [HKBR17] Jonathan L. Herlocker, Joseph A. Konstan, Al Borchers, and John Riedl. An algorithmic framework for performing collaborative filtering. *SIGIR Forum*, 51(2):227–234, 2017.
- [ILC01] Vijay S. Iyengar, Jon Lee, and Murray Campbell. Evaluating multiple attribute items using queries. In Michael P. Wellman and Yoav Shoham, editors, *Proceedings 3rd ACM Conference on Electronic Commerce (EC-2001)*, Tampa, Florida, USA, October 14-17, 2001, pages 144–153. ACM, 2001.
- [IN13] Alessio Ishizaka and Philippe Nemery. *Multi-criteria Decision Analysis: Methods and Software*. Wiley, 2013.
- [ISD84] Chelsea C. White III, Andrew P. Sage, and Shigeru Dozono. A model of multiattribute decisionmaking and trade-off weight determination under uncertainty. *IEEE Trans. Syst. Man Cybern.*, 14(2):223–229, 1984.
- [Kli92] Ronald K. Klimberg. Grads: A new graphical display system for visualizing multiple criteria solutions. *Comput. Oper. Res.*, 19(7):707–711, 1992.

- [KM97] Ignacy Kaliszewski and Wojtek Michalowski. Efficient solutions and bounds on tradeoffs. *J. Optim. Theory Appl.*, 94(2):381–394, 1997.
- [Koj07] Ivan Kojadinovic. Minimum variance capacity identification. *Eur. J. Oper. Res.*, 177(1):498–514, 2007.
- [Kor05] Pekka Korhonen. *Interactive Methods*, pages 641–661. Springer New York, New York, NY, 2005.
- [KR93] Ralph L. Keeney and Howard Raiffa. *Decisions with Multiple Objectives: Preferences and Value Trade-Offs*. Wiley series in probability and mathematical statistics. Applied probability and statistics. Cambridge University Press, 1993.
- [KS14] Gokhan Kirlik and Serpil Sayin. A new algorithm for generating all nondominated solutions of multiobjective discrete optimization problems. *Eur. J. Oper. Res.*, 232(3):479–488, 2014.
- [KVVA17] Sami Kaddani, Daniel Vanderpooten, Jean Michel Vanpeperstraete, and Hassene Aissi. Weighted sum model with partial preference information: Application to multi-objective optimization. *Eur. J. Oper. Res.*, 260(2):665–679, 2017.
- [KWZ11] Murat Köksalan, Jyrki Wallenius, and Stanley Zionts. *Multiple Criteria Decision Making*. WORLD SCIENTIFIC, 2011.
- [KY13] Panos Kouvelis and Gang Yu. *Robust Discrete Optimization and Its Applications*. Nonconvex Optimization and Its Applications. Springer US, 2013.
- [LB11a] Tyler Lu and Craig Boutilier. Budgeted social choice: From consensus to personalized decision making. In Toby Walsh, editor, *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, July 16-22, 2011*, pages 280–286. IJCAI/AAAI, 2011.
- [LB11b] Tyler Lu and Craig Boutilier. Robust approximation and incremental elicitation in voting protocols. In Toby Walsh, editor, *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, July 16-22, 2011*, pages 287–293. IJCAI/AAAI, 2011.

- [LdGS11] Pasquale Lops, Marco de Gemmis, and Giovanni Semeraro. Content-based recommender systems: State of the art and trends. In Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul B. Kantor, editors, *Recommender Systems Handbook*, pages 73–105. Springer, 2011.
- [LHZ14] Benedikt Loepp, Tim Hussein, and Jürgen Ziegler. Choice-based preference elicitation for collaborative filtering recommender systems. In Matt Jones, Philippe A. Palanque, Albrecht Schmidt, and Tovi Grossman, editors, *CHI Conference on Human Factors in Computing Systems, CHI'14, Toronto, ON, Canada - April 26 - May 01, 2014*, pages 3085–3094. ACM, 2014.
- [LPK02] Kyung Sang Lee, Kyung Sam Park, and Soung Hie Kim. Dominance, potential optimality, imprecise information, and hierarchical structure in multi-criteria analysis. *Comput. Oper. Res.*, 29(9):1267–1281, 2002.
- [LPM⁺24] Harrison Lee, Samrat Phatale, Hassan Mansoor, Thomas Mesnard, Johan Ferret, Kellie Lu, Colton Bishop, Ethan Hall, Victor Carbune, Abhinav Rastogi, and Sushant Prakash. RLAIF vs. RLHF: scaling reinforcement learning from human feedback with AI feedback. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024.
- [LS23] Martin Lackner and Piotr Skowron. *Multi-Winner Voting with Approval Preferences - Artificial Intelligence, Multiagent Systems, and Cognitive Robotics*. Springer Briefs in Intelligent Systems. Springer, 2023.
- [LSZ92] Vahid Lotfi, Theodor J. Stewart, and Stanley Zionts. An aspiration-level interactive model for multiple criteria decision making. *Comput. Oper. Res.*, 19(7):671–681, 1992.
- [LWM⁺15] Jie Lu, Dianshuang Wu, Mingsong Mao, Wei Wang, and Guangquan Zhang. Recommender system application developments: A survey. *Decis. Support Syst.*, 74:12–32, 2015.
- [LZH⁺20] Wenqiang Lei, Gangyi Zhang, Xiangnan He, Yisong Miao, Xiang Wang, Liang Chen, and Tat-Seng Chua. Interactive path reasoning on graph for conversational recommendation. In Rajesh Gupta,

- Yan Liu, Jiliang Tang, and B. Aditya Prakash, editors, *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 2073–2083. ACM, 2020.
- [MA04] R. Timothy Marler and Jasbir S. Arora. Survey of multi-objective optimization methods for engineering. *Struc. Multidis. Optim.*, 26(6):369–395, 2004.
- [Mit21] Tom M. Mitchell. *Machine Learning*. McGraw-Hill series in computer science. McGraw-Hill, 2021.
- [MK16] Axel C. Mühlbacher and Anika Kaczynski. Making good decisions in healthcare with multi-criteria decision analysis: The use, current research and future development of MCDA. *Appl Health Econ Health Policy*, 14(1):29–40, 2016.
- [MPBEE08] Luis Martínez, Luis G Pérez, Manuel J. Barranco, and Macarena Espinilla-Estévez. *A Knowledge Based Recommender System Based on Consistent Preference Relations*, pages 93–111. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.
- [MR05] Patrick Meyer and Marc Roubens. *Choice, Ranking and Sorting in Fuzzy Multiple Criteria Decision Aid*, pages 471–503. Springer New York, 2005.
- [MRW13] Radu Marinescu, Abdul Razak, and Nic Wilson. Multi-objective constraint optimization with tradeoffs. In Christian Schulte, editor, *Principles and Practice of Constraint Programming - 19th International Conference, CP 2013, Uppsala, Sweden, September 16-20, 2013. Proceedings*, volume 8124 of *Lecture Notes in Computer Science*, pages 497–512. Springer, 2013.
- [Mun07] Giuseppe Munda. *Social Multi-Criteria Evaluation for a Sustainable Economy*. Springer Berlin Heidelberg, 2007.
- [Mur12] Kevin P. Murphy. *Machine Learning: A Probabilistic Perspective*. Adaptive Computation and Machine Learning series. MIT Press, 2012.
- [MW16] Mojtaba Montazery and Nic Wilson. Learning user preferences in matching for ridesharing. In H. Jaap van den Herik and Joaquim Filipe, editors, *Proceedings of the 8th International Conference on*

- Agents and Artificial Intelligence (ICAART 2016)*, Volume 2, Rome, Italy, February 24-26, 2016, pages 63–73. SciTePress, 2016.
- [NM21] John Von Neumann and Oskar Morgenstern. *Theory of Games and Economic Behavior*. Dancing Drum, 2021.
- [PC08] Pearl Pu and Li Chen. User-involved preference elicitation for product search and recommender systems. *AI Mag.*, 29(4):93–103, 2008.
- [Rai68] Howard Raiffa. *Decision Analysis: Introductory Lectures on Choices Under Uncertainty*. Addison-Wesley series in behavioral science. Addison-Wesley, 1968.
- [RRS11] Francesco Ricci, Lior Rokach, and Bracha Shapira. Introduction to recommender systems handbook. In Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul B. Kantor, editors, *Recommender Systems Handbook*, pages 1–35. Springer, 2011.
- [RRS15] Francesco Ricci, Lior Rokach, and Bracha Shapira. *Recommender Systems Handbook*. Springer US, 2015.
- [RSM⁺23] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.
- [Saa80] Thomas L. Saaty. *The Analytic Hierarchy Process: Planning, Priority Setting, Resource Allocation*. Advanced book program. McGraw-Hill International Book Company, 1980.
- [Sar78] Rakesh Kumar Sarin. Elicitation of subjective probabilities in the context of decision-making. *Deci. Sci.*, 9(1):37–48, 1978.
- [Sav51] Leonard J. Savage. The theory of statistical decision. *J. Amer. Stat. Assoc.*, 46(253):55–67, 1951.
- [Sav72] Leonard J. Savage. *The Foundations of Statistics*. Dover Books on Mathematics Series. Dover Publications, 1972.

- [SC83] Ralph E. Steuer and Eng Ung Choo. An interactive weighted tchebycheff procedure for multiple objective programming. *Math. Program.*, 26(3):326–344, 1983.
- [SGM⁺05] Yannis Siskos, Evangelos Grigoroudis, Nikolaos Matsatsinis, José Figueira, Salvatore Greco, and Matthias Ehrogott. *UTA Methods*, volume 233, pages 297–334. Springer New York, 2005.
- [SGMP18] Olivier Sobrie, Nicolas Gillis, Vincent Mousseau, and Marc Pirlot. UTA-poly and UTA-splines: Additive value functions with polynomial marginals. *Eur. J. Oper. Res.*, 264(2):405–418, 2018.
- [SH92] Ahti A. Salo and Raimo P. Hämäläinen. Preference assessment by imprecise ratio statements. *Oper. Res.*, 40(6):1053–1061, 1992.
- [SH01] Ahti Salo and Raimo P. Hämäläinen. Preference ratios in multiattribute evaluation (prime)-elicitation and decision procedures under incomplete information. *IEEE Trans. Syst. Man Cybern. Part A*, 31(6):533–545, 2001.
- [SKKR01] Badrul Munir Sarwar, George Karypis, Joseph A. Konstan, and John Riedl. Item-based collaborative filtering recommendation algorithms. In Vincent Y. Shen, Nobuo Saito, Michael R. Lyu, and Mary Ellen Zurko, editors, *Proceedings of the Tenth International World Wide Web Conference, WWW 10, Hong Kong, China, May 1-5, 2001*, pages 285–295. ACM, 2001.
- [SLB08] Yoav Shoham and Kevin Leyton-Brown. *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge University Press, 2008.
- [SR91] Wan Seon Shin and A. Ravindran. Interactive multiple objective optimization: Survey I - continuous case. *Comput. Oper. Res.*, 18(1):97–114, 1991.
- [SS78] Ralph E. Steuer and Albert T. Schuler. An interactive multiple-objective linear programming approach to a problem in forest management. *Oper. Res.*, 26(2):254–269, 1978.
- [SV19] Denis Sauré and Juan Pablo Vielma. Ellipsoidal methods for adaptive choice-based conjoint analysis. *Oper. Res.*, 67(2):315–338, 2019.

- [SZ18] Yueming Sun and Yi Zhang. Conversational recommender system. In Kevyn Collins-Thompson, Qiaozhu Mei, Brian D. Davison, Yiqun Liu, and Emine Yilmaz, editors, *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08-12, 2018*, pages 235–244. ACM, 2018.
- [TFL⁺09] Tommi Tervonen, José Rui Figueira, Risto Lahdelma, Juscelino Almeida Dias, and Pekka Salminen. A stochastic method for robustness analysis in sorting problems. *Eur. J. Oper. Res.*, 192(1):236–242, 2009.
- [TK74] Amos Tversky and Daniel Kahneman. Judgment under uncertainty: Heuristics and biases. *Sci.*, 185(4157):1124–1131, 1974.
- [TMHI13] Suvi Tarkkanen, Kaisa Miettinen, Jussi Hakanen, and Hannakaisa Isomäki. Incremental user-interface development for interactive multiobjective optimization. *Expert Syst. Appl.*, 40(8):3220–3232, 2013.
- [TVW21] Federico Toffano, Paolo Viappiani, and Nic Wilson. Efficient exact computation of setwise minimax regret for interactive preference elicitation. In Frank Dignum, Alessio Lomuscio, Ulle Endriss, and Ann Nowé, editors, *AAMAS ’21: 20th International Conference on Autonomous Agents and Multiagent Systems, Virtual Event, United Kingdom, May 3-7, 2021*, pages 1326–1334. ACM, 2021.
- [TW87] Martin A. Tanner and Wing Hung Wong. The calculation of posterior distributions by data augmentation. *J. Amer. Stat. Assoc.*, 82(398):528–540, 1987.
- [TW22] Federico Toffano and Nic Wilson. Minimality and comparison of sets of multi-attribute vectors. *Auton. Agents Multi Agent Syst.*, 36(2):44, 2022.
- [UFL⁺24] Mathias Uta, Alexander Felfernig, Viet-Man Le, Thi Ngoc Trang Tran, Damian Garber, Sebastian Lubos, and Tamim Burgstaller. Knowledge-based recommender systems: overview and research directions. *Frontiers Big Data*, 7, 2024.
- [VB09] Paolo Viappiani and Craig Boutilier. Regret-based optimal recommendation sets in conversational recommender systems. In

- Lawrence D. Bergman, Alexander Tuzhilin, Robin D. Burke, Alexander Felfernig, and Lars Schmidt-Thieme, editors, *Proceedings of the 2009 ACM Conference on Recommender Systems, RecSys 2009, New York, NY, USA, October 23-25, 2009*, pages 101–108. ACM, 2009.
- [VB10] Paolo Viappiani and Craig Boutilier. Optimal bayesian recommendation sets and myopically optimal choice query sets. In John D. Lafferty, Christopher K. I. Williams, John Shawe-Taylor, Richard S. Zemel, and Aron Culotta, editors, *Advances in Neural Information Processing Systems 23: 24th Annual Conference on Neural Information Processing Systems 2010. Proceedings of a meeting held 6-9 December 2010, Vancouver, British Columbia, Canada*, pages 2352–2360. Curran Associates, Inc., 2010.
- [VB20] Paolo Viappiani and Craig Boutilier. On the equivalence of optimal recommendation sets and myopically optimal query sets. *Artif. Intell.*, 286:103328, 2020.
- [Via12] Paolo Viappiani. Monte carlo methods for preference learning. In Youssef Hamadi and Marc Schoenauer, editors, *Learning and Intelligent Optimization - 6th International Conference, LION 6, Paris, France, January 16-20, 2012, Revised Selected Papers*, volume 7219 of *Lecture Notes in Computer Science*, pages 503–508. Springer, 2012.
- [Vin92] Philippe Vincke. *Multicriteria Decision-Aid*. Wiley, 1992.
- [VLBB24] Denis E. C. Vargas, Afonso C. C. Lemonge, Helio J. C. Barbosa, and Heder S. Bernardino. An interactive reference-point-based method for incorporating user preferences in multi-objective structural optimization problems. *Appl. Soft Comput.*, 165:112106, 2024.
- [VLHB20] Ivan Vendrov, Tyler Lu, Qingqing Huang, and Craig Boutilier. Gradient-based optimization for bayesian preference elicitation. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 10292–10301. AAAI Press, 2020.

- [WB03] Tianhan Wang and Craig Boutilier. Incremental utility elicitation with the minimax regret decision criterion. In Georg Gottlob and Toby Walsh, editors, *IJCAI-03, Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence, Acapulco, Mexico, August 9-15, 2003*, pages 309–318. Morgan Kaufmann, 2003.
- [WDF⁺08] Jyrki Wallenius, James S. Dyer, Peter C. Fishburn, Ralph E. Steuer, Stanley Zionts, and Kalyanmoy Deb. Multiple criteria decision making, multiattribute utility theory: Recent accomplishments and what lies ahead. *Manag. Sci.*, 54(7):1336–1349, 2008.
- [Wie80] Andrzej P. Wierzbicki. The use of reference objectives in multiobjective optimization. In Günter Fandel and Tomas Gal, editors, *Multiple Criteria Decision Making Theory and Application*, pages 468–486. Springer Berlin Heidelberg, 1980.
- [Woo09] Michael Wooldridge. *An Introduction to MultiAgent Systems*. Wiley, 2009.
- [WSD84] Chelsea C. White, Andrew P. Sage, and Shigeru Dozono. A model of multiattribute decisionmaking and trade-off weight determination under uncertainty. *IEEE Trans. Syst. Man Cybern.*, SMC-14(2):223–229, 1984.
- [Yag88] Ronald R. Yager. On ordered weighted averaging aggregation operators in multicriteria decisionmaking. *IEEE Trans. Syst. Man Cybern.*, 18(1):183–190, 1988.
- [Yue14] Kevin Kam Fung Yuen. The primitive cognitive network process in healthcare and medical decision making: Comparisons with the analytic hierarchy process. *Appl. Soft Comput.*, 14:109–119, 2014.
- [ZCA⁺18] Yongfeng Zhang, Xu Chen, Qingyao Ai, Liu Yang, and W. Bruce Croft. Towards conversational search and recommendation: System ask, user respond. In Alfredo Cuzzocrea, James Allan, Norman W. Paton, Divesh Srivastava, Rakesh Agrawal, Andrei Z. Broder, Mohammed J. Zaki, K. Selçuk Candan, Alexandros Labrinidis, Assaf Schuster, and Haixun Wang, editors, *Proceedings of the 27th ACM International Conference on Information and Knowledge*

- Management, CIKM 2018, Torino, Italy, October 22-26, 2018*, pages 177–186. ACM, 2018.
- [ZD16] Constantin Zopounidis and Michael Doumpos. *Multiple Criteria Decision Making: Applications in Management and Engineering*. Multiple Criteria Decision Making. Springer International Publishing, 2016.
- [ZGD⁺15] Constantin Zopounidis, Emiliós Galariotis, Michael Doumpos, Stavroula Sarri, and Kostas Andriosopoulos. Multiple criteria decision aiding for finance: An updated bibliographic survey. *Eur. J. Oper. Res.*, 247(2):339–348, 2015.
- [ZLJ21] Qian Zhang, Jie Lu, and Yaochu Jin. Artificial intelligence in recommender systems. *Compl. Intell. Syst.*, 7(1):439–457, 2021.
- [ZRL⁺18] Luisa M. Zintgraf, Diederik M. Roijers, Sjoerd Linders, Catholijn M. Jonker, and Ann Nowé. Ordered preference elicitation strategies for supporting multi-objective decision making. In Elisabeth André, Sven Koenig, Mehdi Dastani, and Gita Sukthankar, editors, *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2018, Stockholm, Sweden, July 10-15, 2018*, pages 1477–1485. International Foundation for Autonomous Agents and Multiagent Systems Richland, SC, USA / ACM, 2018.
- [ZTK14] Edmundas Kazimieras Zavadskas, Zenonas Turskis, and Simona Kildienė. State of art surveys of overviews on MCDM/MADM methods. *Technol. Econ. Dev. Econ.*, 20(1):165–179, 2014.
- [ZW76] Stanley Zionts and Jyrki Wallenius. An interactive programming method for solving the multiple criteria problem. *Manag. Sci.*, 22(6):652–663, 1976.
- [ZZZ⁺20] Kun Zhou, Yuanhang Zhou, Wayne Xin Zhao, Xiaoke Wang, and Ji-Rong Wen. Towards topic-guided conversational recommender system. In Donia Scott, Núria Bel, and Chengqing Zong, editors, *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 4128–4139. International Committee on Computational Linguistics, 2020.